10th International Conference on
Computer and IT Applications in the Maritime Industries

COMPIT’11

Berlin, 2-4 May 2011

Edited by Volker Bertram
This work relates to Department of the Navy Grant N62909-11-1-1049 issued by the Office of Naval Research Global. The United States Government has a royalty-free license throughout the world in all copyrightable material contained herein.
Welcome to the 10th COMPIT Conference

Greetings from Erik van der Noordaa, CEO GL Group

The maritime industry could not survive without modern information technology. Today, the informed public and legislative bodies all over the world expect that shipping contributes to fighting global warming and reduces its carbon footprint. While the industry encounters a period of tightening emission limits, commercial pressures and fuel surcharges, substantial savings are only feasible by advanced applications and programs.

This simple observation corresponds with the idea of the Conference for Computer and IT applications in the Maritime Industry. COMPIT has and will continue to promote the dialogue between the maritime and IT industries as well as key research groups from academia. If the conference had not been initiated ten years ago, it would be high time to get started and enter a new phase of energy efficiency. While there are many other exciting topics on the agenda of the 10th Conference for Computer and IT applications in the Maritime Industry, the ultimate challenge is to provide smart technical solutions.

Today’s ability to undertake even more complex simulations vastly improves the scope and sophistication of designs and their integrated environments. This is reflected by the introduction of commercial software tools which help to address the dominant topics of fuel efficiency and emission reduction.

Looking beyond the urgent requirements for shipping, advances in individual and swarm intelligence are opening up new applications in surveying and search tasks, in offshore, oceanography and navy applications. The conference covers a broad spectrum of themes which apart from the life cycle of ships, offshore structures and equipment, address frontier developments of a highly complex nature.

I wish everyone an inspiring exchange of ideas!

Erik van der Noordaa
CEO GL Group
Index

Milovan Peric, Volker Bertram
*Trends in Industry Applications of CFD for Maritime Flows*  8

Anthony S. Daniels, Morgan C. Parker, David J. Singer
*Intelligent Ship Arrangements (ISA) Passage Variable Lattice Network Templating Application*  19

Valery Mrykin, Alexander Leshchina, Alexander Antonov, Manucher Dorri
*Hardware-in-the-Loop Simulation System for Submersible Dynamics with Visualization of Environment Near the Sea Bottom for Training of Submersible Crews*  31

Auke van der Ploeg
*A Comparison of Strategies for the Optimization of a Ship’s Aft Body*  43

Philippe Renard
*IT Supported Maintenance, Inspection and Repair of Offshore Floating Windmill Farms*  56

Tobias Huuva, Magnus Pettersson
*Optimization for Improved Propulsive Efficiency and Increased Bollard Pull*  66

Marco Bibuli, Gabriele Bruzzone, Massimo Caccia, Thomas Vögele, Markus Eich, Alberto Ortiz, Leonidas Drikos, Yannis Koveos
*MINOAS Project: Localization, Task-Allocation and Path-Planning Architecture System*  74

Wilfried Abels
*Using a Mathematical Symbolic Solver to Transfer Analytic Theory to Numerical “First-Principle-Methods”*  88

Thomas Koch
*Simulating the Production of Future Marine Products*  98

Leif Christensen, Niklas Fischer, Sven Kroffke, Johannes Lemburg, Reinhard Ahlers
*Cost Effective Autonomous Robots for Ballast Water Tank Inspection*  109

Rikard Mikalsen
*Patent Protection of Software-Based Inventions in the Maritime Industry*  124

Roberto Penas, Carlos González
*Integration of DB Oriented CAD Systems with Product Lifecycle Management*  134

Rolf Nagel
*Software Applications for Life-Cycle Management at a Shipyard*  148

David Thomson
*The Role of Social Computing in Shipbuilding*  156

Josep Aulinas, Yvan R. Petillot, Xavier Lladó, Joaquim Salvi, Rafael Garcia
*Vision-Based Underwater SLAM for the SPARUS AUV*  171

Andrea Caiti, Vincenzo Calabrò, Sergio Grammatico, Andrea Munafò, Mirko Stifani
*Control-Oriented Modeling of an Underwater Wave Glider*  181
<table>
<thead>
<tr>
<th>Title</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>Noise Limits in Harbour – A Method to Optimise the Noise Emission of Vessel’s Ventilation Systems</td>
<td>188</td>
</tr>
<tr>
<td>Evaluating the Feasibility of Open Source CFD Software for Small Ship Design Offices</td>
<td>195</td>
</tr>
<tr>
<td>Numerical Hull Series for Calm Water and Sea-Keeping</td>
<td>206</td>
</tr>
<tr>
<td>CAD-Centric, Multi-Source Data and Information Management</td>
<td>221</td>
</tr>
<tr>
<td>Generating Contract Scenarios for the Conceptual Design Optimization of Non-Transport Vessels</td>
<td>237</td>
</tr>
<tr>
<td>Optimal Hull-Form Design Subject to Epistemic Uncertainty</td>
<td>245</td>
</tr>
<tr>
<td>Virtual Reality Supported Assembly Planning in the Shipbuilding Industry</td>
<td>253</td>
</tr>
<tr>
<td>Reduced Beam Model – The Seek after Adequate Accuracy with Feasible Performance in Global Structural FE-Analysis of a Naval Vessel</td>
<td>264</td>
</tr>
<tr>
<td>Discrete Event Production Simulation and Optimisation of Ship Block Erection Process</td>
<td>271</td>
</tr>
<tr>
<td>A Particle Swarm Algorithm-Based Optimisation for High-Strength Steel Structures</td>
<td>283</td>
</tr>
<tr>
<td>A Novel Ship Subdivision Method and its Application in Constraint Management of Ship Layout Design</td>
<td>292</td>
</tr>
<tr>
<td>A Machine-Learning Approach to Predict Main Energy Consumption under Realistic Operational Conditions</td>
<td>305</td>
</tr>
<tr>
<td>Evaluating Evacuation Simulation Results in a Virtual Reality Environment</td>
<td>326</td>
</tr>
<tr>
<td>Introducing Highly-Efficient CAE Pre- and Post-Processing Solutions in Maritime Design</td>
<td>335</td>
</tr>
<tr>
<td>The Use of Paramarine and modeFRONTIER for Ship Design Space Exploration</td>
<td>347</td>
</tr>
<tr>
<td>Non-Disruptive Development of a Next-Generation CAD Application Program</td>
<td>359</td>
</tr>
</tbody>
</table>
Sebastian D. Bade, Andreas Junglewitz
*Automated Strength Analysis for Propeller Blades*

Andrea Caiti, Vincenzo Calabrò, Giuseppe Conte, Pierre Drap, David Scaradozzi, Silvia Zanoli
*Underwater Archaeology Surveys with Autonomous Robots*

Jukka Ignatius
*Fleet Wide Operational Reporting - Performance and Environment*

Ties van Bruinessen, Hans Hopman, Thomas DeNucci, Bart van Oers
*Generating More Valid Designs during Design Exploration*

Stefan Harries, Fabian Tillig, Marc Wilken, George Zaraphonitis
*An Integrated Approach for Simulation in the Early Ship Design of a Tanker*

Richard Pawling, David Andrews
*Design Sketching – The Next Advance in Computer Aided Preliminary Ship Design?*

Hamid Hefazi, Josh Hamel, Adeline Schmitz
*Development of Decomposition-Based Design Optimization Tools for Ship Design*

Dirk Steinhauer
*The Simulation Toolkit Shipbuilding (STS) – 10 Years of Cooperative Development and Interbranch Applications*

Anna Bruns, Katja Christiansen, Dirk Rossow
*FSG.EcoPilot – An Onboard Tool for Fuel Efficient Speed Profile Selection*

David E. Hess, William E. Faller
*Ensuring Stability in Synergistic Computing: Combining Flow Simulations and Neural Nets to Predict Maneuvering*

Carlos V. M. Santos
*The Employment of 3d Collaborative Environment from the Very Beginning of the Conceptual Warship Design Stage*

Marc Wilken, Henner Eisen, Matthias Krömer, Christian Cabos
*Hull Structure Assessment for Ships in Operation*

Fernando Henrique e Paula da Luz, Humberto S. Makiyama, Liria Matsumoto Sato, Kazuo Nishimoto
*A New Proposal of Communication in a Numerical Model Basin Simulator*

Denis Morais, Mark Waldie, Darren Larkins
*Driving the Adoption of Cutting Edge Technology in Shipbuilding*

Tomi Holmberg, Shaun D. Hunter
*Increasing Efficiency in the Ship Structural Design Process*

Index of authors

Call for Papers COMPIT’12
Trends in Industry Applications of CFD for Maritime Flows

Milovan Peric, CD-adapco, Nuremberg/Germany, milovan.peric@de.cd-adapco.com
Volker Bertram, FutureShip, Hamburg/Germany, volker.bertram@GL-group.com

Abstract

The paper surveys developments in CFD applications for maritime structures (ships, propellers and offshore structures) over the past decade. Progress is significant in integrating the process chain, particularly more automated model generation. Increased hardware power and progress in various aspects of the flow solvers allow more sophisticated applications and wider scope of applications. Selected examples taken from industry and research applications show the increasing importance of CFD in earlier design stages.

1. Introduction

Computational fluid dynamics (CFD) denotes collectively techniques solving equations describing the physics of flows. We interpret “CFD” here as techniques solving Euler, RANSE (Reynolds Averaged Navier-Stokes Equations) or Navier-Stokes equations, using field methods, Ferziger and Peric (2010), Bertram (2012).

Fig.1: Simulation of coolant flow in an engine block in 1988

Fig.2: Modern CFD analyses with high level of detail
CFD became a research field in the late 1960s. First commercial CFD software appeared in the 1980s including codes like PHOENICS, FLUENT, STAR-CD, CFX, TASCFLOW, and FLOW3D. By today’s standard, these codes were very limited in terms of complexity of geometry and physics. Applications were severely limited by the available computer power in those days. An example may illustrate how CFD progressed over the decades. In 1988, an advanced CFD application in the automotive industry investigated the coolant flow in an engine block using some 10000s of cells, Fig.1.

Two decades later, the progress in CFD allows the simulation of fluid and heat flows in engine compartments with some 700 parts and typically around 30 million control volumes, even 100 millions for detailed studies, Fig.2. However, the increase in grid size and associated capturing of geometry and flow details is only one aspect of the CFD developments of the past decades. It is a perhaps surprising fact that computational times have increased over the years. The demand for ever more ambitious simulations (in terms of cell count and flow complexity) thus outpaced the exponential growth in computational power.

Despite the increase in average computational time, CFD projects today are often noticeably shorter than they were two decades ago. This is due to less frequent re-modelling and re-analyses, and also generally significantly reduced time in pre-processing. The reason is that CFD tools have become more user-friendly, Fig.3. This is perhaps best illustrated in the case of integrated design environments, e.g. the Friendship Framework, Abt and Harries (2007). The integrated design environment combines freeform hull description using parametric modelling, interfaces to most modern CFD solvers including STAR-CCM+, several optimization algorithms, and software to handle process management and user interface. The design engineer can then work on simulation driven designs (e.g. of hulls, appendages or propellers) with one integrated user interface from model generation to post-processing. The user-friendliness of this approach has certainly lowered thresholds in using CFD for designers.

There are many more aspects that have in sum advanced the wide acceptance of simulation first as a design and more recently as an optimisation tool in industry. The most important among these aspects are:
- The ability to handle complex geometry with all relevant details, including moving parts;
- Efficient simulation process (from geometry to solution, parametric studies, optimization studies, user interface...);
- Adequate modelling of turbulence, free-surface effects and cavitation;
- Coupled simulation of flow and flow-induced motion (and in some cases deformation) of bodies.

These will be discussed in more detail in the following.

2. Key aspects of progress in CFD for maritime flows

2.1. Handling of complex geometry

During the past decade, the ability to handle complex geometry with all relevant details has been greatly improved. Components that have contributed to this trend include:

- Tools for automatic and user-friendly manual repair of CAD models (which are often imperfect) have been developed; IGES files coming from designers frequently feature overlaps and gaps. These are not problematic for design purposes (e.g. volume computations are required for ship stability and capacity), but frequently lead to fatal errors for CFD grid generation.
- Surface-wrapping tools have been introduced, which create a closed surface around assemblies of solid parts, Fig.4;
- Tools for automatic generation of polyhedral, trimmed hexahedral or extruded meshes have been developed;
- Automatic and manual definition of local mesh refinement requirements have been created, based on:
  - local curvature, proximity of other walls etc.;
  - pre-defined regions, interfaces, wakes etc.;
  - indication or estimate of numerical error...

Grid generation has improved, making it easier to generate high-quality grids for accurate CFD simulations. A key aspect for complex geometries consisting of many components (such as offshore platforms in the maritime context) is geometry recognition. The software then recognizes automatically cylinders (with extrusion along centreline, using prismatic cells) and thin solids or gaps, with projection from one side to another, using prismatic cells).

![Fig.4: Re-meshed surface of a complete oil rig after surface-wrapping (left) and simulated air flow field around the oil rig with surface pressure and wake velocity (right)](image-url)

More sophisticated analyses for ships and offshore platforms employ a variety of techniques that have become widely available (through commercial and open-source software):
• The ability to handle moving parts using morphing, sliding interfaces or overlapping grids (e.g. propellers, rudders etc.);
• The ability to model complete systems rather than single parts (e.g. ship with all appendages, Fig.5, complete oil platforms, etc.);
• The ability to easily replace geometry and perform a new simulation (automation of simulation process; e.g. ship with and without a wake equalizing nozzle, Fig.6).

Fig.5: Ship with rudder and propeller illustrating the trend towards modeling complete systems

Fig.6: Analyses of ship without (left) and with (right) nozzle to assess quantitatively the effect of the nozzle at full scale before deciding on the investment; block-structured approach allows exchanging block with nozzle and rapid re-analyses

2.2. Turbulence modelling

Turbulence modelling was in the “villain” of the 1980s and 1990s. Unsatisfactory results were often blamed on turbulence modelling. Several dedicated validation workshops have shed more light on adequacy and inadequacy of turbulence modelling for marine flows, Figs.7-9. For most applications in industry practice, the importance of turbulence modelling is over-rated. Turbulence models play a significant role on the flow structures and resulting resistance of bare hulls, as investigated in most validation studies. However, the propeller behind the ship dominates flows and reduces the effect and importance of the turbulence model. Since only the propulsion case is relevant for industry, turbulence modelling is thus of lesser importance for classical resistance & propulsion applications. For seakeeping, the free-surface effects dominate anyway. This leaves manoeuvring and propeller flow investigations as areas of application where turbulence modelling remains an important issue.

For most applications in industry, the standard $k$-$\varepsilon$ or $k$-$\omega$ turbulence models are adequate. In order to predict secondary flows better, more sophisticated models are needed. The Reynolds-stress model
(RSM) is then frequently a popular and appropriate option. A special turbulence model is needed to predict transition from laminar to turbulent flow, e.g. when predicting resistance of a competitive sailing yacht. Such models are also available. For predicting noise sources, wall vibration etc., large-eddy-simulation (LES) or detached eddy simulation (DES) type of analyses with special subgrid-scale turbulence models are used. These are rather subject to research than state of the art in industry. The CFD expert needs to select the most appropriate model for any given analysis task (and may decide not to use any turbulence modelling…).

Fig.7: RSM turbulence model for KLVCC test case; wind tunnel test (top) and CFD (bottom)

Fig.8: Measured and predicted velocity profiles

Fig.9: Resistance prediction for KLVCC without free surface, using RSM turbulence model
2.3. Modelling of free-surface effects

Ships operate at the interface of water and air. Correspondingly, free-surface flows are of prime interest for naval architects. The wave resistance of a ship is one example, as this component of the total resistance offers the largest improvement potential for small to moderate changes in the hull shape. Other applications of free-surface flows are seakeeping (interaction with waves), slamming (external impact due to waves) and sloshing (internal impact in partially filled tanks). Interface-capturing methods (volume of fluid, two-phase flow, level set, etc) allow the simulation of highly nonlinear free-surface flows. Where the two fluids (typically water and air) are not expected to mix, a sharp interface (within one control volume) can be obtained, Fig. 10. This minimizes numerical mixing. Resulting quantities of engineering interest, e.g. induced loads in tanks with sloshing, are so well predicted that such simulations are widely accepted by classification societies for load determination in strength analyses. Arbitrary free-surface deformation, even trapped gas bubbles or detaching droplets are adequately accounted for, as gravity and surface tension effects are included. If deemed necessary, both gas and liquid can be modelled as compressible fluids. Phase change models (cavitation, boiling) may be integrated into this method to allow more complex phenomena to be modelled. Despite the significant progress in free-surface modelling, research continues in this field, as the modelling of breaking waves can still be improved in terms of air mixing and turbulence interaction with the free surface. In regions, where in reality white foam appears (mix of air and water), current CFD simulations show smeared surfaces, Fig. 12, and predict the propagation of these waves less accurately.

Fig.10: Free-surface resolution within one cell after 100 periods (= 20 min in corresponding real time)
Fig.11: Measured and computed pressures at one point in the tank
Fig.12: Computed wave field around DTMB 5415 (destroyer geometry); smeared surface at bow and stern where waves break, sharp surface elsewhere
2.4. Cavitation modelling

Cavitation modelling may seem as an extension of free-surface modelling, as it involves the interface between water and gas (or vapour in this case), which is a priori unknown and part of the solution. Albeit, in this case, also the growth and collapse of cavitation bubbles need to be described adding a further complexity.

Fig. 13: Cavitation around NACA0015 foil at 10.3° angle of attack; experiment of HSVA (top) and CFD simulation (bottom)
If the aim is to avoid cavitation, one only needs to predict its onset (usually expressed by a pressure below saturation level). However, in most propellers and several rudders, cavitation is unavoidable. If cavitation cannot be avoided, its effect on performance needs to be assessed, hence the need for cavitation modelling. Despite their theoretical shortcomings, models based on bubble dynamics (Rayleigh-Plesset equation) have proven robust and sufficiently accurate for most industrial applications, Fig.13. One additional equation for volume fraction of vapour is then solved, with two parameters: (1) seed density in liquid (number of seeds per m$^3$ of liquid) and on solid walls (surface-roughness effects); (2) initial (representative) radius of seed bubble. These parameters are related to the “liquid quality” and depend on region (for sea water) or treatment (like de-gassing or filtering in a cavitation tunnels).

RANSE simulations with cavitation modelling have become part of modern design procedures for advanced propulsors, such as Voith Schneider Propellers (VSPs). For illustration, Fig.14 shows a snapshot with the extent of vapour regions on each blade of a VSP in off-design condition. The associated diagram shows propulsor performance (torque) full-scale measurements and CFD results. The cavitation model significantly improves the quality of the prediction. Fig.15 shows another application taken from industry practice. Rudders behind highly loaded propellers are susceptible to cavitation and associated erosion which endangers the ship. CFD is by now regularly employed to predict location and extent of cavitation on rudders in these cases. The concerned regions are then often built in more enduring steel, unless local redesign avoids the formation of cavitation erosion.

![Fig.14: VSP in off-design condition; performance in full-scale measurements and simulations (left) and simulated cavitation extent at all blades (right); source: Voith](image1)

![Fig.15: CFD prediction of rudder cavitation (left) and observed erosion at actual ship (right)](image2)
2.5. Motion of floating bodies

For a variety of seakeeping problems, implicitly coupled simulations of flows and flow-induced motions of floating bodies (ships or offshore structures) are desired. These simulations should be implicit, as implicit simulations pose no restrictions on the time-step size for stability reasons. The time step can be chosen then according to accuracy requirements. Highly nonlinear motions (e.g. launching of free-fall lifeboats with subsequent water entry and resurfacing) are better handled in implicit simulations. Such rigid-body motions of freely moving ships have been presented for a variety of applications including many industry projects, e.g. for slamming investigations, Fig.16. The simulations can handle in principle all complexity required in offshore and naval architectural applications, including multi-body configurations moving relative to each other, possible coupling between bodies (via elastic moorings, rigid connections, or flexible links with constraints), inclusion of external forces (e.g. thrusters, mooring, towing), or relative motion of system components (e.g. propellers).

Fig.16: Slamming investigation for a megayacht; CFD snapshot (left) and validation with full-scale measurements (right)

2.6. Fluid-Structure-Interaction (FSI)

Coupled simulation of flow and flow-induced deformation of solid structures have evolved more recently for marine applications. FSI is important for relatively soft structures, for very large ships (e.g. whipping and springing, i.e. hydro-elastic vibrations of the ship hull) and offshore structures (like floating airports) as well as for better prediction of impact loads (slamming and sloshing). So far, coupling of RANS CFD codes and finite-element codes (for the structural analyses) is usually explicit, making the computations inefficient to the point where they are not applicable to most practical problems. Implicit coupling (as already in place for rigid-body motions in waves) is required for robustness and computational efficiency. On the other hand, the structural model can be simplified (e.g. treating the ship as a beam subject to bending and torsion). Such simplified structural models with implicit coupling have already been implemented, e.g. Oberhagemann et al. (2008), Fig.17.

Fig.17: Green water on deck after one slamming event (left); measured and computed accelerations in bow region for a rigid and an elastic ship structure (right)
3. Trends

Computer hardware continues to become more and more powerful. Highly parallel computing environments have become affordable even for small and medium enterprises. The appetite grows at least as fast as the more powerful capabilities become available. Higher demands from simulations come in various forms:

- More complete system analysis, with all geometrical details;
- More transient simulations (URANS (= unsteady RANS), DES and LES);
- Prediction of pressure fluctuation and noise sources (turbulence, cavitation);
- More fluid-structure-interaction (slamming, sloshing) and other multi-physics (wind, fire, pollution etc.) applications;
- Simulation of full manoeuvring tests (circle, zig-zag, etc.) already in conceptual design;
- Simulation of interaction (ship + ice, ship + platform, ship + ship etc.).
- More automatic optimization studies...

Of the many developments on the horizon, we select two for illustrative purposes, namely coupling CFD with formal optimization in ship design and coupling CFD in simulators for training and assessment purposes.

3.1. Design optimization

Optimization strives to maximize desired features (objectives) subject to constraints. The difficulty in practice lies in the expressing all objectives and constraints in mathematical functions of sufficient accuracy and numerical efficiency, Bertram (2003). This task is often more difficult than it sounds. It requires

- a smart engineer, who is able to model the problem at hand with just the right level of detail and sets up a design search space that is large enough to find significant improvements, yet small enough to allow efficient exploration;
- a tool to help the engineer to convert his model into an efficient optimization process

The mathematical algorithm for the actual optimization is secondary. Simulations play a pivotal role as they are used to assess a design, e.g. in terms of required power in fuel efficiency optimizations. Care is needed when changes in candidate designs are small, as then the required accuracy in turn is high. Fortunately, often the task is to find the design that is optimum, rather than an accurate determination of the object function (e.g. required power) for this design. Thus if the relative ranking is right independent of (similar or constant) errors in objective function the task can be solved even with usual discretization or model errors.

Optimization software, e.g. the Friendship Framework, Abt and Harries (2007), is available to guide an automated simulation process towards a (near) optimum design, combining automatic generation of parameterized geometry, automatic mesh generation, automatic CFD simulation and analysis, and subsequent automatic determination of new parameters. Examples of marine applications include funnel optimization for minimum smoke dispersion on the deck of a yacht, Harries and Vesting (2010), and the optimization of the nozzle geometry for a Voith radial propeller, Palm et al. (2010).

3.2. Simulation of experiments

CFD can be used to generate data sets for subsequent fast evaluation in design and operation. One example are ship simulators, used to train ship crews in the handling of ships. The manoeuvring coefficients for these simulators used to come from dedicated model basin tests. More recently, Voith has replaced model tests by CFD simulations to determine the manoeuvring coefficients, Fig.17.
4. Final remarks

No matter what the software can do, it remains just a tool. How quickly and well problems get solved depends on the craftsman using the tool – the engineer remains indispensable. This has also been phrased as “the pilot is more important than the plane”.

With this in mind, proper training for CFD analyst is vital. Engineers need to be educated how to best use the tools at hand, be it theoretical or qualitative analysis, numerical simulation (based on different tools involving different limitations and required effort) or experiments (in model or full scale). Never fall in love with your model; be aware of different approaches and choose intelligently.

In the end, the engineer is paid for his modelling skills, choosing the model that offers reliable and sufficiently accurate answers obtained with minimum cost (in terms of time and money).

References


HARRIES, S.; VESTING, F. (2010), Aerodynamic optimization of superstructures and components, 9th Conf. on Computer Applications and Information Technology in the Maritime Industries (COMPIT), Gubbio, pp.335-347


Intelligent Ship Arrangements (ISA)  
Passage Variable Lattice Network Templating Application

Anthony S. Daniels, University of Michigan, Ann Arbor/USA, adaniels@umich.edu  
Morgan C. Parker, University of Michigan, Ann Arbor/USA, mcparker@umich.edu  
David J. Singer, University of Michigan, Ann Arbor/USA, djsinger@umich.edu

Abstract

The Intelligent Ship Arrangements (ISA) system, under development at the University of Michigan, is a Leading Edge Architecture for Prototyping System (LEAPS) compatible software system that assists the designer in developing rationally-based arrangements. These arrangements satisfy the design specific needs, general Navy requirements and standard practices to the maximum extent practicable. This software system is intended to be used following or as a latter part of Advanced Ship and Submarine Evaluation Tool (ASSET) synthesis. Recent improvements to the current ISA application are discussed. The issues covered are: the introduction of passage templating on a zone deck level as well as full deck templating of passages. Examples of both types of templating will be presented. In addition, the Passage Variable Lattice Network (PVLN) Agent’s passage and staitower stitching algorithms will be detailed.

1. Introduction

Parsons et al. (2008) introduced the ISA platform as a software package for the quantification and optimization of general arrangements in surface ships. It is a native C++ / Microsoft Foundation Class (MFC) application, which is intended to be used as a post processing step or latter phase of the U.S. Navy’s ASSET synthesis process, ASSET (2007). This allows the users to gain insight into the general arrangements of the vessel at a much earlier stage in the design process, the Analysis of Alternatives level of design. Generating arrangements earlier in the ship design process opens up the opportunity to perform more detailed analyses (such as survivability) earlier. ISA provides four new design enabling capabilities to the naval architecture community as well as an overall paradigm shift in general arrangements theory and practice. It provides:

- The ability to capture U.S. Navy design rules, regulations, best practices, and intent in a quantifiable and consistent manner using the ship specific template databases (e.g. NAVSEA 1992 requirements). This provides an important knowledge capture capability to the naval architecture community.
- The ability to quantify and compare general arrangements of vessels in a rational process.
- The ability to apply that rational process to the improvement and optimization of the general arrangements for a ship design.
- The ability to directly integrate general arrangements trades studies into the Analysis of Alternatives level of design.

![ISA overall schematic](image)

Fig. 1: ISA overall schematic
ISA gets its inputs from an ASSET generated LEAPS database, LEAPS (2006), and a ship specific template library, Fig. 1. The ASSET / LEAPS database provides the ship geometry as well as manning, major components, etc. The ship specific template library database, or ISA Library, provides a set of template spaces and accompanying constraints for that specific ship type to be used in the population of the model. From these two sources of information the ship model is created and the optimization on the general arrangements for the vessel can be performed. Resulting arrangements can then be exported back into the LEAPS database as new Concept objects.

ISA takes inputs from two separate databases and generates arrangements for the designer. The process in which this happens is a three step process. The first step is the allocation of the spaces in the ship template to the various zone decks of the ship. Recall that a zone deck is one major structural region of the ship, such as a region surrounded by one deck and one subdivision. The allocation of the spaces is handled by a Hybrid Genetic Algorithm – Multi Agent System (HGA-MAS). Once an optimal allocation is achieved the algorithm then enters a two-step arrangement phase where the spaces within the individual zone decks are topologically and then geometrically arranged. This arrangement portion of the process is handled by a genetic algorithm and stochastic growth algorithms. For detailed information on ISAs design and its algorithms refer to Parsons et al. (2008).

In Daniels et al. (2009), the ISA platform was given modifications and upgrades to the manner in which the compartment and access networks were handled in the optimization algorithms. The methodology that was introduced was termed a Passage Variable Lattice Network (PVLN) and allowed the application to represent more complicated passage configurations above and beyond the standard H and parallel passage, configurations available in phase one of development. The PVLN creates a much more adaptive and robust compartment and access model, Fig. 2. This paper will discuss further modifications to the PVLN system with the addition of passage and stairtower templates both at the zone deck level and the full deck level.

2. ISA Passage Variable Lattice Network Review

The following section is an excerpt from Daniels et al. (2009), whose purpose is to provide continuity and review. A new passage formulation was introduced for the development of more realistic compartment and access networks in a ship. This was needed because most ships often have passage configurations that do not follow the H and II configurations that were used in the previous round of ISA development. Therefore a more generic method of passage network generation had to be devised. After studying General Arrangements drawings from sample ships, it was determined that a Passage Variable Lattice Network (PVLN) was a good candidate for representing most of the various types of
passage networks that are seen on ships. Most passage networks on a ship follow a city grid style lattice with intersecting longitudinal and transverse (athwartships) passages as seen in Fig. 3. The city grid street pattern from civil engineering was the inspiration behind this methodology. It should be mentioned that these interfacing passages are not required to be orthogonal in layout. In addition, each zone-deck’s lattice passage members can be optionally fixed in their geometry and also linked to neighboring zone-deck (ZD) lattices.

The result is a collective passage lattice structure that can span every level of the ship and represent a large number of possible passage configurations. These lattice networks do not have to be symmetric, nor do their passage segment members have to span an entire zone-deck. In addition, passage segments in the lattice may have multiple waypoints to represent more complex inter-node geometries. The individual passages will also have geometry configuration controls including:

- Minimum and maximum segment transition angles
- Minimum and maximum segment lengths
- Orthogonality restrictions
- Limit controls on number of waypoints allowed

![Fig. 3: Passage Variable Lattice Network (PVLN)](image_url)

The variable aspect of the PVLN refers to the passage’s ability to vary in both geometry as well as existence. Each zone-deck of the ship has a passage lattice of M longitudinal passages by N transverse passages as an upper limit on the lattice size. These parameters are settable, however practical use limitations suggest that grid sizes will most likely be less than ten by ten. Whether or not a passage is used in the arrangement is a variable Boolean flag that is manipulated via the optimization algorithm. In a given round of arrangement generation, the passage and access network are generated based on which passages are active at that moment. From this starting point, a stochastic growth algorithm is applied to each space until a stable arrangement has been achieved. This process is repeated generating multiple geometries per allocation before cycling back to the allocation level of detail for the next generation. For more in depth discussion of the baseline PVLN system please refer to either Daniels et al. (2009,2010).

3. Preliminary PVLN Studies Summary

One of the main observations of the PVLN Demonstration code was that it had two primary deficiencies from a performance standpoint. The first deficiency was that through the optimization process it generated too many passage configurations that were not realistic. For example a Naval Architect would not design a passage like the one illustrated in Fig. 4. There are too many passages in this example, and they create large amounts of unusable void regions.
The second area for improvement was the fact that the solver took too long (too many generations) to reach a reasonably good solution (on the order of magnitude of multiple hours for a single zone deck). Both problems made the PVLN system, while extremely flexible and comprehensive, computationally cost prohibitive from a designers standpoint. Optimizing a single zone deck would take on the order of multiple hours. In order to address these two problems the development team decided that some of the flexibility should be reduced by applying templates, or standard commonly used configurations.

4. Passage and Stairtower Templating

Templates are patterns of configurations that are commonly used in design practice. A good example of a template from the previous phase of development was the standard longitudinal H pattern. This type of pattern is used heavily on the damage control deck because it provides two longitudinal access passages and a single athwart ship passage. The templates are generated parametrically. All of them are based on starboard, port, fore and aft offset dimensions. Not all parameters are used for a given template. For example the standard longitudinal H patterns are created in the following manner. The two longitudinal passages are created first and aligned with their starboard and port offsets. The offsets can be either uniform (port and starboard have the same offset), or they can be two independent offsets. Once the two longitudinal passages are placed, then the transverse passage is placed at the YZ plane of the center of the zone deck’s axis aligned bounding box. The passages are still allowed to be pushed by spaces in the growth phase of the geometry generation so the template provides a starting point that is more realistic and practical.
The following additional templates are implemented in ISA:

- Single Longitudinal
- C-Pattern Fore
- Standard Longitudinal Parallel
- Single Transverse
- Y-Pattern Aft
- Standard Transverse Parallel
- Single Cross
- Y-Pattern Fore
- Standard Transverse H
- C-Pattern Aft
- Standard Longitudinal H
- S-Pattern One
- S-Pattern Two
- S-Pattern Three
- S-Pattern Four
Fig. 6: Example of improper passage template application
It should be noted that with all of the templates, not every zone deck shape can accommodate all of the different passage templates. For example, Fig. 6 shows an improper choice of passage template for a narrow triangular zone deck of the ship. A Naval Architect would not use a longitudinal H pattern in this situation. In addition, depending where in the ship the zone deck is created, different sets of rules and guidelines for which passage templates can be used.

In order to incorporate these two constraining situations, the PVLN Agent was given a series of editable lookup tables to determine what passage templates could be used. This set of lookup tables is currently hard coded into the agent, but could easily be made available to the user in future generations of the software. The lookup table system is keyed on two parameters, the type of zone deck and the shape of the zone deck, Table I.

### Table I: Passage template lookup key factors

<table>
<thead>
<tr>
<th>Zone Deck Types</th>
<th>Zone Deck Shape Patterns</th>
</tr>
</thead>
<tbody>
<tr>
<td>Above Damage Control Deck Hull</td>
<td>Rectangle Longitudinal High Aspect Ratio</td>
</tr>
<tr>
<td>Below Damage Control Deck Hull</td>
<td>Rectangle Longitudinal Small Aspect Ratio</td>
</tr>
<tr>
<td>Damage Control Deck Hull</td>
<td>Rectangle Transverse High Aspect Ratio</td>
</tr>
<tr>
<td>Damage Control Deck Superstructure</td>
<td>Rectangle Transverse Small Aspect Ratio</td>
</tr>
<tr>
<td>Superstructure</td>
<td>Trapezoid Fore Large Angle</td>
</tr>
<tr>
<td>Example of a Trapezoid Fore Medium Angle Zone Deck</td>
<td>Trapezoid Fore Medium Angle</td>
</tr>
<tr>
<td></td>
<td>Trapezoid Fore Small Angle</td>
</tr>
<tr>
<td></td>
<td>Trapezoid Aft Large Angle</td>
</tr>
<tr>
<td></td>
<td>Trapezoid Aft Medium Angle</td>
</tr>
<tr>
<td></td>
<td>Trapezoid Aft Small Angle</td>
</tr>
<tr>
<td></td>
<td>Square</td>
</tr>
<tr>
<td></td>
<td>Triangle Aft</td>
</tr>
<tr>
<td></td>
<td>Triangle Fore</td>
</tr>
</tbody>
</table>

Similarly to the passages, the stairtowers were implemented with pattern templates as well. All of the pattern templates concentrate on arranging the first two stairtowers of a zone deck. If the zone deck has more than two stairtowers, then the third and greater stairtowers are placed randomly. The templates that are applied in the PVLN algorithms are provided in Table II.

### Table II: Stairtower Placement Templates

<table>
<thead>
<tr>
<th>Stairtower Templates implemented in ISA</th>
</tr>
</thead>
<tbody>
<tr>
<td>Starboard Fore – Port Aft</td>
</tr>
<tr>
<td>Starboard Fore – Port Fore</td>
</tr>
<tr>
<td>Starboard Aft – Port Aft</td>
</tr>
<tr>
<td>Starboard Aft – Port Fore</td>
</tr>
<tr>
<td>Transverse Centered</td>
</tr>
<tr>
<td>Longitudinal Centered</td>
</tr>
</tbody>
</table>

One important feature of the templating systems for both the passages and the stairtowers is that when the PVLN Agent fails a placement of a template a certain number of times, it can resort to random placement as a fallback. This allows the system to have a failsafe behavior.

### 5. Deck Templating

In order to implement a contiguous and practical deck plan, the passage templates need to be organized in a manner that makes it simpler to construct a connected network. To accomplish this, a combination of deck templates and stitching algorithms were employed. Deck templates are “templates of templates”. They are an organized set of passage templates for individual zone decks that are grouped together to make up a cohesive deck plan. The deck templating method chosen involves a three-part template. The first part is the type of passage template at the forward end of the
deck. If the zone deck is less than 40 m² in area, a single longitudinal passage is chosen, and the algorithm goes to the next zone deck. This is repeated until the forward template is used. Next is what type of midbody template to be used. The midbody template can be a full template, meaning the same passage template is used for all zone decks in-between the fore and the aft capping passage templates. The midbody template can also be an alternating template where the primary and secondary templates are alternated for each successive zone deck. A good example of this is alternating standard longitudinal H and standard longitudinal parallel templates. Lastly is what type of aft template caps the deck plan off in the aft. An example of a deck plan template is shown in Fig. 7. This type of template is a CAFT_HFULL_YFORE template. A full list of the available templates is shown in Table III.

![Fig. 7: Example deck template, CAFT_HFULL_YFORE](image)

Table III: Available deck templates in ISA

<table>
<thead>
<tr>
<th>Y AFT TEMPLATES</th>
<th>CAFT TEMPLATES</th>
<th>H AFT TEMPLATES</th>
</tr>
</thead>
<tbody>
<tr>
<td>YAFT_HFULL_YFORE</td>
<td>CAFT_HFULL_YFORE</td>
<td>HAFT_HFULL_YFORE</td>
</tr>
<tr>
<td>YAFT_HALT_YFORE</td>
<td>CAFT_HALT_YFORE</td>
<td>HAFT_HALT_YFORE</td>
</tr>
<tr>
<td>YAFT_HFULL_CFORE</td>
<td>CAFT_HFULL_CFORE</td>
<td>HAFT_HFULL_CFORE</td>
</tr>
<tr>
<td>YAFT_HALT_CFORE</td>
<td>CAFT_HALT_CFORE</td>
<td>HAFT_HALT_CFORE</td>
</tr>
<tr>
<td>YAFT_HFULL_HFORE</td>
<td>CAFT_HFULL_HFORE</td>
<td>HAFT_HFULL_HFORE</td>
</tr>
<tr>
<td>YAFT_HALT_HFORE</td>
<td>CAFT_HALT_HFORE</td>
<td>HAFT_HALT_HFORE</td>
</tr>
<tr>
<td>CENTERLINE TEMPLATES</td>
<td>PORT TEMPLATES</td>
<td>STARBOARD TEMPLATES</td>
</tr>
<tr>
<td>CENTERLINE_BRANCH</td>
<td>PORT_BRANCH</td>
<td>STBD_BRANCH</td>
</tr>
<tr>
<td>CENTERLINE_NOBRANCH</td>
<td>PORT_NOBRANCH</td>
<td>STBD_NOBRANCH</td>
</tr>
<tr>
<td>CENTERLINE_ALTBRANCH</td>
<td>PORT_ALTBRANCH</td>
<td>STBD_ALTBRANCH</td>
</tr>
</tbody>
</table>

6. Passage Stitching Algorithm

As denoted by the circles in Fig. 7, when the passage templates are initially laid down, they do not necessarily have contiguous connections between the zone decks. In order to achieve full connectivity, a passage stitching algorithm was implemented to complete the connections between zone decks on the same deck. This stitching only occurs for the zone decks that are on or above the damage control deck. Below damage control deck there is no need for passage stitching due to the watertight bulkheads. The algorithm stitches the zone decks by sweeping fore to aft, connecting the forward zone deck to the aft zone deck as it goes. For each pair of zone decks it performs the following algorithm:

First the algorithm determines the available connection points for the two zone decks. The available set of connection points follow a 3 x 3 grid around the zone deck (port, centerline, starboard; and fore, mid, aft), see Fig. 8. As a rule, the agent wants two connected passages, if possible, via the Port and Starboard connection points. Once the connection points have been established, then the forward zone deck attempts to connect via the port side first, the starboard side second, and the centerline third. The forward zone deck will be referred to as zone deck A, and the aft zone deck will be referred to as zone deck B. In any case, connections are done via extension of the main passage and the growth of an appendage on the passage in zone deck A if needed. The connection sequences are listed below:
Fig. 8: Sample passage stitching with connection points shown

Port Connection Sequence:
1. If the Port Aft connection on zone deck A exists and the Port Fore connection on zone deck B exists, then make that connection.
2. If the Port Aft connection exists on zone deck A and the Centerline Fore connection on Zone Deck B exists, then make that connection.
3. If the Centerline Aft connection exists on zone deck A and the Port Fore connection exists on zone deck B, then make that connection.

Starboard Connection Sequence:
1. If the Starboard Aft connection on zone deck A exists and the Starboard Fore connection on zone deck B exists, then make that connection.
2. If the Starboard Aft connection exists on zone deck A and the Centerline Fore connection on Zone Deck B exists, then make that connection.
3. If the Centerline Aft connection exists on zone deck A and the Starboard Fore connection exists on zone deck B, then make that connection.

If either the port or the starboard connection sequence was unsuccessful, then resort to centerline connections if available.

Centerline Connection Sequence:
1. If the Centerline Aft connection on zone deck A exists and the Centerline Fore connection on zone deck B exists, then make that connection.
2. If the Centerline Aft connection exists on zone deck A and the Port Fore connection on Zone Deck B exists, then make that connection.
3. If the Centerline Aft connection exists on zone deck A and the Starboard Fore connection exists on zone deck B, then make that connection.

After zone deck A has been stitched to zone deck B, then the algorithm steps one subdivision aft and the zone deck B becomes the forward zone deck of the next round of stitching. This process is repeated for the entire deck. Fig. 9 and Fig. 10 show two example seedings where the deck has been stitched together.
7. Stairtower Stitching Algorithm

The whole ship PVLN Agent also has the task of stitching multiple decks together via stairtowers. The stitching algorithm that connects the decks works similarly to the zone deck passage stitching. The algorithm starts on the damage control deck after the stairtowers for that deck have been placed. It then works its way down through the ship first. After the below damage control deck stairtowers have been stitched, then stairtowers for above the damage control deck are stitched. As it goes down or up, it first tries to co-align the stairtowers exactly with the deck below or above. If the projection of the stairtower onto the other deck is outside the envelope of the ship, it shifts the stairtower for the next deck down laterally until it is inside the hull. Then on the previous deck, it grows an appendage to cover the distance to the next stairtower in order to reserve the area as a transfer region.

8. Conclusions and Future Work

With the basic passage and deck templating algorithms completed and the passage and stairtower stitching completed the quality of the passage networks generated improved substantially. Thus, the first generation of the PVLN Agent has yielded a number of improvements over the previous round of development. First, the number of infeasible or non-practical passage configurations suggested by the optimization engine has gone down. The passage templating and the corresponding zone deck type and shape type lookup table system provides adequate control of the types of templates used. Specifically, what templates can be used when is something that in future generations need to be user editable. Second, the stairtower templating provides the user with more realistic stairtower placements from a Naval Architecture best practices standpoint. The stitching algorithms at both the passage and stairtower level allows for the generation of cohesive compartment and access networks to be generated at the seeding phase prior to agent based growth.

While the PVLN Agent has made improvements on the compartment and access generating capabilities, there is still room for improvement. The first area of improvement is in context sensitive
passage template definition. A good example of this type of situation is when fixed location spaces are present in a zone deck and the PVLN agent needs to accommodate for the no longer available regions of the zone deck, Fig. 12. In the example shown, there are two fixed space mooring areas at the aft of the zone deck. This means that remaining available area represents a “T” shaped region. Not all passage templates would work well projecting into this “T” shaped region. The algorithm needs to be able to detect the shape of available area and pick the best candidate passage template to fit the current situation.

Fig. 12: Fixed space example of available area shape / Passage template generation needs

Another area for improvement is the algorithm’s ability to have a more intelligent stairtower placement engine. With the current engine, it is possible for a stairtower to partially overwrite a passage when it grows an appendage to translate, Fig. 13. In these and other similar scenarios, the PVLN Agent needs the ability to create a new stairtower on the fly. These new algorithms also need to do a better job of stitching decks together and trying to maintain vertical access that is co-aligned as much as possible. This work as well as additional updates to the ISA Core will be pursued in the next phase of the software development.

Fig. 13: Example of stairtower overlap

Acknowledgements

The ISA development team would like to thank NSWC-Carderock and the Office of Naval Research for sponsoring this ongoing research and development effort. In particular the team would like to thank Bob Ames, Richard Van Eseltine, Keawe Van Eseltine and the rest of the LEAPS development group at NSWCCD. We would also like to thank Kelly Cooper of ONR for her ongoing support of the project. In addition the team would like to thank Mike Parsons and Eleanor Nick Kirtley for their contributions as well as their ongoing support to this project. Their advice and consultation are appreciated.
References

ASSET (2007), Advanced Surface & Submarine Evaluation Tool, Naval Surface Warfare Center Carderock Division, Version 6.0

DANIELS, A; PARSONS, M.G. (2006), An Agent-Based Approach to Space Allocation in General Arrangements, 9th IMDC, Ann Arbor

DANIELS, A; PARSONS, M.G. (2007), Development of a Hybrid Agent-Genetic Algorithm Approach to General Arrangements, COMPIT 2007, Cortona

DANIELS, A.; SINGER, D.J.; TAHMASBI, F. (2009), Intelligent Ship Arrangement (ISA) Methodology Improvements and Capability Enhancements, 10th IMDC, Trondheim


LEAPS (2006), Leading Edge Architecture for Prototyping Systems, Naval Surface Warfare Center Carderock Division, Version 3.5


NICK, E.; PARSONS, M.G. (2007), Fuzzy Optimal Arrangement of Spaces within a Zone-deck Region of a Ship, PRADS, Houston

NICK, E.; PARSONS, M.G.; NEHRLING, B. (2006), Fuzzy Optimal Allocation of Spacers to Zone-decks in General Arrangements, 9th IMDC, Ann Arbor


Hardware-in-the-Loop Simulation System for Submersible Dynamics with Visualization of Environment near the Sea Bottom for Training of Submersible Crews

Valery Mrykin, Alexander Leshchina, Central Design Bureau for Marine Engineering "Rubin", St.Petersburg/Russia, neptun@ckb-rubin.ru
Alexander Antonov, Manucher Dorri, V.A. Trapeznikov Institute of Control Sciences of the Russian Academy of Sciences, Moscow/Russia, antonov@ipu.rssi.ru

Abstract

The presentation covers the hardware-in-the-loop simulation system with the use of measurement information obtained during submersible trials for development and testing in real time of algorithms and software of submersible control systems. Measurement information from submersible onboard computers is used to update mathematical models and display the attitude of submersibles in the virtual environment. Computers of hardware-in-the-loop real-time simulation system for submersible motion parameters, installed together with the systems processing data of measurement information are used for formation of closed loop control systems interacting with data bases containing documented measurement information on submersible dynamics. For this purpose, measurements are used from navigation sensors, control elements (plane/rudder and tanks), power drives of plane/rudder, tank valves as well as control signals generated in control systems. Then, the results of hardware-in-the-loop simulation system of submersible motion parameters are transmitted to the graphical station for 3D visualization of submersible motion dynamics in the environment. The results of hardware-in-the-loop simulation of submersible motion demonstrate that this system can contribute to checking the algorithms and software of submersible control systems and reduce the scope of trials due to prediction of submersible behaviour in the wide range of conditions for conduct of trials. Proposals are given how to use the suggested hardware-in-the-loop simulation system as a training simulator to exercise tasks of submersible motion control.

1. Introduction

As per the practice adopted nowadays, training aids that can ensure proper training both for individual specialists and the crew as a whole are used for training of submersible crews. This difficult task is normally solved by many establishments: design bureaus, scientific research institutes, instrument making research and production concerns, trainer centres.

The main trend of activity in this field is development of information-engineering aids of training, special-purpose and integrated simulators and training systems. Further, all the above-mentioned facilities are supplied to a training centre for crew training. An important trend in the above-mentioned works is adoption of modern computer technologies having prospects for further development, and advancement of development processes for training facilities for submersible crew training. Computer simulators can be referred to such training facilities.

Emergence of computer technology for training of submersible crews predetermined a technical and methodical opportunity for development of computer simulators. They reproduce a dynamic model of submersible dynamics and operation of technical facilities with a high degree of adequacy.

The important constituent part of field trials of marine controlled objects is processing of measuring information obtained in trials during conduct of various manoeuvres. Further analysis of results obtained during trials of objects can confirm their characteristics as well as a possibility for stabilization of motion parameters under conditions of various disturbances and limitations imposed on the parameters by safety of navigation.

Quality of processing and display of measuring information recorded during trials considerably
determine an advisability for conduct of special trials of objects and makes it possible to implement objectives and tasks set for the designer.

Evolution of recording and processing systems of experimental data of marine controlled objects is mainly due to improvement of sources of measuring information (onboard control systems, sensors, signalisers) and effective use of means of modern computers and in the data processing systems themselves. The sources that receive information are both onboard systems and off-line sensors additionally installed to measure parameters that are not transmitted by onboard systems or not generated by them in principle.

Conduct of motion parameter measurements during field trials of objects with the help of off-line sensors or getting them from various ship's onboard systems, their recording and further processing can solve the following tasks:

- More precise determination of mathematical model of objects' motion based on motion parameters
- Estimation of forces acting on the object under various disturbances
- Check and adjustment of control systems for compensation of this force.

Within the frames for solution of these tasks based on recordings of actuators that are used for control of control systems, real time characteristics of ballast intake and removal and operation of valves are determined, functionality of various algorithms of control systems is checked.

These systems include manoeuvring control system (MCS), general ship's systems control system (GSS CS), depth measuring system (DMS), main propulsion plant control system (MPP CS), control system of technical facilities system controlling buoyancy (CS of TFS CB).

The task of processing and display of measuring information can be solved on the basis of development of hardware-in-the-loop simulation system using information from MCS and GSS CS control panels, navigation system, DMS and other systems, with the content of measuring information provided by regular systems being known in advance.

The parameters coming to MCS and GSS CS from the navigation complex as well as other control systems are subjected to special processing in these systems and together with other parameters received from onboard measuring sensors or generated by these systems can be displayed in repeaters and special video frames of MCS panel. This information can be used for visual monitoring of objects’ motion parameters, signals and commands of various systems. The information from control systems is recorded by information accumulation boards in accordance with the protocol of information exchange agreed upon between designers of the above-mentioned systems and the designer of information processing system. On the basis of recorded measurements, submersible trial results can be analysed by special computer aids of hardware-in-the-loop simulation.

To obtain the missing measuring information in the real-time scale it is restored on the basis of available measured parameters. This information is designed to determine the force action on the submersible. The totality of restored parameters forms the system of virtual sensors that checks the sensor readings comparing them with the values predicted in real time. Then, the decision is taken on validity of the readings. It is important to check readings of the sensors prior to use them in closed loops of submersible automatic control systems. This task is solved using the hardware-in-the-loop simulation system on the basis of preliminary comparison of simulation and actual processes obtained during trials.

The values of sensor readings under monitoring either accepted as correct ones or marked as erroneous. Assessments of sensor valid readings are transmitted to the simulation system. As per our experience, it was found that typical failures of sensors can be detected and corrected. Suggested approach gave positive results during such fault diagnosis.
Testing and integration of control algorithms during field sea trials may take a lot of time. Trials are expensive and rather dangerous operations. To reduce expenses and ensure high reliability of these operations, there comes to light a necessity to use simulation during development of algorithms beginning with a check of individual components and ending with a determination of expected behaviour in the wide range of application conditions.

Simulating computers send a simulated vector of submersible status in real time to a graphical station through a local network for visualization of submersible motion in the environment with the 3D technology using the RDS (calculation of dynamic systems) software developed by the V.A. Trapeznikov Institute of Control Sciences. For simulation of submersible motion parameters, workstations use measuring information obtained during submersible trials and supplemented with the help of virtual sensors.

Section 2 describes the status of activities with the development of virtual dynamic systems. Section 3 describes the proposed simulation structure of submersible dynamics and visualization of the environment as per the developed methodology. This structure is used during processing of measuring information in subsystems of processing, visualization and simulation in real time. Section 3 presents some simulation results.

2. Analysis of research activities and achieved results

Information technologies, in particular virtual dynamic systems are widely used all over the world including marine industry.

Virtual dynamic systems are successfully used for the following:

- Development of trainers and prototypes of objects (aircraft, submersibles, ships, etc.)
- Creation of conditions increasing efficiency of operator’s work (especially this is important for submersibles where the external situation can be approximately simulated)
- Proof of automatic control algorithms of objects being under construction anew or under modernization.

The following can be referred to the software products having direct or close relation to the virtual dynamic systems in the field of formation and research of control systems:

- Software packages for simulation and synthesis of control systems
- SCADA (Supervisory Control and Data Acquisition) systems
- Mathematical packages.
- Computer-aided simulation systems for manufacturing processes.

The first group includes: PC-MATLAB (developed by MathWorks Inc.); LabView (developed by National Instruments); ISEE Simnon. Russian systems from this group include: Dynamic System Calculation (DSC) by the V.A. Trapeznikov Institute of Control Sciences of the Russian Academy of Sciences and software package by the Bauman Moscow State Technical University.

The second group includes SCADA packages: AIMAX (developed by Autenco), LOGOVIEW (developed by LogoSystem), Real Flex (developed by B J Software, USA), FIX DMACS (developed by Itellution, Inc.), FACTORY LINK 1V (developed by U.S. Data), IN Touch (developed by Wonderware), GENESIS and MOSCAD (developed by Motorola).

Russian systems from this group include: VTS (developed by InSAT) and TRACE MODE (developed by Ad Astra). The last one is rather widely known.

The third group includes: Derive 2.6 (developed by Soft Warehouse); MathCAD Plus 6.0 (developed by MathSoft); Maple V R3 (developed by Waterloo Maple Software); Mathematica (developed by Wolfram Research); MathLab (developed by MathWorks); Axiom (developed by Numerical Algorithms Group).

The fourth group includes various systems developed by large research centres mainly as in-house systems that are not designed for commercial use.
At present, software products, both foreign and Russian ones, capable to solve the task of creating virtual dynamic systems in full scope for control of dynamic objects are not known, which causes necessity in further works oriented to the development of principles of virtual systems formation, development of theoretical basis for simulation structural methods necessary for solution of both technical and situational tasks. It is also necessary to improve mathematical methods of research of sophisticated control systems and processes and develop mechanisms of interaction of virtual systems with data bases and modules describing the objects under research and helping reconstruct easily the tasks to be solved.

3. Software of hardware-in-the-loop simulation system for submersible motion control

3.1 Design of simulating systems

The hardware-in-the-loop simulation system includes a processing subsystem for measuring information, a subsystem of submersible motion simulation in real-time scale, and a subsystem for visualization of submersible dynamics and environment.

Requirements to the hardware-in-the-loop simulation applicable to submersibles include the following:

• Full dynamic models of submersibles
• Models of navigation parameter sensors
• Models of control elements
• Visualization of coordinated control of submersibles in 3D underwater environment.

![Fig.1: Structure of hardware-in-the-loop simulation system for submersible dynamics](image-url)
The hardware-in-the-loop simulation system forms a state vector of submersible and transmits it to the graphical station for visualization of submersible dynamics and environment by display technology based on the software of virtual dynamic systems. General structure of hardware-in-the-loop simulation system of submersible dynamics is shown in Fig. 1.

In this system, the simulating server computer implements a high-resolution graphics for 3D visualization of submersible motion and virtual ocean environment. Users are closed loops of hardware and software modules of simulation united with real measuring information data bases and transmitting updated parameters to the subsystem for simulation in real time as shown in Fig. 2.

![Diagram of data transmission between systems](image)

**Fig. 2: Diagram of data transmission between systems**

### 3.2 Subsystem for simulation of submersible dynamics

Subsystem for simulation of submersible dynamics, manoeuvrability and control included in the hardware-in-the-loop simulation system is a toolkit for simulation of submersible dynamics and refinement of hydrodynamic models of submersibles considering the measuring data obtained during trials. The refinement is based on the study of submersible motion parameters on the basis of measuring data base which are corrected by addition of components that take into account a scale effect, which ensures the prediction of manoeuvring characteristics of submersibles.

During generation of equations for 3D motion of submersibles, values of coefficients of hydrodynamic forces and moments, as per the practice adopted today, are obtained on the basis of experiments with submersible models in the laboratory environment.

The essence of the suggested method for refinement of mathematical models of submersible motion is in the use of the identification method to determine more precisely inaccurate values of sought-for...
characteristics (these values were specified in the beginning of iteration process) during numerical integration of the system of differential equations describing submersible motion. In this method, unsteady processes, normally carried out during acceptance trials, are considered, which makes it possible to avoid the conduct of special manoeuvres. The criterion for sought-for mathematical models is a minimum value of divergence between the calculated motion parameters and those measured during trials under similar control. These measurements are processed using the least squares method, where the value of divergence between the measured and calculated parameters is minimized.

Parametric model of submersible motion is expressed in the form of generalized model of state vector. To determine hydrodynamic coefficients, the functional is minimized by the method of nonlinear programming (combined method of Gauss-Newton and quickest descent). During determination of hydrodynamic coefficients, initial values of the vector of sought-for coefficients are added to corrections ensuring better coincidence of actual and simulated motion parameters. As a result, the numerical sequence of values of hydrodynamic coefficients vector is determined. This sequence converges to the values determined in terms of minimum functional of mismatch.

Expressions of hydrodynamic forces and parameters included in the equations of 3D motion of submersible contain a large number of expansion coefficients, which does not make their simultaneous determination possible based on the data of actual trials. To facilitate the solution of task and taking into account the nature of most operating manoeuvres, particular kinds of motion are considered, namely: motion in the vertical planes and spatial motion of submersible at circulation turn. Common numerical method for determination of positional and rotational hydrodynamic coefficients corresponding to the motion only in the vertical plane and rotational coefficients corresponding to manoeuvring modes at circulation was developed.

To check the applicability of this approach, the predicted parameters were obtained using the simulation system with similar manoeuvres of submersible. After determination of common conditions of simulation and conduct of manoeuvres, the simulation results were compared with the experimental data obtained during trials of real submersible, which is shown in Fig. 3.

![Fig. 3. Vertical manoeuvre of submersible prior to refinement of coefficient. Left: Depth, Right: trim; Experimental data: Bold line (red), Simulation data: Thin line (blue).](image)

Results of comparison showed that some manoeuvre parameters corrected with the help of the approach above demonstrated agreement with actual results, while others considerably depend on errors in conduct of simulation. Thus, a preliminary conclusion can be drawn that the models used can be useful for determination of residual forces and moments and these solutions are reliable for some types of manoeuvres that were conducted (vertical manoeuvres) including manoeuvres with considerable nonlinear characteristics.
Fig. 4: Vertical manoeuvre of submersible after refinement of coefficient. Left: depth. Right: trim; Experimental data: Bold line (red), Simulation data: Thin line (blue).

Fig. 5: Vertical manoeuvre of submersible prior to refinement of coefficient. Left: Forward planes, Right: Aft planes; Experimental data: Bold line (red), Simulation data: Thin line (blue).

Fig. 6: Vertical manoeuvre submersible after refinement of coefficient. Left: Forward planes, Right: Aft planes; Experimental data: Bold line (red), Simulation data: Thin line (blue).

Additional advantage of the approach using identified models is that the simulation model can be developed immediately after obtaining the experimental data, directly on the basis of suitable experimental data and these simulation results can be further used directly during full-scale simulation of control system. The above-stated determines the requirements for gathering the data for the development of full-scale model as the data necessary in the scale of physical model are available for all underwater objects.

3.3. Subsystem for visualization of submersible dynamics and environment

Functioning as a simulating server, the graphic station visualizes the 3D motion of submersible by the display technology that uses the state vector received from models in real time and transmitted over
the local network. The graphic station also receives signals from the submersible control systems.

Simulating subsystem is developed for 3D visualization of submersible behaviour that involves plenty of information display technologies such as object compensation, complicated texture mapping configurations, dynamic topographic generations and control, switch-over of points of view and so on and requires a large amount of computing. OpenGL software is used to satisfy the requirements of the subsystem and for better implementation in real time.

In the subsystems simulating 3D motion of submersible, forms are developed using OpenGL. With the help of these forms it is possible to select geometry, colours and other characteristics of submersible 3D motion. They enable to change and specify data on-line and visually display a result on the monitor screen.

The software package is capable of simulating systems represented as a set of interconnected modules. Each module may contain a calculation program determining interaction with other modules of the system and a response to user’s action. Groups of functionally connected modules can be united into subsystems that may have their own program. Figs. 7 to 11 give examples of environment visualization and devices displaying submersible attitude.

![Fig. 7: Coloured visualization of map and 3D surface corresponding to it](image1)

![Fig.8: Heel and trim indicators](image2)
Fig. 9: General view of devices displaying the shape of bottom

Fig. 10: View of trim indicator

Fig. 11. Sea bottom area
3.4. Manoeuvring control system and method of coordinated control

Manoeuvring control system (MCS) is a core of control system of each submersible. It receives measurements of motion parameters from own navigation system and navigation parameters related to presence with respect to surface or bottom and implements coordinated control in distributed mode. In MCS, speed is changed by setting the propeller speed and control/monitoring of diving depth, trim and course is implemented by aft and forward planes.

MCS as a coordinated control system of submersible receives specified parameters from the system operator prior to commencement of task execution including route points relative to the bottom, depth and trim commands, commands about speed, etc. During execution of task, MCS uses depth and trim control in the vertical plane and route motion control in the horizontal plane.

Method of buoyancy and plane/rudder coordinated control is in the selection of modes in plane/rudder and buoyancy control in such a way that plane/rudder should not counteract the buoyancy, i.e. buoyancy control should not be excessive and should be stopped in due time.

During execution of manoeuvres by the submersible near the sea bottom, normally, accurate balancing is required as volume and weight change considerably and ballast tanks are drained.

Evaluation of behaviour by submersible trim to ensure safety while moving neat the sea bottom is an important task to be solved at the stage of proving submersible control during operation. Prior to that, relevant instructions are to be written.

Metacentric stability of submersible plays an important role during its motion under water and depends on such design characteristics as metacentric height, area of flood holes of ballast tanks, flow rate and duration of compressed air supply to them as well as initial depth and set of drained or flooded tanks. Normally, ballast tanks are united in groups of several tanks: forward, aft and central groups. In the process of submersible motion near the bottom the operator monitors trim and depth variation in such a way the motion should be without considerable change in trim.

Important factor of safe motion of submersible near the sea bottom is a selection of optimal control algorithms for buoyancy and plane/rudder. Submersible’s manoeuvres near the sea bottom are much diversified in the manner of execution. This is related to the possibility of motion at various diving depths under different combinations of ballast tank blowing. The issue of operator’s selection of optimal control algorithms for buoyancy and plane/rudder is the decisive one when solving the task of submersible safe motion near the sea bottom. At the same time, information on the submersible position displayed in the required scope and format convenient for perception is a necessary condition for the submersible manoeuvring control operator in the process of control for correct execution of the manoeuvre. The operator should have an idea of the arrangement of the submersible’s ends to ensure safety during execution of this manoeuvre.

The manoeuvring control panel operator has the opportunity to use additional information extremely important for control on the basis of virtual dynamic systems, giving 3D picture of submersible moving in water relative to bottom with indication of motion parameters values on scale instruments. The 3D picture of submersible changes colour background when approaching the sea bottom (transparent background is becoming darker when approaching the sea bottom).

3.5 Real-time models

The basic structure of real time software running in the closed loop of simulation system for submersible consists of the closed loop between the simulating system and data bases of recorded information that includes commands and signals documented during submersible trials. The simulating system generates values of submersible motion parameters using commands and signals of control systems recorded during trials instead of simulated ones and comparing the parameters with
sensor readings forwarded to the simulating system. To provide the operating environment for each control computer, the real time simulation system should carry out the following functions.

- Selection of control commands of technical facilities system controlling buoyancy and tilting of plane/rudder of manoeuvring control system, transmitted by recording computers to data bases for further simulation of submersible dynamics and plane/rudder control.
- Simulation of submersible dynamics.
- Generation of unprocessed output signals of navigation system sensors in a prescribed output format.
- Provision of user-friendly human-machine interface implementing setting of simulation parameters and activation of models.

All real time simulation systems use an industrial controller installed together with data accumulation board arranged for collection and output of digital-analogue signals. Software for simulation in each simulating system consists of model of submersible dynamics, models of propulsor, plane/rudder and navigation system. The hardware-in-the-loop simulation system is used together with the RTX real time software.

4. Simulation results

Using this software for simulation of submersible dynamics in the closed loop we can check the coordinated control of buoyancy and plane/rudder. Simulation process can be described by the block diagram in Fig. 13.

![Block diagram for solution of task of coordinated control](Image)

Coordinated control of submersible buoyancy and plane/rudder as per the data of measuring information on commands and signals of automatic control systems of a real submersible works well together with the simulated state vector of submersible and virtual ocean environment and can be fully tested with any boundary conditions. The closed-loop simulation systems checked by this software, the developed software and algorithms can be used for the designated purpose in the real submersible and are suitable for sea trials with a reduced risk level of accidents.

5. Conclusions

Use of the proposed hardware-in-the-loop simulation system will enable to update and check control algorithms of control systems directly on board the submersible, gather additional information on the operation of actuators, force action of the submersible, and refine the mathematical model of dynamics.

Use of computer technology, local area networks, large data bases and imitating simulation of various...
processes can create virtual dynamic model of activities of operators in CIC and in the crew as a whole. Adequate display by trainers of real control processes of submersible motion gains in importance, and adequate display is possible only during conduct of comparative analysis between full-scale and simulated motion parameters, comparison of simulation results and real processes taking place during submersible trials. This allows the adequacy of mathematical models of submersible dynamics and operation of its technical facilities to be evaluated.

References

MRYKIN, V.; TOLKACHEV, P.; LOMOV, V.; DORRI, M. (2009), Desktop research and training stand for assessment of submersible motion control algorithms and skill exercising in technical facilities control, 6th Int. Symp. Underwater Technology (UT 2009), Wuxi

MRYKIN, V.; LOMOV, V.; KURNOSOV, S.; DORRI, M. (2010), Training complex for training submersible motion control skills on the basis of virtual dynamic systems, COMPIT, Gubbio

MRYKIN, V.; LOMOV, V.; DORRI, M. (2010), Promising submersible motion control console with elements of environment visualization on the basis of virtual dynamic systems applicable to submersible surfacing manoeuvre, 19th Symp. Theory and Practice of Shipbuilding, Lumbarda
A Comparison of Strategies for the Optimization of a Ship’s Aft Body

Auke van der Ploeg, MARIN, Wageningen/The Netherlands, a.v.d.ploeg@marin.nl

Abstract

This paper describes a procedure to optimize ship hull forms based on double body viscous flow computations with a RANS-solver. A flexible and effective definition of parametric hull form variations is used, based on interpolation between basis hull forms. One of the object functions is the estimated required power. In this paper we focus on how to improve this estimate by using the B-series of propellers. Results of genetic algorithms and systematic variations together with scale effects in the trends are discussed. In addition, we demonstrate how the techniques discussed in this paper may be used to design a model scale hull with a wake field that strongly resembles the wake of a given ship at full scale.

1. Introduction

One of the most effective ways to reduce ship fuel consumption (and thereby emissions), is by improving the hull form design. Nowadays, computational techniques play a central role in this, permitting extensive design studies prior to any model testing. The usual procedure is to analyze the results of a computation, and derive recommendations on hull form changes from that, based on an understanding of the flow physics and its relation with the hull form. Instead, automatic optimization procedures may be used based on a series of CFD computations. There is a substantial development on this subject, and the first practical applications are making their way into practical ship design; mainly using potential-flow solvers, but increasingly also RANS codes. At MARIN, the RANS solver PARNASSOS is combined with the GMS-Merge tool, which varies hull forms by a special interpolation between several pre-defined hull form variants. The first application of this procedure was for a workshop organized as part of the VIRTUE project, Marzi (2008), in which all participants optimized the same tanker aftbody with respect to resistance and wake field quality, van der Ploeg and Hoekstra (2009). Both the computed 3% reduction in resistance and the computed change in wake field (Fig. 1) were confirmed by measurements (Fig. 2).

Fig. 1: Computed wake fields
Optimized and original VIRTUE tanker

Fig. 2: Measured wake fields
Optimized (left) and original (right)

One of the object functions was the quality of the wake field at model scale. However, since scale effects in the wake are important, the RANS computations for all variants have to be performed at full-scale Reynolds numbers. In addition, it is not obvious that minimizing resistance improves fuel
44

efficiency: a decrease in resistance is usually accompanied by a relatively strong decrease of the nominal wake fraction, and the question is then how much this influences the power required for a given speed. Therefore, in this paper, we evaluate alternative object functions, one of which is a power estimate. This requires RANS computations, both with and without actuator disk for each hull form variant, to estimate the thrust deduction and propeller efficiency. In van der Ploeg and Raven (2010), a relatively simple estimate of the required propeller efficiency was used. In this paper, we will focus on how to improve this estimate by analyzing the efficiencies obtained from the B-series of propellers described in Kuiper (1992). Some results for the VIRTUE tanker are presented.

Section 6 shows that the strategy described in this paper can be used to inversely design a scale model hull that generates a wake field that closely resembles a ship scale target wake field. Section 7 demonstrates how the techniques described in this paper can be combined with a genetic algorithm.

2. Hull form variations

For the parametric deformations of the geometry we use the GMS-Merge tool, Hoekstra and Raven (2003), which makes a special interpolation between several basis hull forms that differ from the original hull and that have been created in the CAD system GMS. It is important here that the variations are defined specifically for the ship that has to be optimized, based on a first CFD computation for the original vessel. Thus, specific and effective variations can be selected. Although design experience may be used in the choice of the basis hull forms, successive cycles will provide guidance, as they clearly display the dependence of the flow on the hull form and may indicate possibilities for further improvement. Moreover, if the basis hull forms satisfy the geometric constraints, often all variations in the design space do so, although this is not guaranteed.

Fig. 3: Illustration of GMS, tool for hull form variation

Next, we perform systematic variations of the parameters, using GMS-Merge to generate combinations of the basis hull forms. This is illustrated in Fig. 3, for the case in which we start with an original hull and five basis hull forms. By pulling none of the five slides (top-left), the original hull form is obtained. By pulling one of the slides down completely (top-right and bottom-left), the basis hull forms are obtained. One can pull several slides simultaneously to make combinations of these basis hull forms, as is illustrated in the bottom-right picture. The parameters of the hull forms to be
evaluated are the weighting percentages of the respective basis hull forms. In this example, we have a five-dimensional parameter space and if we use three steps to go from one hull form to the next, we have a total of $4^5 = 1024$ variants. The next step is to perform RANS computations for each variant, using a completely automatic way to generate the grid for each hull form. After an evaluation of all variations and inspection of trends, one may refine in particular parameter ranges, add new basis hull forms and delete others, etc. The process may thus proceed in a number of consecutive steps, providing large freedom and clear insights in possibilities for further improvement.

3. RANS computations

3.1 The RANS code

The RANS code we use is PARNASSOS, a code developed and frequently applied by MARIN and IST, Hoekstra (1999). It is dedicated to the prediction of the steady turbulent flow around ship hulls and solves the discretized Reynolds-Averaged Navier-Stokes equations for steady incompressible flow. Various eddy-viscosity turbulence models are available. For the applications described in this paper we use the standard k-ω SST turbulence model without corrections. Structured multi-block body-fitted grids of H-O topology are used. A finite-difference discretization scheme is used, with second and third-order schemes for various terms. The solution technique for the resulting linear system of equations is very efficient with respect to both CPU-time and memory usage, van der Ploeg et al. (2000), which makes it very well-suited for performing systematic variations or combination with an optimization strategy.

3.2 Automatic grid generation

Grids need to be generated around each variant by a fully automatic grid generation procedure. To minimize the effect of discretization errors on the computed trends, the grids have to be as closely corresponding to each other as possible. An automatic procedure for deformation of the wall grid was developed that allows strong modifications of the hull form. As a first step in the construction of the grid for a hull form variant, the wall grid for the original hull form is projected on the variant. Next, the 3D-grid is obtained using the usual grid generation techniques.

3.3 Parallel computation

When performing the systematic variations, the RANS computations for all variants are distributed over a large number of idle desktop PCs at MARIN overnight, using the Condor tools, Thain et al. (2005). The limited memory and CPU requirements make this relatively straightforward as each RANS computation runs on just one PC. Each PC in the cluster runs a daemon that watches user I/O and CPU load. When a PC has been idle for two hours, a job from the batch queue is assigned to the PC and will run until the daemon detects a keystroke, mouse motion, or high non-Condor CPU usage. At that point, the job will be removed from the PC and placed back in the batch queue. In this way, several hundreds of RANS computations, each performed on a mesh containing 1.7 million cells, can be made in only a few hours. This kind of parallelization can be combined with genetic algorithms, since all computations within one generation can be performed simultaneously.

4. Object functions

As mentioned before, it is not obvious that minimizing the resistance improves the fuel efficiency. It was found that a decrease of the resistance is usually accompanied by a relatively strong decrease of the nominal wake fraction and the question is how much this influences the required power to sail at a given speed. Therefore, in this paper we evaluate alternative object functions, of which one is an estimate of the power, $P_D$, delivered to the propulsor:

$$P_D = \frac{R_r \cdot (1 - w) \cdot V_t}{1 - \eta} \cdot \frac{V_t}{\eta_r \cdot \eta_0}$$ (1)
$R_T$ is the towing resistance, $w$ the estimated effective wake fraction, $V_t$ the ship’s speed, $t$ the thrust deduction fraction, $\eta_0$ the propeller efficiency in open water, and $\eta_R$ the relative rotative efficiency. The latter is approximated by 1, while for $\eta_0$ we use two different methods:

- **Method I.** $\eta_0$ is an approximation based on a simple function of the ideal efficiency $\eta_i$ that can be obtained by an actuator disk model in open water:
  \[
  \eta_i = \frac{2}{1 + \sqrt{1 + C_T}}
  \]  
  in which the $C_T$ is the thrust coefficient, van der Ploeg and Raven (2010).

- **Method II.** $\eta_0$ is obtained by analyzing the efficiencies obtained by the B-series of propellers described in Kuiper (1992). For each hull form, the computed nominal resistance $R_T$, and the hull efficiency $\eta_H=(1-t)/(1-w)$ are required to perform this evaluation.

The effective wake fraction is estimated from the nominal wake using the NOMEFF tool. This computes the induction velocities and interaction velocities by a force-field method that solves simplified Euler equations for a given actuator disk. The interaction velocities are then added to the nominal wake field found from the first RANS computation performed for each hull form variant.

To compute the thrust deduction, we also perform a second RANS computation including an actuator disk model with an imposed thrust $T_0$. This thrust is in the neighborhood of the thrust $T$ required for self-propulsion, such that we may assume a linear behavior of the force on the hull as a function of the imposed thrust. The thrust deduction fraction can then be computed from $t=(R_0-R_T)/T_0$, in which $R_0$ is the resistance force resulting from the second RANS computation. The self-propulsion thrust $T$ is equal to $R_T/(1-t)$.

In case of danger of erosive cavitation, one would like to prevent strong variations of the wake in circumferential direction, especially in the top half of the propeller plane. In the sequel of this paper, we use as the wake object function (WOF) the L2-norm of the derivative of the arriving water velocity $V_A$ in circumferential direction averaged over a number of radii.

\[
WOF = \frac{2}{\pi(R_{prop}^2 - R_{hub}^2)V_A} \int_0^R \int_0^{\pi/2} (\partial V_A/\partial \theta)^2 d\theta r dr
\]  

\[
(3)
\]

5. Results for the VIRTUE tanker

5.1 Test case

We consider the same test case used in the VIRTUE project, Marzi (2008): the VIRTUE tanker. Main dimensions are ($L= 320$ m, $T= 21$ m, $B = 60$ m), the ship’s speed is (15.55 kn), displacement and propeller position are kept unchanged, and only the aftbody was allowed to be changed. Fig. 4 shows the body plan of the original vessel, together with the dashed line which indicates the geometric constraints which had to be met in order to leave sufficient room for the engine. Free surface effects were not taken into account: all computations were performed for double body flow. The Reynolds numbers at model scale and full scale were, $6.5 \cdot 10^7$ and $2.25 \cdot 10^9$, respectively. At the VIRTUE workshop, the optimization was performed for model scale and the first object function was to minimize nominal resistance. Good agreement with measurements was obtained, Fig. 1.
5.2 Grid dependence study

To limit the large amount of computational work in an optimization, one may be tempted to work on coarse grids and impose less strict convergence criteria. However, the computed trends can well be influenced by numerical disturbances like grid dependence or insufficient convergence. Several preliminary studies of this have been performed, van der Ploeg and Raven (2010). In this paper, we only show a result from a grid dependence study of the computed Pareto front, using as objective functions the Eqs. (1) and (3), in which the estimate of the open water efficiency was obtained by the first method described above. The coarse grids used in this study have 1.7 million cells, the fine grids 3.4 million. The fine grids were obtained after refinement in all directions.

The results are summarized in Fig. 5, which shows the computed Pareto front obtained on both grid densities. The red squares indicate computations on a coarse grid, and black squares indicate fine grid computations. Blue lines connect computations for the same hull form on different grid densities. Therefore, the length of these lines gives an indication of the grid dependence. There are several groups of almost parallel blue lines of approximately the same length, which indicates that there is not much scatter in the results. The predicted decrease in $P_D$, as computed on the coarse grid, can be up to 0.4% lower on the fine grid. Therefore, it is dangerous to draw conclusions from trends computed on the coarse grid, as long as the relative change in $P_D$ is below 0.4%. However, for most hull forms...
the difference in predicted decrease in $P_D$ as computed on the coarse grid and the fine grid is significantly smaller.

### 5.3 Scale effects

One of our object functions is the variation in circumferential direction of the nominal wake field. However, as indicated by Fig. 6, the scale effects in the computed wake field can be quite significant. Moreover, since the first object function, Eq. (1), depends on the estimated effective wake fraction, it is likely that the trends in this object functions are subject to significant scale effects as well. Therefore, in the remainder of this section, we only show results of full-scale computations.

![Fig. 6 Scale effects wake field VIRTUE tanker](image)

Table I: Results for the VIRTUE tanker (full scale)

<table>
<thead>
<tr>
<th></th>
<th>Original</th>
<th>Optimized</th>
</tr>
</thead>
<tbody>
<tr>
<td>$1000 C_{vp}$</td>
<td>0.269</td>
<td>0.226 (-16%)</td>
</tr>
<tr>
<td>$1000 C_{vf}$</td>
<td>1.469</td>
<td>1.465 (-0.3%)</td>
</tr>
<tr>
<td>$1000 CD_T$</td>
<td>1.738</td>
<td>1.691 (-2.7%)</td>
</tr>
<tr>
<td>Thrust deduction coeff. $t$</td>
<td>0.21</td>
<td>0.200 (-4.7%)</td>
</tr>
<tr>
<td>Wake fraction $w$</td>
<td>0.255</td>
<td>0.224 (-12%)</td>
</tr>
<tr>
<td>$\eta_H$</td>
<td>1.061</td>
<td>1.031 (-2.8%)</td>
</tr>
<tr>
<td>$\eta_o$ (method I)</td>
<td>0.600</td>
<td>0.617 (+2.8%)</td>
</tr>
<tr>
<td>$\eta_o$ (method II)</td>
<td>0.596</td>
<td>0.610 (+2.3%)</td>
</tr>
<tr>
<td>$2000 P_{D}/(V_f^3 \rho S_w)$ (method I)</td>
<td>2.729</td>
<td>2.657 (-2.6%)</td>
</tr>
<tr>
<td>$2000 P_{D}/(V_f^3 \rho S_w)$ (method II)</td>
<td>2.750</td>
<td>2.689 (-2.2%)</td>
</tr>
<tr>
<td>Advance coeff. $J$</td>
<td>0.535</td>
<td>0.557 (+4.1%)</td>
</tr>
</tbody>
</table>

Table I shows the values of several quantities for both the optimized and the original hull form, in which we used method II to compute the open water efficiency. We see a decrease of the viscous resistance coefficient $CD_T$ of about 2.7%, which almost completely results from the relatively strong relative decrease (16%) of the pressure part of the resistance.
The open-water diagrams for both hull forms are shown in Figure 7. The values that were used in the power estimate are indicated by the green crosses in this figure.

5.4 Computed trends

As explained in the previous section, we have two methods to estimate $\eta_o$. If method II is applied, the following input is used in the evaluation of the Wageningen propeller B-series: a propeller tip speed of 35 m/s, a number of blades of 5, and a submersion of the propeller shaft of 17.3 m. First, we study the influence of the choice of method to estimate the open water efficiency. The question is whether the method to compute $\eta_o$ influences the computed trends in the first object function. For many variations of the VIRTUE tanker, we have computed $\eta_o$ with both methods and show the relative differences in $\eta_o$ (as percentages) in Fig. 8 as a function of the decrease in $P_D$ compared to the original hull form. Each square in the figure corresponds to the result of two full-scale RANS computations, with zero thrust and imposed thrust $T_0$, as explained in the previous section. From this figure, it appears that the open water efficiency as estimated by method I is slightly higher than as computed by method II.

Fig. 9 shows how much this influences the computed trends in the first object function. On the axis is the increase of the estimate of the first object function as computed by method I (horizontal axis) and II (vertical axis). Hence, the result of the original tanker is found at the origin. It appears that the correlation between the increase of both estimates for $P_D$ is very good, although with method II the
maximum decrease in $P_D$ that can be obtained is slightly smaller than with method I. In the remainder of this section, we will use method II.

The left-most picture in Figure 10 shows the correlation between the achieved reduction of $R_T$ and of $P_D$ for all variants. Clearly $R_T$ and $P_D$ are closely correlated. However, the decrease that can be obtained in $R_T$ is significantly smaller than the decrease obtained in $P_D$. From Eq. (1) it follows that this effect must be explained by the fact that for the hull forms with the lowest $P_D$ the product of the efficiency in open water and the hull efficiency decreases compared to this product obtained for the original vessel. This is confirmed by the right-most picture in Fig. 10. From this figure it appears also that there is a small number of hull forms for which the decrease of this product of efficiencies is relatively small, resulting in the strongest decrease of $P_D$.

![Fig. 10: Increase of $P_D$ as a function of $R_T$ (left) and as a function of the efficiency (right)](image)

The right-most picture in Fig. 11 shows that the hull efficiency decreases significantly compared to that of the original vessel. However, the lowest $P_D$ is not obtained in combination with the strongest decrease in the hull efficiency: There are a few hull forms that have a relatively small decrease of the hull efficiency and show the strongest decrease in $P_D$. The left-most picture in Fig. 11 shows that the decrease in hull efficiency is only partly compensated by an increase of the open-water efficiency.

![Fig. 11: Increase of $P_D$ as a function of the increase of the open-water efficiency (left) and as a function of the increase of the hull efficiency (right)](image)
The picture on the left in Fig. 12 shows that the lowest $P_D$ is obtained in combination with a significant increase in the advance coefficient, although the lowest $P_D$ does not correspond to the highest advance coefficient; there are several hull forms with a higher value for $P_D$ which have a higher advance coefficient. The picture on the right shows that the lowest $P_D$ does correspond with the lowest thrust deduction. A comparison of the body plans of the optimized and original tanker is shown in Fig. 13. The more slender aft body results in a reduced wake peak as is shown in Fig. 14.

Fig. 12: Increase of $P_D$ as a function of the increase of the advance coefficient (left) and as a function of the increase of the thrust deduction (right).

Fig. 13: Hull form with lowest $P_D$ (red) compared with the original form (yellow)

Fig. 14: Computed full-scale wake fields

6. Smart dummy

Although the extrapolation of measurements from model basins to full scale is successful for many applications, this is not always the case for predictions of propeller-induced vibratory hull excitation pressures. When model testing large single-screw container ships the measured hull pressure fluctuations sometimes turn out to be too high compared to full scale. One likely cause of such over predictions is the difference between the wake fields of model and ship, resulting in a difference in cavitation behavior and propeller-induced hull excitation forces. To overcome this problem, we investigated the possibility to design for a given ship a model in such a way that it generates a model-scale wake field that resembles the wake field of the original ship (at full scale), especially in the top of the top sector of the propeller plane. The name ‘Smart Dummy’ for such a model originates from
the dummies used in cavitation tunnels in which large geometrically similar models would not fit. We used the technique described in Section 3 to determine the shape of a dummy that generates the full scale wake field of the original vessel, a six-bladed container ship, Fig. 15. Measurements of the nominal wake for the original, geometrically similar model were performed. Fig. 16 shows that there are indeed significant differences between the measured wake field of the model and the computed wake field at full scale, especially in the top of the propeller plane, which might cause the above-mentioned over predictions of hull excitation pressures.

Fig. 15: Original model of the container ship, geometrically similar

![Fig. 15: Original model of the container ship, geometrically similar](image)

Fig. 16: Measured nominal wake of original model (left) and computed nominal wake of original vessel at full scale (right)

![Fig. 16: Measured nominal wake of original model (left) and computed nominal wake of original vessel at full scale (right)](image)

Fig. 17: View from behind the Smart Dummy (left), and the original hull (right)

![Fig. 17: View from behind the Smart Dummy (left), and the original hull (right)](image)
In the design of the Smart Dummy we imposed several constraints: above the stern the dummy must be geometrically similar to the original vessel, since the pressure sensors have to be placed at positions corresponding with full scale, in order to have the diffraction constants identical. In addition, there are some constraints in order to guarantee sufficient space for the propeller powering system, and preferably the length, trim and sinkage must be retained to prevent significant changes in the wave system.

We started with varying the length and width of the model, but it appeared to be possible to design a model of which the wake field resembles the one of the ship, without changing the length and width. Only the aft part was modified in such a way that the smart dummy could be milled with the original geometrically similar scale model, as input, Fig. 17.

Measurements of the nominal wake for the Smart Dummy were performed as well. The left-most picture in Fig. 18 compares the measured nominal wakes of the Smart Dummy and the geometrically similar model (indicated by geosim). There are significant differences, especially in the top of the propeller plane: the wake peak of the geosim model is much deeper than that of the smart dummy. The right-most picture in Fig. 18 shows a similar comparison of computed nominal wake fields. The computed trends are very similar to the measured trends.

Fig. 18: Measured (left) and computed (right) trends in the nominal wake fields

Fig. 19: Measured nominal wake of smart dummy (left) and computed nominal wake of original vessel at full scale (right)
As mentioned before, the aim was to construct the hull form of the smart dummy in such a way that its wake field resembles the full-scale wake of the original vessel. Therefore, Fig. 19 compares the measured nominal wake field of the smart dummy and the computed nominal wake field (full-scale) of the original vessel. Especially in the top of the propeller plane the wake fields are very similar. In Schuiling et al. (2011) it is shown that with the Smart Dummy concept blade rate order hull pressure fluctuations could be reduced to values close to those found on full scale.

7. Genetic algorithm

Instead of performing systematic variations as explained in Section 2, one may combine the above-mentioned parallelization over a PC network with a genetic algorithm as well. All computations within one generation can be computed in parallel. A genetic algorithm has the advantage that the total number of RANS-computations can decrease significantly. This is illustrated in Fig. 20, showing the result of applying a genetic algorithm to the optimization of a bulk carrier in a 5-dimensional design space using Eqs. (1) and (3) as object functions. Each generation consists of only 20 individuals. A global idea of a Pareto front could be obtained in only 200 function evaluations. A drawback of this strategy is however that only the computations of the same generation can be performed in parallel. If the PC-network is large enough, and the RANS-solver is efficient enough to make hundreds of RANS computation in one night, a quicker way to obtain the Pareto front is to use systematic variations. However, when the dimension of the search space increases, the number of RANS computations of course increases dramatically, and in that case a genetic algorithm will be more efficient. A combination is also possible: one could perform some steps of a genetic algorithm in order to determine which parameters are important, and after that perform a systematic variation of only those parameters.

![Fig 20: Example of a Pareto front corresponding to the optimization of a bulk carrier](image)

8. Conclusions and discussion

In this paper, we have described techniques for doing optimization of the lines of a ship using a RANS code which is very efficient with respect to both CPU-time and memory usage. Using the techniques described in this paper, one can optimize ship hull forms with respect to power and wake field quality, or even inversely design a model hull such that a given wake field is obtained. The design space is constructed by varying between some pre-defined basis hull forms. This was shown to be an effective way to set up the design space since it allows to define parametric hull form variations based on experience and to add new basis hull forms in the process.
The speed of the RANS code combined with parallelization over a PC network appeared to be very powerful; hundreds of RANS computations can be done in only a few hours. This type of parallelisation can be used not only in combination with systematic variations, but with genetic algorithms as well. The scale effects in the wake object function are quite significant.

Therefore, in this paper the optimization of the VIRTUE tanker was performed using RANS computations at full scale.

We have described two estimates of the required power to maintain a given ship’s speed. For the optimization of the VIRTUE tanker, we have shown that both estimates give similar results. However, in cases where the propeller diameter has to be chosen smaller, for example to guarantee sufficient clearance, the first method described in Section 4 to estimate the open water efficiency may give unrealistic results. In such cases, estimating the open water efficiency by evaluating the Wageningen B-series of the propeller is expected to give more realistic predictions of the decrease of the required power that can be obtained by hull form optimization. For the variations of the VIRTUE tanker, a clear correlation was found between the decrease of the nominal resistance and the power to sustain a given ship’s speed. However, the relative decrease of the power appeared to be lower than the decrease of the resistance. The latter was accompanied by a decrease of both the nominal wake fraction and the hull efficiency. This was not completely compensated by an increase of the efficiency in open water.

The final decrease of required power appeared to result from a subtle combination of several ingredients: a decrease of both the hull efficiency and nominal resistance, and an increase of the open water efficiency. It is therefore imperative to check that numerical disturbances coming from grid dependence, insufficient convergence or automatic grid generating procedures do not influence any of those ingredients.

References


PLOEG, A. van der; RAVEN, H.C. (2010), *CFD-based optimization for minimal power and wake field quality* 11th Int. Symp. Practical Design of Ships and other Floating Structures, Rio de Janeiro


IT Supported Maintenance, Inspection and Repair of Offshore Floating Windmill Farms

Philippe Renard, Bureau Veritas, Paris/France, philippe.renard@bureauveritas.com

Abstract

We are dealing here with the development of a Decision support system for the online maintenance of an offshore floating windmills farm. This system is expected to be the backbone of the future Condition Monitoring Centres, which will remotely monitor the farms. The automated Decision making module will take into account the specificities of such a farm, in particular the fact that it consists of many identical units. The use cases are investigated and the logics are explained with the necessary level of abstraction and generality, to accommodate future technological evolutions of offshore windmill sensors, NDT measurements and blade repairs.

1. Introduction

This paper uses the results of the ongoing Eurogia+ project named "HLC-AIMS": Eurogia+ is an Eureka "cluster", specialized in the promotion of energy oriented projects. This project deals with an "Hull Life Cycle - Asset Integrity Management System" of offshore floating windmills farms. It was created to develop a methodology and an integrated computer package for the monitoring such farms, which may consist of up to 100 offshore floating windmills (hereafter also referred to as "units") located at sea and normally far away enough not to be seen from the shore. Beyond the required modules for processing all relevant technical data, the ultimate purpose is to build a Decision making interface, advising the farm operator.

This system aims at replacing the real floating object, which is difficult to access at sea, by a virtual representation, for all maintenance, inspection and repair purposes. Seen in that way, applicative tools are ready to supply any requested "animation" of the data kept available in the central database.

It makes sense that off-shore windmills will have significantly more damages from structural causes than on-shore windmills; therefore we have concentrated our effort on structural monitoring.

The anchoring system is part of the project, and some people even think that it is a critical sub-system. It influences the motion of the unit in the waves and therefore the vibrations of the column. Anchoring gives shocks to the structure when it has reached its larger extension. Anchoring is clearly a factor in the RBI analysis. The monitoring of chains in-service is difficult and supporting equipment should be made ready for renewing of chains on site.

This is not a floating windmill design project: as a joke, we could say that we just need a "piece of wood with a fan on the top" as a demonstrator, and take margins regarding stability and wave height. Of course we will try to get closer to an operational design, which was optimized, to be sure that we keep the right order of magnitude for the parameters we are interested in. We will simulate some typical use cases, using records of past experimental data to emulate a floating platform at sea.

2. Condition Monitoring Centre

2.1 Operating schema

The system is designed to support the activity of a Condition Monitoring Centre (CMC), an office for qualified operators, equipped with computers running the system developed in the project. We may imagine the CMC located in high tech areas, far away from floating offshore windmill farms. When an alarm pops up on a computer screen, the CMC operators will call the local Service Centre, which is located on the shore close to the farm, who will send operators to the farm, by boat probably.
Orders are always associated with a feed-back confirmation that the order was executed. Feed-back confirmations will be anyway taken care of in the applicative software or in the workflow. They will be two levels of feed-back: the remote CMC sends alarms to the Service Centre, and receives feed-back from the Service Centre, and the local Service Centre gives orders to the farm personnel, and then receives feed-back from the farm personnel.

Commands to the offshore windmills will be fed back to the farm. We could think of commanding ballasting as a function of the sea state, or changing blade inclination when the offshore windmill tilts under the environmental forces.

2.2 Use cases for Condition Monitoring Centre

2.2.1 Influence of parameters on decision making modules

- In the Rules check module, when steel thickness is smaller, the steel goes from "good" to "substantial corrosion" and finally to "to be repaired", leading to decisions appropriate to each case.
- In the RBI module, the Probability of failure Pf(t) is associated to a given Life duration of the structure Lf: the shorter Lf, the higher Pf(t). If we observe higher stress at a given hotspot, the life duration becomes lower for the structure, therefore the probability of failure Pf(t) is higher and we need shorter inspection intervals to get back to the same level of risk.
- In the FEM module, if steel thickness becomes smaller, the same load is applied on a reduced steel section, therefore we get an higher stress at the hotspot.

2.2.2 Functioning of use cases

- As a typical illustration of how the Decision making module works, let's track the chain of events that occur within the system, when we observe that the stress variation measured by a sensor on an unit, is higher than calculated at the design phase.

As we have a higher stress variation at the gauge location, the FEM module will show a higher stress variation at the hot spot point. Then, combination of higher stress variation level at the hot spot point and original sea waves scatter diagram, provides a shortened expected Life duration (Lf) of the structure. This modifies the RBI analysis, because Pf(t) becomes higher, whence we finally need shorter inspection intervals.
FEM and RBI modules then feed the Decision making module. The Decision making module will require either to reduce the inspection intervals, or to reinforce the structure in order to bring the stress variation back to initially calculated values. This can be obtained by some sort of iteration with the FEM module.

- In following use cases, that we do not detail here, the Decision making module will also take the appropriate decisions:
  - If the measured sea states were higher than those used for the initial calculations, which means that the observed scatter diagram, viz. the matrix of sea states ($H_s$, $T_s$), shows higher values than the scatter diagram used in the design phase.
  - If a measured UTM shows a reduction of steel thickness.
  - If a damage is recorded on one unit.
  - If the observed frequency of the structure is closer to resonance than in design conditions.
  - If more vibration cycles were observed so that fatigue calculation needs to be re-run with more cycles.
  - If the observed vibration had a higher amplitude, so that we need to re-run both F.E.M and fatigue calculation.

3. Condition Monitoring Centre software platform schema

3.1 Central database (WCM format)

We have derived from the existing OpenHCM format used for ships and ships-like offshore objects, an exchange format called WCM and dedicated to offshore windmills. This is not that we expect many exchanges through this standard for floating windmills, but it is better to start from a standard which was developed over several years of discussion and which has been continuously updated since its first publication. WCM should represent the unit geometry and objective measurements.

We model the segments, i.e. the large hollow cylinders of steel which constitute a windmill column. One segment is significant in the real world because it is transportable by route. It is made out of several ringshells of homogeneous thickness, which are also represented in the model. We do not refine the model below the ringshell level, because we do not expect a lot of corrosion and we can consider the corrosion as homogeneous in a ringshell.

The format includes very few details for the blades. We define a blade section, typically, every 5 m along the blade. In the blade, all is optional, because we may not get much blade data, which could be deemed confidential.
3.2 General data flow schema

Data from the measurements module (e.g. stress gauges, ultrasonic thickness measurements UTM) and WCM databases, feed the FEM, RBI and Rules assessment modules, which in turn feed the Decision making module. In a few cases, the Decision making module is informed directly, thus bypassing the FEM, RBI and Rules assessment modules.

![Fig. 3: Simplified General Architecture schema](image)

4. Measurements on windmills

4.1 Real time measurements

We can expect that all sensors permanently installed on the windmills will send an enormous amount of data, in quasi real time, to the CMC. These data should be processed separately from the data stored into WCM, which are to be exchanged between the actors of the process.

To reduce the volume of these data, we may completely instrument a few windmills and very partially instrument the others. A correlation algorithm would enable generation of operational alerts from the very partially instrumented windmills.

The goal is to get alerts, and then know what to do when there is an alert. We are speaking here of several hours for intervention delays: in case of high stress, meaning storm, we will not be able anyway to go on board for intervention anyway.

4.2 Instrumentation of the windmill

- Generally speaking, permanent sensors will detect damage location and damage type, while, later on, NDT techniques will be used for verification.

  We will locate the hotspots, in order to install the stress sensors at the hotspots only, which will reduce both the number of sensors and the volume of measurement dataflow.

- Existing sensor technologies applicable to offshore windmills need additional research for appropriate use on board windmills, and especially on blades. Following sensors were considered as usable for the role of permanent sensors on offshore floating windmills:
- Vibration analysis (accelerometer or Laser Doppler vibrometer),
- Strains (extensiometric gauges, fibre optics, strain memory alloys),
- Video supervision,
- Water pressure captors (inside tanks),
- Impressed Current Cathodic Protection (ICCP) for corrosion monitoring (parts in seawater).

• Vibration detection is relevant, due to dephasing between wind and hydro-dynamic forces and different wave directions. One complete blades axis revolution lasts about 3 s, in which there are 3 passings of a blade against the column. Measuring vibrations is useful to analyse the observed problems (e.g. correlate the components breakdowns and the vibration levels) or in case of transformation of the structure. The evolution of the natural frequency during the life of the structure is indeed not well known. The structural natural frequency could be higher or lower than the frequency of the waves which is, for instance, about 0.42 Hz in the North Sea.

![Fig. 4: Offshore windmills frequencies](image)

• Regarding stress measurement technology, we can imagine using long-range extensiometric gauges (like sometimes used on ships' decks), fitted inside the windmill unit (to keep them protected from the wind and the sea).

• Nowadays, for offshore structures, there is usually no coating under the sea level, but ICCP protection instead. This should normally be also the case of floating windmills. ICCP monitoring will be part of real time monitoring, because the logics related to imposed current intensity, voltage and hull potential are well known.

4.3 NDT measurements

NDT available technologies applicable to offshore windmills are:

- Visual inspection,
- Ultrasonic inspection,
- Acoustic wavefield imaging (AWI, i.e. exciting the structure with acoustic waves and recording the disturbances caused by the wavefield at the surface of the structure),
- Acoustic emission,
- Eddy current,
- Tapometry (excitation of the structure with a periodically hitting hammer and measurement of the output of an accelerometer located on the structure),
- Infrared thermography (illumination of the structure and recording the infra-red radiation emitted by the structure),
- Shearography (comparison of the material surface before and after loading, using interferometry).
Not much corrosion is expected on offshore windmills. We could have fatigue corrosion at the column bottom, but, as it is in the air, wastage is expected to be very little. Cracks detection is more relevant here and the best detection tool for cracks is eddy current.

For blade inspection, we can use a carriage (like those used for window cleaning), or a caterpillar appliance held against the blade by vacuum or magnetic devices.

4.4 Environmental conditions

By contrast with ships, the offshore windmill will always stay at the same location at sea, therefore it will be possible to calculate a normative loading for this area and to update life duration prediction when we have the observed sea states. We will have the history of local sea states recorded by an oceanographic buoy, close to the windmills farm, and we may suppose that we will have the same sea conditions in the future for prediction calculations purposes.

5. Decision making at windmill level

5.1 Finite element model calculations (FEM)

The Finite Element Model (FEM) tool is expected to be useful as a diagnostic tool, to analyse unexpected stress or fatigue levels or to check the effect of structural reinforcements on the unit. We need a parametric FEM model to take care of thickness reduction from wastage and decision making loops in the system when trying to specify reinforcements: as there will be one single windmill model for the whole windmills farm, it will always be possible to define the structure in a parametric way. As shown in Fig. 5, we extract first the data out of the WCM file, in order to define the structure in a parametric way. Then we generate the FEM file, which is an Ansys file in the prototype module.

5.2 Risk Based inspection

5.2.1 Principles

The RBI analysis tells "what, when and how to inspect" for the individual unit: the main tool for decision making at the unit level is the RBI module.

In this project, we minimize the total operating cost of the unit, and calculate its optimum probability of failure.
5.2.2 Workflow description

- The values at unit level and at Component level are noted hereafter with different fonts.

- The probability of failure of the unit \( Pf \) is always kept lower than the Risk Acceptance Criteria for the unit, which either reflects the operator's attitude towards the risk or is imposed by normative limitations. The Risk Acceptance Criteria is usually expressed as a probability of failure per year (e.g. \( 10^{-3} \) failures/year).

- Given an unit with a probability of failure \( Pf \), we can allocate \( Pf \) down to all individual components of the unit, for instance to the blades and nacelle, the column and floats. Let's say that a given component is thus allocated the maximum probability of failure \( Pf \).

We draw, using formulas available in the literature, the probability of failure \( Pf(t) \) of a selected hotspot in this component as a function of time. After each inspection at time \( T_i \), \( Pf(t) \) is supposed to come back to 0.

![Fig. 6: Pf(t) at component level](image)

- For each component, we can draw the graphs of all possible scenarios \( S_i \) of defects, showing detection/non-detection, survival/failure branches. Knowing \( Pf(t) \), we can calculate the expected cost of maintenance of the component \( Cm[Pf(t)] \), which is the sum of the cost of repair, inspection and monitoring:

\[
Cm[Pf(t)] = \Sigma_i \text{Cost (} S_i \text{)} \times \text{Prob (} S_i \text{)}
\]

- For each component, we can calculate the cost of failure of the component \( Cf \), as the sum of the costs of failure related to personnel, environment and assets, corresponding to all possible scenarios consecutive to the failure of the component.

- We can propagate, upwards in the schema below, the failures from the lowest levels of the unit to the highest levels of the hierarchical expansion schema, keeping only into account the chains of events that will eventually lead to the failure (collapse) of the unit. Collapse of the unit can result from breaking of anchoring lines, floater ties, blades, etc. One may think of the famous video widely circulated through Internet, where a windmill blade breaks down in high winds, quickly causing the complete collapse of the column.

Starting from \( Pf(t) \), \( Cm[Pf(t)] \) and \( Cf \) of all components and using the "Hierarchical expansion schema", we obtain the probability of failure \( Pf(t) \), the cost of maintenance \( C_m \) and the cost of failure \( C_f \) of the unit.
Fig. 7: Hierarchical expansion schema

- For each unit $P_f$, we can run the above steps and draw the expected costs (viz. Cost of maintenance, Cost of failure and the Total cost) versus $P_f$. The minimum of the Total cost gives the Optimum $P_f$ and the corresponding $C_m$ and $C_f$ for the unit. We can also easily calculate the Criticality of the failure of the unit as Criticality = (Optimum $P_f$) x $C_f$

Fig. 8: Total costs curve

6. Decision making at farm level

6.1 Goals

The goals of the Decision making module are to minimize the farm exploitation cost over the whole operational life of the structure, and, in the future, to keep the electricity production beyond a certain level, in accordance with the clauses of the contracts covering the sale of the electricity produced by the farm.

In this project, we will estimate the involved costs only roughly. However, the succession of the tools to be used in the process must be correctly defined. Later on, specific parameters will be re-actualized for operating farms, but tools should remain relevant.

We will also answer to questions such as:
• For a given type of interruption of windmill production, when preventive measures are profitable?
• Do we always inspect the same windmills or we create rotations for the choice of inspected windmills instead.
• After a storm, when do we inspect a windmill? Immediately afterwards, 15 days later or when meteorological conditions become favourable again?

The "Decision making" module addresses mostly the farm aspects of this project, which means the "numerous units" issues. We expect to deal with a low risk cost, because there should normally be neither casualty nor pollution.

6.2 "Philosophic" issues

This "Decision making" task contains a lot of "philosophic" issues. The approach of the measurements in this project is rather theoretical and generic, rather than purely technological. Thus, generic parameters, such as sampling, probability of detection of defect (POD), or probability of false detection, will remain applicable when technology of offshore windmill sensors, NDT technologies and blade repairs evolve. We need to distinguish between continuous monitoring (e.g. stress gauges, imposed protective current, movements of unit, position of unit) and inspections where we go and inspect an unit and then find something (e.g. we have seen no crack). There are also processes which can be processed through RBI (e.g. corrosion, fatigue), and those which cannot be processed through RBI (e.g. coating degradation).

After an RBI analysis, there could be either a "local loop" to update one unit only, or a "farm loop" that would update all units in the farm. For instance, the probability of detection (POD) of a defect, which plays a role in "detection/no detection" branches of the graph of scenarios, is influenced by what we know of the whole farm condition.

The continuous update of the system parameters is a major feature of this system which could be qualified as a learning system. The fact that farms are made of numerous identical units, favours the efficient correction of theoretical values by the observed measurements.

On top of the Decision making module, there is always an expert's decision. For instance, the observation of higher stress than expected could induce the expert to have a look at sea conditions (exceptional stormy conditions) or vibration levels (maybe sea frequency is closer to structural frequency than expected).

6.3 Strategic options

Decision making should incorporate some sort of strategic options. This may go from the "discardable windmill" principle to a long term maintenance policy of all units of the farm. Class rules can also be seen as a simple sort of strategy: for instance, Class rules lead to the decision of plate replacement when a certain normative wastage is reached.

As there is no drydocking for offshore windmills, maintenance strategy becomes more significant than for ships. Among well known maintenance strategies, some do not apply to a small (less than 100) number of objects (e.g. MTBF).

6.4 Sampling issue

The main sampling problem can be stated as follows: if we inspect only a sample of windmills, what is the risk of drawing wrong conclusions at the farm level? A possible strategy is to prescribe that after a certain time, all units must be inspected. The rationale would be that, after a while, there is so much uncertainty that we cannot avoid physical inspection. But, if we always monitor the same units, the quality of follow-up is improved.
The most challenging issue is not, from the observed "number of default observations / size of the sample" (p/N) of the sample, to extrapolate to the average p/N for the population, with a certain degree of confidence, but to tell how this extrapolation influences the RBI process.

Acknowledgements

The author would like to thank the following persons for supplying valuable material used to write this paper:
- Antoine Rouhan, Jean Goyet, Bureau Veritas, Paris/France for WCM format, general system architecture schema and RBI module specification,
- Piet Moerland, Materiaal Metingen Europe, Ridderkerk/Netherland, for measurement and NDT technologies applicable to windmills,
- Arnaud Thiry, Amrouche Amrane, Philippe Rigo, Jean-David Caprace, ANAST ULg, Liège/Belgium, for vibrations and FEM module specification,
- Adrian Constantinescu, DN&T, Liège/Belgium, for windmills condition monitoring system and maintenance strategy,
- Zakoua Guédé, Benoît Bigourdan, Ifremer, Brest/France for demonstrator definition and specification of decision making at farm level.

References

RENARD, P.; ROUHAN, A. (2009), Asset integrity management system for ships and offshore units hull maintenance, 8th Int. Conf. Computer Applications and Information Technology in the Maritime Industries (COMPIT), Budapest
Optimization for Improved Propulsive Efficiency and Increased Bollard Pull

Tobias Huuva, Berg Propulsion, Öckerö/Sweden, tobias.huuva@bergpropulsion.com
Magnus Pettersson, Berg Propulsion, Öckerö/Sweden, magnus.pettersson@bergpropulsion.com

Abstract

Nozzles are mainly used to increase thrust at low and medium speed operations at high power, e.g. during tug and towing operations but are also used for increasing the efficiency at free running conditions for some vessel types. The standard nozzle profile is the 19A nozzle originating from Wageningen in the 1930:s and it is believed that this profile can be improved by the use of modern computational fluid dynamics methodologies. In the present work an automated optimization routine is setup in order to investigate the possibility to find high efficiency nozzle profiles for two operating conditions, free running and bollard pull. Also, manual variations of the nozzle dimensions are performed in order to see which nozzle that is optimal for a certain condition. Finally, model tests are done to validate the findings from computational work.

1. Introduction

Nozzles are today used in many different ships to increase thrust at low speed operation speed, e.g. in towing operations. Most vessels are using standard nozzle designs originating from the early 1930s, Carlton (2007). With the help of modern computational tools a number of new nozzle designs have been presented and highly improved efficiencies compared to the standard nozzles are claimed, especially as compared to the 19A nozzle, Kuipers (2007). The 19A nozzle is the most commonly used profile and it is suited for a large variety of applications.

Since the profile was developed more than 70 years ago and the fact that it is suited for several different applications, it is believed that there are possibilities to improve the profile. To investigate how much efficiency that can be gained by changing the profile and to investigate the influence of different parts of the nozzle an optimization study has been performed.

Two methods for automated CFD simulations were developed; one using a 2D axisymmetric simulation model with a simplistic model of the propeller, and one with 3D geometry including the propeller geometry. The automated methods use design of experiments, optimization algorithms and statistical methods to evaluate the effect of geometrical changes on the efficiency of the propeller and nozzle.

In order to investigate the influence on performance, at different operating conditions, caused by changing the main dimensions of the nozzle, manual CFD simulations were performed.

2. Procedure

In the automated 2D axisymmetric model, the propeller was simply defined as a line source with a pressure increase corresponding to a constant power. The resulting nozzle geometries showed significant improvement of thrust compared to the reference 19A nozzle. These improvements were well in line with claims from other investigations showing over 10% improvement both in bollard pull and in free running. To verify the results a full 3D investigation was performed to investigate if any 3D effects had been lost in the 2D representation. In this very careful investigation it was found that the 2D model could not predict the separation point in the diffuser part because of the representation of the propeller through the simple line source. In the 3D model the actual blade is represented and the rotation is represented by multiple reference frame (MRF). To overcome this problem with the 2D model a new optimization loop was performed using a 3D model. A large number of profiles were investigated using the automated optimization process. This investigation
gave valuable information on important parameters in nozzle design and information about limits in the possibility to design nozzle profiles with much higher efficiency as compared to the 19A profile for various conditions.

### 2.1. Geometric Representation

The system under study is the nozzle and propeller geometry, as seen in Fig. 1. The flow conditions far from the propeller-nozzle system were assumed to be rotationally invariant, i.e. the influence of the ship hull and the surface was neglected. The propeller chosen for the simulations has a diameter of 2.65 m and a pitch ratio of 1.19 and the nozzle geometry was parameterized by morphing boxes with the same parameterization for both the 2D model and the 3D model. The modification of the geometry is done by moving control points for the splines representing the nozzle corresponding to rotation, scaling and translation.

![Fig.1: System under study, nozzle with propeller and shaft](image)

### 2.2. Computational Domain

For the 2D simulations the computational domain consisted of a slice of a cylinder having the measures 13.5 m in radial direction and 41 m in stream wise direction. For the 3D simulations the bounding domain was smaller. The periodicity in geometry, i.e. four blades, was used to reduce the computational domain in the tangential direction. For the 3D free running nozzle optimization the computational domain was a quarter of a cylinder, with a 6 m radius, extruded 3 m in front of the nozzle and 13 m behind the nozzle, as depicted in Fig. 2. Using the same computational domain for the bollard pull simulations caused divergence. The cause of the divergence was that the flow through the nozzle was mainly sucked from the outlet, producing a *short circuit* between the turbulent nozzle jet and the nozzle inlet. The computational domain for the bollard pull simulations was instead reconstructed as a quarter of a sphere. If the computational domain is to be cylindrical, very large computational domain is needed.

![Fig.2: Domain used for the free-running condition](image)
2.3. Mesh

For the 2D optimization studies the mesh consists of 10 boundary layers close to the nozzle. The mesh outside the boundary layer consists of mainly quadratic cells except in the vicinity of the nozzle where triangular cells are used. The mesh consist of about 70 000 cells. In 3D, the mesh consist of a hexahedral trim mesh with 8 prism layers, which results in a $y^+$ on the nozzle between 30 and 200. The mesh consists of about 1.7 million volume cells for the free running cases and about 2.5 million for the bollard pull cases. The surface mesh size was about 5 cm. Fig. 3 shows the volume mesh at a cross section through the propeller. Refinement boxes were used to increase the resolution in the wake of the nozzle and propeller.

![Fig.3: Cross-section of the mesh on the 19A nozzle](image)

2.4. Profile Optimization

The optimization algorithms used are Latin hypercube (LHC) for design of experiment (DOE) and genetic optimization algorithms to control changes of a parameterized geometry of the nozzle. The nozzle geometry is then analyzed using automated CFD techniques, which handle meshing, solving and post processing of the results. The CFD results are then used as input for the optimization algorithm. In order to understand the response of the geometrical shape in the thrust, data from the simulations were also investigated using statistical tools, i.e. correlations, student t-test, scatter plots, parallel coordinate diagrams and analysis of meta-models such as polynomial fits and radial basis functions. For the 2D model the fan surface modeling the propeller was designed to give a certain power independent of the nozzle geometry. The total thrust evaluated over both the nozzle and the propeller was used as optimization parameter. For the 3D optimization method it was found that the main effect of increasing the thrust on the nozzle and propeller was to increase the torque on the propeller. By changing the rotational speed in the simulations it was found that the force/torque was reasonably constant. It was concluded that thrust/torque could be used as the optimization variable to maximize.

The first optimization in 2D was performed varying all input parameters and the force on the nozzle and the propeller was calculated. Using the resulting profiles, much higher efficiency as compared to the reference nozzle 19A is achieved. However, when these nozzles were investigated in 3D it was found that the total thrust from the optimized nozzles actually was less as compared to the 19A nozzle due to heavy separation in the diffuser part.
Based on the conclusion from the 2D simulations an automated CFD-method that includes the propeller geometry was developed and used. The objective was to maximize the thrust on the propeller and nozzle while minimizing the propeller power. The values of these three mentioned optimization parameters varied widely when changing the nozzle geometry. However, by varying the rotation speed of the propeller it was concluded that a reasonable parameter to optimize is the total thrust/torque, where the total thrust is evaluated over both the propeller and nozzle and the torque is evaluated over the propeller only. It was found that although changing the loads between the propeller and nozzle the efficiency or thrust/torque was more difficult to improve. For the 3D simulations two optimization loops were simulated, one for bollard pull and one for free running.

The fully automated CFD-optimization for 3D geometries was performed in three steps to limit the degrees of freedom. The first step was to investigate the thrust response for outer radius geometrical parameters. The second step was to perform optimization of efficiency, which for constant rotational speed for the propeller is proportional to thrust/torque, for free running using all parameters. The third step was to perform optimization of efficiency at bollard pull. This was performed using a DOE using the LHC algorithm and analyzing the result using the statistical tools of correlation coefficients, student t-test, parallel coordinates, meta models by first and second order polynomials and radial basis functions. For the free running optimization loop, the leading edge and trailing edge length parameters were locked to the length of the 19A nozzle. The optimization was performed by using 26 geometries and then 11 subsequent generations, resulting in a total of about 300 analyzed geometries. The resulting geometry is seen in Fig. 4. There is strong resemblance to the 19A profile. However, there is a tendency to a form a dolphin tail. The profile is also thinner, which is due to the need of minimizing the drag component on the nozzle.

![Fig.4: Free running profile after finalized DOE](image)

For the bollard pull optimization loop the leading edge and trailing edge length parameters were released within some limits. The optimization was performed by using 26 geometries and then 7 subsequent generations, resulting in a total of about 200 analyzed geometries. The profile of the resulting nozzle geometry is shown in Fig. 5. There is strong resemblance to the 19A profile; however the outer diameter is increased. On the contrary to the free running profile, there is no dolphin tail tendency.
2.5. Optimization Results

For both profiles the cylindrical section, where the propeller blade is passing, has been extended and the diffuser part is initiated further aft. The reason for this is that the propeller slipstream is disturbing the flow and if the diffuser starts directly there is a greater risk of separation in the diffuser which will limit the performance of the nozzle significantly. The optimized profiles, together with knowledge learnt from the computations, can now be used to produce a number of profiles with high efficiency together with good producability. By analyzing the optimized profiles a high resemblance with the 19A is found, with some small distinctions. The high speed nozzle has a smaller frontal area, has a dolphin tail and has a longer cylindrical section as compared to the 19A profile. The bollard pull nozzle has a larger outer diameter, a longer sectional length and a longer cylindrical area as compared to the 19A nozzle. As many conical and cylindrical parts as possible is to be used in the design of the nozzle since these are much easier to produce as compared to general double bended surfaces. The nozzle optimized for free running conditions improves the nozzle-propeller system by 3% in free run conditions with maintained bollard pull and the nozzle optimized for bollard pull conditions improves the bollard pull merit with 5% in bollard pull and decreases performance with 1% during free run condition as compared with the 19A nozzle. By definition, the open water efficiency is zero at bollard pull. Instead, a bollard pull merit figure can however be calculated based on the Bendemann static thrust factor, e.g. Carlton (2007).

3. Nozzle Size Study

One trend from the optimization study was very clear: a nozzle designed to give high bollard pull should be larger, both in diameter and in length, than a free running nozzle. The bollard pull nozzle reached the upper size limit which the optimization routine allowed, while the free running profile became very slender. To also account for the influence of the size of the nozzle, a manual study was performed with profiles scaled from the 19A profile. The scaling was performed with maintained diffuser angle and trailing edge thickness. If the diffuser angle is increased due to the scaling there is a risk that the separation limits is reached. This limit, where the separation occurs on a curved double bended surface in the wake of the propeller, is however hard to predict in the current computational
model and the diffuser angle is consequently kept constant. As a reference to the computations the Wageningen 22 nozzle, Kuipers (1992), with a length over propeller diameter \( \frac{L}{D} = 0.8 \), and Wageningen 24 nozzle, \( \frac{L}{D} = 1.0 \), are used. These nozzles are however scaled without maintaining the diffuser angle and can hence only be used as a reference to the computations. The computations were performed at four different advance ratios (\( J \)); at bollard pull condition, a towing condition, at 19A design condition and at an over optimal condition. At the 19A design point, a large nozzle loses about 3% in free-running condition as compared to a smaller sized nozzle while the larger nozzle gains up to 4% in bollard pull condition. If however the design point is adapted for the nozzle, higher efficiency gains can be found, Fig. 6.

![Computed open-water efficiency with various 19A scaled nozzle sizes](image)

**Fig. 6:** Computed open-water efficiency with various 19A scaled nozzle sizes. D corresponds to diameter change, 19A is D1.21, L corresponds to length change, 19A is L0.5D and LD corresponds to scaling of nozzle, 19A is LD0.5.

This is valid within a limited range in sizes, around 10% change in nozzle diameter and 25% change in nozzle length as compared to the 19A profile. For nozzles larger than this the efficiency in the free running condition becomes heavily decreased. If the diameter is increased above 10% there is a risk that separation occurs on the outer side of the nozzle and the free running efficiency is lowered by as much as 15-20%. The same occurs when the nozzle becomes very long, the Wageningen 22 nozzle has 6% lower free running efficiency as compared to the 19A nozzle and the 24 nozzle has 15% lower free running efficiency. When the nozzle becomes very small the bollard pull merit is decreased. The efficiency or bollard pull merit decreases with decreased nozzle size and as a reference, an open propeller can be used which has significant lower bollard pull thrust as compared to a nozzle propeller.

### 4. Model Test

Based on knowledge learnt from the optimization computations and on the study of nozzle size influence on the performance, two nozzles have been designed and tested experimentally. The nozzles are referred to as Berg Efficiency Nozzle (BEN) – High Speed (HS) and BEN – Heavy Duty (HD). The experiments are open water tests in a towing tank giving torque on the propeller and thrust on propeller and nozzle, respectively. The experimental setup can be seen in Fig. 7.
The results from the experiments validate the trends seen in the computations. The free running nozzle (BEN-HS) has a higher efficiency at the original design point, but it also maintains high efficiency and has an optimum design point at a higher J value while the bollard pull remains similar to the 19A nozzle. The heavy duty nozzle BEN-HD has a lower optimal design J, while the efficiency in bollard pull and towing conditions becomes significantly higher, Fig. 8.
The experimental investigation validates the computational findings that the BEN-HS nozzle is about 2% better than 19A at the 19A design point. However, if the design point is changed, further improvements can be made. Similarly, the efficiency of the BEN-HD is reduced by about 1% at the 19A design point. However, the BEN-HD is couple of percent points more efficient at lower J. The predicted increase in bollard pull for the BEN-HD nozzle is clearly justified by the experiments, not shown in Fig. 8.

5. Conclusions

The work with automated CFD-simulations of nozzles and propellers has resulted in important understanding for new designs of nozzles optimized for free running, towing and bollard pull conditions. The process includes automated process as well as manual adaptation to obtain an optimum design and to understand the underlying physical processes. In order to get realistic resulting trends from CFD-simulations it is important to include the 3D propeller geometry. This was concluded from using both a fully automated 2D axisymmetric simulation model and an automated 3D simulation model including propeller geometry. For simulations at bollard pull conditions it is effective to use a quarter of a sphere as boundary for the computational domain. The work of optimizing the performance of the nozzle-propeller system results in two new nozzle designs, one for free run and one for bollard pull conditions. The nozzle optimized for free running conditions improves the nozzle-propeller system by 3% in the computations at free running condition as compared to 19A. Also, the free running nozzle maintains the efficiency at bollard pull condition as compared to 19A. Larger efficiency increases can be achieved by changing the design point. In the computations, the nozzle optimized for bollard pull condition improves the bollard pull merit with 5% and decreases performance with 1% at free run condition as compared to the 19A nozzle. Parts of these gains comes from profile itself and part of the improvement comes from changing the outer dimensions of the nozzle. It is concluded that without changing the propeller design point, the 19A nozzle has a good performance and changing the profile has a limited effect if remaining at the same design point. However, if the propeller design point is adapted for the designed nozzle, the improvement will be greater. Generally for the nozzles, the thickness of the profile is the most important feature for free running condition and the geometry at the diffuser part of the nozzle is very important. For bollard pull, the outer dimensions is of large importance, a large nozzle, both in length and dimension, with an optimized profiles produces higher bollard pull as compared to the 19A nozzle setup. The experimental investigation of the BEN-HD and BEN-HS nozzle, which are based on the present investigation, verifies the results presented in the computational investigation. At the 19A free running design point, the difference between the nozzles is limited, but if the design point is adapted for the new nozzle design, the improvement will be greater.

References


KUIPERS, G. (1992), The Wageningen Propeller Series, MARIN Publication 92-001
MINOAS Project: Localization, Task-Allocation and Path-Planning Architecture System

Marco Bibuli, CNR - ISSIA, Genova/Italy, marco.bibuli@ge.issia.cnr.it
Gabriele Bruzzone, CNR - ISSIA, Genova/Italy
Massimo Caccia, CNR-ISSIA, Genova/Italy
Thomas Vögele, Robotic Innovation Center – German Research Center for AI, Bremen/Germany
Markus Eich, Robotic Innovation Center – German Research Center for AI, Bremen/Germany
Alberto Ortiz, Universitat de les Illes Balears, Spain
Leonidas Drikos, Glafcos Marine Ltd, Greece
Yannis Koveos, Glafcos Marine Ltd, Greece

Abstract

This paper reports the current developments of the MINOAS project. The project proposes the re-engineering of the overall vessel inspection methodology, by introducing an innovative system concept that incorporates state-of-the-art technologies, but at the same time formulates a new standardization of the overall inspection process. MINOAS proposes the development of a new infrastructure that complements human personnel by high locomotion enabled robots and 'teleports' the human inspector from the vessel's hold to a control room with virtual reality properties. In particular, the paper focuses on the architecture of the localization, task allocation and path-planning system specifying individual robots’ modules and the integration within a centralized system.

1. Introduction

Naval transportation of humans and goods is definitely one of the most cost and time efficient method so far. An always increasing safety standard level has to be achieved in order to improve the transportation service quality. To this aim, constant surveying and subsequent repairs have to be carried out to ensure structural reliability of the ships. The surveying and subsequent repairs require many of the tanks (cargo, fuel, ballast, etc) to be cleaned and ventilated, and for suitable access to be arranged - usually using scaffolding. In the case of older ships, this preparation of each tank is necessary to allow the repair crew to enter and access the element that has failed. Thus a surveyor can also use the same facilities. However, in a younger ship, the number of required repairs is small, so the bulk of these arrangements are only required for the survey itself. As a result a greater proportion of the preparation cost is entirely due to the survey, which for many well run ships is likely to reveal no defects in the early years.

Nowadays, the structural integrity is assessed through periodical inspections, carried out by surveyors who operate using well established methodologies and techniques. However, despite standardized methodologies and qualification criteria, the objectivity of the result cannot be guaranteed. The effort in the inspection tasks is huge and this often has to be carried out by a single surveyor within a short amount of time in order to return the vessel to service. The quality of the inspection results is also affected by both exogenous (exhaustion, stress, inaccurate judgment) and endogenous factors (different cultures, educational background, skills) whenever human intervention is used.

The maritime industry is forced to follow the general trend of rationalization so as to reach a higher level of standardization of the procedures related to marine transportation that will enable higher performances according to safety and financial criteria. One way to achieve this is through the incorporation of more technological means that increase the level of automation.

The MINOAS project (Marine Inspection rObotic Assistant System) is an European project under development in the framework of an EC funded project SCP8-GA-2009-233715 under the topic SST.2008.5.2.1 Innovative product concepts.
The project proposes the re-engineering of the overall vessel inspection methodology, by introducing an innovative system concept that integrates human personnel with high locomotion-enabled robots, effectively "tele-porting" the human inspector from the vessel's hold to a control room with virtual reality properties.

The proposed innovative system concept considers the assembly of a robot fleet with advanced locomotion capabilities and sets of tools that are dedicated to the tasks attached to the inspection process, the development of control techniques and algorithms that provide a semi-autonomous nature to the operation of the robot fleet and a hierarchical controller that enables online processing of the harvested data and operate as a Decision Support System in the aid of the inspector. A further description of the general concepts and ideas of the project is reported in Bibuli et al. (2010).

2. Localization, task-allocation and path-planning system architecture

The MINOAS overall system architecture relies on a hierarchical controller handling and coordinating the activities of a set of heterogeneous robots. As shown in Fig. 1, the mission control activities are organized in two layers handling the allocation of the inspection tasks to the members of the robot team and the path-planning of the robot motion respectively. Since both task allocation and path-planning require information about the overall robotic team, e.g. for generating cooperative behaviors and collision free paths, the corresponding modules are centralized in a hierarchical structure, which also includes monitors of the progress in motion, i.e. path following, and inspection task execution. On the other hand, navigation functions are distributed: each robot estimates its motion on the basis of on-board sensors, e.g. video camera, and external sensors, e.g. trackers for localization.

The result is that the embedded controller of each robot manages the execution of guidance and control motion primitives as well as navigation algorithms, generating suitable commands to the actuation system while filtering sensor data and perceiving the operational environment, e.g. detecting obstacles. As shown in Fig. 2, where the control architecture of a generic MINOAS robot is represented, the navigation system estimates the robot linear and angular position and speed and, if equipped with suitable sensors and perception system, detects obstacles, i.e. their position, and, when
possible, velocity and type. On the other hand, the guidance and control system receives reference path commands, specifying the desired robot motion primitive and its parameters, from the upper path-planner.

![MINOAS robot control architecture](image)

**Fig. 2: MINOAS robot control architecture**

3. Ship map representation

As discussed in Caccia et al. (2010), the position inside a ship is not specified by surveyors in 3-D Cartesian or polar coordinates, but with respect to compartments and structural components of the vessel itself according to a topology-based approach.

As a result, a hybrid tree-based structure is proposed. Acting in this way, the ship, object of the inspection, can be logically divided into main logical parts (sections) as it is usually done by human inspectors. Sections can be in their turn subdivided in sub-sections and so on, proceeding in a recursive way by using a top-down procedure. The partitioning terminates when a sub-section, i.e. an atomic operating area, is not more topologically divisible. The resulting tree-like structure is then integrated, and topologically transformed in a generic graph, with a set of arcs representing the connections between operating areas.

For instance, as depicted in Fig. 3 a generic ship can be subdivided in three main zones: rear, middle and front ones. In each on them, different areas to be inspected can be found. For instance in the middle zone, different cargo holds are present. Each cargo hold can be further detailed dividing between port and starboard sides, and then, for each side, listing all the elements to be inspected: bottom plating, sloping, side frames, etc. From this logical subdivision, a hierarchical tree structure can be obtained, as shown in Fig. 4.

4. Localization

4.1 Localization requirements

The precision requirements for the robot localization depend on the different application scenarios. The inspection process as looked at in MINOAS can be divided into 3 stages:

**Stage 1:** At this stage of the inspection process, a fast surveying is required. A larger area has to be covered with a type of visual sensor. This will give the surveyor a rough estimate of the ship coating and damages. Because a large area has to be covered, fast movement and a high maneuverability is required.

---

1 Two operating areas are connected with respect to a robot class when a robot of that class can travel from one area to the other one.
Installation of infrastructure has to be as minimal as possible because the team of surveyors has to move around the ship to get a good overview of damages. The sensor information can be coarse. After the first stage is completed, the surveyor decides which parts and areas of the ship need a more thoroughly investigation.

**Stage 2:** In stage two a pre selection of areas of the ship has already been done. The surveyor team needs now high quality images of the pre-selected areas. Because the areas to be covered with sensors are smaller, the locomotion speed and the requirements in maneuverability can be reduced. It is more important that the images taken at this stage have a higher precision in localization, i.e. the position of where the images are taken have to be more accurate. The procedure in Stage 2 is to get a better impression of the coating or damages on the ship and mark the damaged area in order to repair the damage or perform thickness measurements. A high resolution camera is mandatory at this stage. Also a low effort in system setup is required because the surveyor team has still to move around and inspect different parts of the ship, which have been identified in Stage 1.

**Stage 3:** At this stage the surveyor team needs to take thickness measurement at some damaged parts of the ship. The areas where to take the thickness measurement are narrowed down to a few spots, therefore speed and maneuverability requirements are low. The same is true for the installation of infrastructure, because the installation has only to cover small areas of the ship. Because the sensor type is tactile and covers non-destructive testing the system has to be very close to the wall. The locomotion type has to be climbing or clinging. During the measurement the system has to be stabilized.

Each stage has specific requirements for position accuracy. The positioning requirements indicator (low, medium, high) states how accurate the positioning and localization of the system has to be it ranges from low (positioning within several meters) to high (positioning within a few centimeters).
For the inspection process it is mandatory to localize the robot systems and the sensor data. The specification requirement gives different localization methods usable for different stages. The main issue is about the requirements for stage 1 and 2 of the inspection process. Both systems for the first two stages have to be lightweight and thus cannot carry onboard navigation sensors.

<table>
<thead>
<tr>
<th>Inspection stage</th>
<th>Description</th>
<th>Requirements for position accuracy</th>
</tr>
</thead>
<tbody>
<tr>
<td>Stage 1</td>
<td>Initial fast survey</td>
<td>Low</td>
</tr>
<tr>
<td>Stage 2</td>
<td>High-quality survey</td>
<td>Medium</td>
</tr>
<tr>
<td>Stage 3</td>
<td>Thickness measurements</td>
<td>High</td>
</tr>
</tbody>
</table>

### 4.2 Localization: proposed approach

As discussed in section 4.1 in inspection stages 1 and 2, the only option for measuring the flying and light climbing robots is to track them with an external device. The key idea is to use a man portable tracker which will give two angles (pitch and yaw) plus the distance. The pure laser based 3D trackers is in principle usable, but it is very sensitive to the angle of reflection of the target. The common tracking targets have to be aligned to the active tracker with a maximum angle of 20°. The problem arises once the tracking laser loses the target. In that case, it has to be caught again manually which can be difficult if for instance the crawler is several meters high in the wall. It is even harder to cache the laser beam again with an UAV. The option proposed for the MINOAS Project is a system which will consist of an active LED on the mobile platform. The external system will include an IR sensitive camera mounted on a 2DOF pan tilt unit. By this the camera can be used to track the bright light source of the LED. This will be used to get the two angles for pitch and yaw. Attached on the pan tilt system is a laser distance measurement device which will give the distance. Through this three dimensional vector the position of the mobile device (e.g. the robot carrying the LED) can be defined.

Summarizing the proposed external tracking system for accurate robot localization will consist of three parts, shown in Fig. 5: i) a pan tilt unit including a high resolution camera; ii) a high resolution laser range finder; iii) an active target.
As a first design, a 2DoF pan tilt was developed. This servo driven pan tilt has an exchangeable mount for several high resolution camera. The camera is actuated by 2 digital servos with an accuracy higher than 0.5°. This allows an active tracking, providing a distance with an accuracy of less than 10mm. The laser range finder is mounted parallel to the camera on the pan tilt unit.

5. Robot motion capabilities and constraints

In the framework of the MINOAS project four different typologies of robotic platforms are currently under development.

- A light, flying robot that will be used to perform fast and coarse observation/evaluation of the surfaces of interest (Fig. 6.a).
- A light inspection vehicle equipped with magnetic wheels, designated for fine inspection and structure failure marking (Fig. 6.b).
- A heavy magnetic-tracks crawler with high payload capability (Fig. 6.c), which mounts a small robotic manipulator equipped with surface cleaning and thickness measurements devices (Fig. 6.d).

5.1 Light, flying robot

The motion capabilities of the light flying robot will be as follows:

- \textit{take\_off}, which makes the vehicle self-initialize, take off and attain a certain height (the position before taking off is assumed as the home position, although it could be required to be identifiable by visual means - e.g. a cross over the ground - or any other means found suitable for the application);
- \textit{go\_to}, which makes the vehicle attain a pose \((p, \psi)\) relative to the home position, where \(p\) is a location specification and \(\psi\) is the yaw angle to attain (the vehicle enters in hovering state after attaining the pose);
- \textit{go\_to\_waypoints}, which makes the vehicle attain a list of poses \((p_k, \psi_k)\) (the vehicle enters in hovering state after attaining the last waypoint);
- \textit{go\_home}, which makes the vehicle go to the home position and attain a certain height (the vehicle enters in hovering state after getting home);
- \textit{land}, which makes the vehicle land at the current XY position.

5.2 Light inspection robot

The light inspection robot will be able to manoeuvre over horizontal, sloping and vertical smooth surfaces, executing motion primitives such as \textit{go\_to\_LOS}.

- \textit{go\_to\_LOS}: go to the desired point heading the target at each time, i.e. Line-Of-Sight (LOS);
Basic motion constraints are the maximum advance speed $v_{\text{MAX}}$ as well as the minimum radius of curvature as a function of the vehicle advance speed $r_{\text{min}} = r_{\text{min}}(v)$.

5.3 Heavy inspection robot

The heavy inspection robot will be able to manoeuvre over horizontal, sloping and vertical smooth surfaces, executing motion primitives such as $\text{go\_to\_LOS}$ and $\text{line\_following}$.

- $\text{go\_to\_LOS}$: go to the desired point heading the target at each time, i.e. Line-Of-Sight (LOS);
- $\text{line\_following}$: follow a target straight line; this motion primitive can be easily extended to the task of following a generic path, i.e. $\text{path\_following}$.

Basic motion constraints are the maximum advance speed $v_{\text{MAX}}$ as well as the minimum radius of curvature as a function of the vehicle advance speed $r_{\text{min}} = r_{\text{min}}(v)$. 

Fig. 6: MINOAS robotic platforms
5.4 Electro-mechanical arm for thickness-meter displacement

The heavy inspection robot will be equipped with an electro-mechanical arm able to displace a thickness measurement tool in the desired position and orientation. For the sake of completeness, at this generic and theoretical level, a five degrees-of-freedom manipulator structure is considered. It consists of a three-link planar arm, mounted on a revolute joint perpendicular to the vehicle base, and with a prismatic joint at the other end. This configuration allows the arm to work in a plane determined by the position of the first joint.

6. Task allocation

The final goal of the MINOAS project is to optimize the usage of a heterogeneous robotic team, to perform inspection mission of ships’ areas, like hull, ballast rooms, etc. The main objectives of such application are:

- the improvement of the safety level for human operators, that can rely on robotic platforms to inspect dangerous and/or hardly reachable areas;
- the development of suitable procedures for methodological and repeatable inspections;
- inspection resources optimization through formal methods.

The goal achievement mainly requires the definition of a procedure to classify the ship's areas, starting from the architectural drawings, and thus creating a table of the surfaces to be inspected, detailed by the characteristics of each area of interests: internal/external, underwater/in-air, ground/wall, etc. A complementary classification has to be done for the vehicle classes applied for the inspection: underwater vehicles, ground robot, climbing platforms, etc.

Depending on the capabilities (speed, type of inspection, operative constraints, etc.) of each class of vehicles and the number of available robots for each class, a procedure to optimally associate and share the robotic platforms among the different zones to be inspected has to be developed. This topic, referred as task allocation problem, has the aim of associating each vehicle, in space and time, to every interested area to be inspected.

The result is a complete association and scheduling of all the inspection resources to all the interested areas; in this context, the human operator can be also scheduled, for instance in those cases where no robotic platforms have the compliant characteristics to perform the inspection. Thus, the user can decide to perform a human inspection, or discard the inspection if it represents a too risky operation.

Due to the increasing interests in the multi-robot frameworks applied to cooperative explorations and sampling, inspections, area observation and protections, vehicle formations, search and rescue operations, a number of works have been proposed to face the problems of task allocation and path planning. A description of an empirical guideline for task allocation strategies in multi-robot systems, identifying four distinct task allocation approaches is proposed in work Østergaard et al. (2002). The idea is to perform an action selection for multi-robot coordination, generating a mapping from the combined robot state space to the combined robot action space. Work Meuth et al. (2009) explores the problem of optimizing the behavior of a swarm of heterogeneous robotic vehicles executing a search area coverage task; the problem is further complicated with the introduction of dynamic vehicle and environmental properties making adaptability a necessary requirement in order to achieve a high level of mission assurance using unmanned vehicles. The thesis work Tompkins (2003) examines scenarios where multiple autonomous agents collaborate in order to accomplish a global objective. In the considered environment, there is a network of agents, each offering different sets of capabilities or services that can be used to perform various tasks. The paper Bellingham et al. (2002) addresses the problem of cooperative path planning for a fleet of UAVs (Unmanned Aerial Vehicles). The paths are optimized to account for uncertainty/adversaries in the environment by modeling the probability of UAV loss. The approach extends prior work by coupling the failure probabilities for each UAV to the selected missions for all other UAVs. In work Gerkey and Mataric (2004) a general approach is
defined to face the problem of the multi-robot task allocation, proposing a formal study and avoiding
ad-hoc and empirical solution to the problem. A domain-independent taxonomy of multi-robot task
allocation problems is given, and it is shown how many such problems can be viewed as instances of
other, well-studied, optimization problems. Work Ducatelle et al. (2009) proposes a situation where a
swarm of robots is deployed to resolve multiple concurrent tasks in a confined arena. The tasks are
announced by dedicated robots at different locations in the arena. Each task requires a certain number
of robots to attend to it simultaneously. Work Maddula et al. (2007) considers an extended
environment with $M$ UAVs, $N$ targets and $P$ threats; the goal is to assign all the targets to the UAVs so
as to minimize the maximum path length, divide work equitably among the UAVs, and limit the threat
faced by each UAV. A four stage approach is used to address this problem.

Relying on the analysis and study of the different existing technique for task allocation, an ad-hoc task
allocation algorithm is proposed for the application to the MINOAS project. On the basis of the
vehicle and area classifications, a task allocation algorithm, performing an optimal mapping search
can be developed in order to assign each vehicle to the area that most fit with its characteristics and
capabilities. The assignment of more vehicles to a specific area requires the further vehicle allocation
in space and/or time; for instance, two observing vehicles can inspect a floor at the same time,
observing half the floor each one. But one observing vehicle and one marking vehicle may be
scheduled in time, i.e. the observing robot will start the task looking for damages, and then, delayed in
time, the marking robot will mark the damaged zones indicated by the other vehicle.

The algorithm for the task allocation requires the following set of information:

- **ACTUAL_NODE**: the current area, described by the tree node, to be allocated to the robotic
  resources;
- **CHILDREN_NODE_NUMBER**: the number of children of the actual node;
- **CHILDREN**: the list of children nodes;
- **LEAF_NODE**: a flag variable specifying if the current node is a leaf of the tree or not;
- **TOTAL_AREA_COUNT (TAC)**: a recursive count of the sub-areas linked to the actual node;
- **AVAILABLE_ROBOT_NUMBER (ARN)**: the number of available robots that can assigned
  for area inspections;
- **ASSIGNED_ROBOT_NUMBER**: the number of robot assigned to the actual area.

The first phase of the task allocation procedure requires the progressive count of all the basic structure
composing the target of the inspection, in order to compute the TOTAL_AREA_COUNT values; in
other words, for each parent node of the tree, the count stored in each child node has to be summed
and then stored, starting from the lowest level of the tree, where leaf nodes have count equal to 1 by
definition.

The next step is to allocate the available robots, thus while AVAILABLE_ROBOT_NUMBER is
greater than zero, the following operations are executed:

- the **CHILD** node, not allocated yet, with the minimum TOTAL_AREA_COUNT is selected;
- the number of robots to be assigned is computed as:
  \[
  \text{robots} = \left\lfloor \frac{TAC(\text{child})}{TAC(\text{actual_node})} \times \text{ARN} \right\rfloor
  \]
  - if (\text{ANR} \geq \text{robots}) the number of assigned robot for the child node is equal to \text{robots}, else it is
    equal to ANR.

When no more robotic platforms are available, the remaining not assigned areas are allocated to the
robots using the following procedure: each remaining area is assigned to the robot with the current
minimum number of areas to inspect.
Following these guidelines, a balanced allocation of the areas to the robotic platform is carried out. The reader can refer to Caccia et al. (2010) for details and examples of the algorithm.

7. Path-planning

The path-planning problem is decoupled in two layers handling the discrete motion of the agents through the atomic operating areas, considered as a set of contiguous cells, and the continuous motion of the agents inside a specific area respectively. When a direct interaction with the environment is required through the manipulator, e.g. for executing thickness measurements, a third layer to plan the arm motion in order to position the end-effector in the desired place is considered. The result is a hierarchical planning scheme handling the motion of the robotic vehicle through and inside the operating areas and the final high precision approach of the end-effector to the target:

- **Structure-based path-planning**: the so-called structure-based path-planning, given an agent and its motion capabilities, has to solve the problem of finding a suitable sequence of connected operating areas between the start and the goal position, where, as already stated in this report the problem assumes the form of path-finding in a graph with additional constraints given by the simultaneous presence of multiple agents.
- **Continuous/local path-planning**: the task of the local continuous path-planning is of computing a path, compatible with the maneuverability constraints of a given agent, connecting a couple of points inside a specific operating area. Additional constraints on the agent orientation at the beginning and at the end of the manoeuvre can be given.
- **Manipulator path-planning**: the task of the manipulator path-planning is to compute a path, compatible with the joint configuration of the robotic arm, able to drive the end-effector in the desired position and orientation. Since the precision in the arm motion control is higher than in the case of the vehicle, e.g. in the case of its yaw orientation in the working position, the manipulator path-planning has to compensate for these uncertainties.

7.1 Structure-based path-planning

Referring to the topological representation of the ship introduced in Caccia et al. (2010), the problem of structure-based path-planning reduces to find a suitable sequence of connected operating areas between the start and the goal position inside a graph. Indeed, the ship map assumes the form of a hierarchical tree, where the operating areas constitute the leaves. Leaves are then connected, according to their topology and robot motion capabilities, constituting a graph. Once defined, also heuristically, the cost of the transitions between each pair of connected leaves (e.g. their distance), it is possible to find (quasi-)optimal solutions to the problem of traversing the graph from a start to a goal node. The proposed algorithm consists of three stages, the first offline and the others online.

**Stage 1: Construction of node connections and checking on the map structure**

1.1 Given the connections between the tree leaves, the connections between ancestor nodes are constructed according to the following rules:
   - if a leaf $p_2$ of layer $n$ is connected to a leaf $p_1$ of layer $m$, with $n > m$, then all the ancestors of $p_2$ of layer $m$ are connected to $p_1$;
   - given two nodes, $n_1$ and $n_2$ of the same layer, if there are two connected leaves $p_1 \in \text{child_of}(n_1)$ and $p_2 \in \text{child_of}(n_2)$, then $n_1$ and $n_2$ are connected.

1.2 For all the nodes in the tree, verify if their children constitute a connected graph, otherwise signal the consistency failure to the human operator.

**Stage 2: Construction of the hierarchical path**

Given a start and a goal leaf, a sequence of ancestors connecting them through the map tree structure is found: in practice, the tree structure is backed up from the start and goal leaves until a common an-
cestor or a couple of connected ancestors at the same layer (or an ancestor connected to the other leaf) are found. In this way a couple of node sequences are found, originating from the start and goal leaf respectively. The hierarchical path is thus obtained by joining them after having reversed the goal sequence.

**Stage 3: Expansion of the hierarchical path**

Once computed, the hierarchical path has to be expanded in order to determine a suitable sequence of operating areas, i.e. leaves, connecting the start and goal ones. This is done, expanding the hierarchical path from its lower layer until there are nodes. Thus, considering a generic layer \( i \), three steps are performed.

3.1 Nodes at layer \( i \) are expanded at layer \( i+1 \). Considered that inside the sequence representing the (expanded) hierarchical path each node has a predecessor and a successor, expanding a generic node \( n \) means to substitute it with a couple of set of its children determined as it follows. For each node \( n \) at layer \( i \), for each connection between node \( n \) with node \( m \), i.e. for each pair of nodes in the hierarchical path sequence, substitute node \( n \) with the set of nodes \( n_C \) such that \( n_C \) is child_of \((n)\) AND \( n_C \) is connected_to \((m)\), when node \( m \) is a leaf or is at layer \( i + 1 \); or \( n_C \) is connected_to \((m_C)\) where \( m_C \) is child_of \((m)\). The result is an expansion of the hierarchical path at level \( i+1 \) where the nodes with the same ancestor are not, in general, directly connected.

3.2 Determine a predicted optimal path at layer \( i + 1 \). Given the unconnected expansion of the hierarchical path at level \( i+1 \), dummy connections are established between the nodes originated by the same ancestor on the predecessor and successor sides. The result is a connected graph at layer \( i + 1 \) that can be searched with a conventional algorithm to find an optimal path between the start and goal leaves.

3.3 Find optimal paths connecting nodes with the same ancestor. The path computed at stage 2 can contain consecutive nodes, originated by the same ancestor, which are not directly connected. At this stage optimal paths connecting these pair of nodes are computed applying conventional path-planning techniques to the reduce sets of nodes with common ancestor. A final result example of the structure-based path-planning is reported in Fig. 7. For a complete description of the procedure, user can refer to (Caccia et al., 2010).

Fig. 7: Local/continuous path-planning for the inspection of floor, side-slope and web-frame areas with five robots
7.2 Continuous/local path-planning

Once that the ordered sequence of basic structures to be inspected, obtained by the task allocation, complemented by the list of structures that have to be traversed to reach non-adjacent inspection areas, obtained by the previous path-planning phase, then the continuous/local path-planning operation can take place. This phase basically produces the paths that the robots will have to follow in order to visit all the structures defined by the inspection plan, with the constraint of exploring all the points of interest defined for each single structure.

The goal of the continuous/local path-planning phase is to produce a feasible motion path for each class of inspection vehicle, taking into account obstacles and constraints of the interested structures and, at the same time, minimizing the path length to visit all the points of interest of each area.

For each basic structure, a set of interesting inspection points is pre-defined; as emerged by interviews with expert inspectors, plate structures as floor and sloping parts have usually five points of interest. For the web-frame structures, the number of points depends on the length of the structure itself.

The path-planning is composed by two stages: in the first one, the parts of path to connect adjacent structures are computed, while the second stage regards the construction of the part of path to visit all the points of interest, within the structure, minimizing the length of the path. During stage 1, for each pair of subsequent structures, a research among the way-point sets of each structure determines the pair of way-points, one for each of the two structures, characterized by the shortest distance. These two way-points are connected to become part of the overall path. In the case of passage through 'bridging' structures, the transfer path can be computed automatically keeping into account the geometry of the structures involved, or adding a set of bridging points, similar to inspection waypoints but used only for robot transfer. During stage 2, a minimization algorithm is applied to compute the optimal path to visit all the way-points of each structure. The parts of path computed during stages 1 & 2 are then merged together to obtain the final path that allows the inspection of the overall structure subject to the task allocation plan. An example of local/continuous path-planning is shown in Fig. 8, where the reference paths of five robots are planned for floor, side-slope and web-frames inspection.

![Fig. 8: Local/continuous path-planning for the inspection of floor, side-slope and web-frame areas with five robots](image)
7.3 Manipulator path-planning

Due to the specific tasks required to the manipulator, that is assumed to work in a structured environment and suitably positioned by a maneuvering carrier vehicle, the synergy of two (complementary) approaches have been considered for its path-planning:

1. definition and planning of default paths also useful to support tele-operation, or human supervised operations;
2. handling of the manipulator inverse kinematics.

In particular, the MINOAS arm has to be able to compensate possible uncertainties in the positioning of the carrier robot due to the harsh environment conditions as well as maneuvering capabilities and sensor performances. In this context, the path-planner has been designed in order to allow human intervention focusing on the possibility of introducing local corrections to nominal conditions.

According to the general objective of the MINOAS project of developing technology that can be relatively easily introduced in field operations, and considering the operational constraints given by the presence of the structural components of the ship in the operating area, that can be seen as obstacles to be avoided, a two stage path-planning strategy is proposed:

1. definition of default way-points for routine operations;
2. computation of paths between default way-points through kinematic inversion.

In addition, a prismatic arm joint and the introduction of local waypoints would allow the system to comply with uncertainties related to maneuvering, sensing and environment modeling. Fig. 9 shows the robotic manipulator geometry and a motion planning example. Mathematical details are reported in Caccia et al. (2010).

![Fig. 9: Robotic manipulator geometry and motion planning example](image)

8. Conclusions

According to the concept of the MINOAS system as an assistant to ship inspection based on robotic and virtual reality technologies, the proposed approaches, at least at the highest levels of task allocation and path-planning, have been designed in such a way of facilitating interactions with the human supervisor. Indeed, the human operator needs to have the possibility of modifying the activity plan on the basis of his/her information at any time. The two step task allocation procedure, decoupling space and time robot allocation to the various tasks and operating areas, as well as the hierarchical path
planner at the layer of structures to be traversed by the robots for exploring/reaching the desired working points, have been thought with in mind this basic requirement.

References

BELLMINGHAM, J.S.; TILLERSON, M.; ALIGHANBARI, M.; HOW, J.P. (2002), Cooperative path planning for multiple UAVs in dynamic and uncertain environments, IEEE Int. Conf. on Decision and Control.


Using a Mathematical Symbolic Solver to Transfer Analytic Theory to Numerical “First-Principle-Methods”

Wilfried Abels, Hamburg University of Technology, Hamburg/Germany, w.abels@tu-harburg.de

Abstract

This paper describes an approach using a mathematical symbolic solver to implement numerically complex mathematical theories. This will be illustrated for an unsteady potential theory to calculate the lift of propeller blades rotating in a given wake. After a short overview about the analytic theory, it will be explained how the mathematical model was developed using the symbolic solver MAPLE. Afterwards this mathematical model was implemented as FORTRAN code and verified. The approach resulted in a compact and very fast method which allows analysing the influence of a wake to the lift distribution on a blade.

1. Introduction

The handling of physical/numerical models is a main task during the development of a technical product. An engineer needs reliable information to design new innovative products. The keen competition within the shipbuilding industry drives the trend towards tools which can analyze new design features rather than using empirical know-how. In modern ship design, reliable information has to be available in a very short time. The concept “Design in Seven Days” (D7D) needs software tools, which generates the required information fast and reliably, Krüger (2003). The wake is one example for important information in ship design. The interaction between wake and propeller is complex. It affects efficiency, vibrations and rudder forces. Wrong design decisions can result in difficult and expensive problems. It can be a significant competitive advantage to have tools which assure correct design decisions.

These tools have to be adjusted to the relevant physical features. Often it is not necessary to analyze the physics comprehensively. The task in early design is to make a decision between two or more design options. For the used software tools, this means that a simplified qualitative model capturing the most relevant effects often suffices. A fast model with known simplifications and a known reliability is often the best choice in design.

Thus the main task in the development of design tools for is to think about the relevant physical features for the model. Then the model has to be translated into a numerical description, which can be implemented in software tools. In the past, especially in the second half of the 20th century, there were many hydrodynamic investigations based on sophisticated analytical theories. Due to severely limited computer performance, there was a strong focus on using analytic techniques. The advantage of highly analytical approaches was that they were very fast, and sometimes even allowed manual solutions. Analytical methods are still very popular for early design as they handle many design variations in short time. Calculation methods which need more than one day for a design evaluation are generally useless for design.

This paper describes an approach to simulate a rotating propeller within a given wake. The aim is to analyze effects of propeller and unsteady wake in early design. Although the used theory so far does not allow using a dedicated propeller, qualitative effects of a specific wake are taken into account and can be used to assess the influence of wake to the unsteady lift distributed on the blade.

A significant aspect is to handle the free vortices downstream the propeller. This is a difficult task. Classical lifting-line approaches can only handle a homogeneous wake, Isay (1964). Even modern three-dimensional viscous approaches have problems to capture the free vortices accurately and they are anyhow too time-consuming for early design. The QCM (Quasi-Continuous Method) approach offers a good alternative. QCM uses a panels on the blades and wake panels to model the free
vortices, Streckwall (1997), Abels (2006). The wake panels have the problem that they generate singularities on a rudder placed behind a propeller.

To avoid these problems, an unsteady potential theory was used, which describes the propeller blades as a bounded circulation. For solving the boundary condition, the Biot-Savart law was integrated from the bounded circulation to infinite downstream the propeller. Because this integration has been done analytically, the whole endless circulation downstream could be taken into account without using discrete wake panels. This approach allows staying compatible with other panel methods.

2. Overview of the unsteady potential propulsion theory

This paper describes the investigation in an unsteady propulsion method, Zwick (1962). The flow is modeled within potential theory with a 3d distribution of free circulation downstream of the propeller. We consider an incompressible fluid, without sources and sinks. The propeller and wake are described in a cylindrical coordinate system \((r, \theta, z)\). The propeller has constant rotation of \(\omega < 0\) and moves with velocity \(v_0 > 0\) in the negative direction of the \(z\)-Axis. The propeller blades encounter a wake \((u_1, v_1, w_1) = \left(u(r, \theta, z), v(r, \theta, z)\right)\) with a period of \(2\pi\) in \(\theta\). In a cylindrical coordinate system fixed to the propeller blades, the propeller experiences a local flow \(u(r, \theta - \omega t, z) = (u, v, w)\). The propeller has \(n\) blades, the hub is at \(R_h\) and the tip at \(R_t\). The chord length is \(c\) with \(\text{chord length} = 2\pi r = \text{const.}\)

![Fig.1: Propeller model, Zwick (1962)](image)

The model of the propeller is particular. The blades are modeled as cylinder sections assuming constant pitch, Fig.1. The wake is modeled with axial and radial components, Fig.2.

![Fig.2: Axial and radial components of the used wake, Zwick (1962)](image)

Despite the simple propeller model, this approach can be used to analyze effects of the wake qualitatively. The propeller is described as a fixed circulation with free circulations leaving
downstream. The mathematical model based on an analytic solution of the Biot-Savart law for the fixed and free circulations at the $\frac{1}{4}$ point $-\beta_0$ and the flow constraint is solved for the $\frac{3}{4}$ point at $\beta_0$. For $w = (w_r, w_\theta, w_z)$, the following equation has to be solved:

$$v_0 + u_y + w_y - \partial (r) \cdot \left( \omega \cdot r + u_\varphi + w_\varphi \right) = 0$$

For $\phi = \varphi_0$, the following equation has to be solved:

$$f_{or} \left\{ \begin{array}{l}
R_0 \leq r \leq R_a \\
y = r \cdot \partial (r) \cdot \beta \\
\phi = \frac{2\pi}{q} \cdot v + \beta \quad v = 0, y
\end{array} \right\}$$

(1)

Now the propeller can be modeled as a circulation in the following form:

$$\Gamma(r, \varphi) = \Gamma \left( \varphi, \frac{2\pi}{q} \cdot v - \beta - \omega t \right)$$

By using such a circulation, it is in principle easy to describe the free circulation downstream the propeller. The “Helmholtz theorems” for fluid mechanics give the free transversal circulation:

$$\frac{\partial \Gamma \left( \varphi, \frac{2\pi}{q} \cdot v - \beta - \omega t \right)}{\partial \varphi} \rho_y$$

(3)

The direction is $(0,1,\beta (\varphi))$. In the same way the free longitudinal circulations are defined:

$$\frac{\partial \Gamma \left( \varphi, \frac{2\pi}{q} \cdot v - \beta - \omega t \right)}{\partial \varphi}$$

(4)

The direction is in this case $(1,0,0)$. To describe the induced velocities, the Biot-Savart law has to be used:

$$v = \frac{\Gamma}{2\pi r}$$

(5)

In principle, it is easy to describe the unsteady flow by a combination of Eqs. (1) to (5). The mathematics becomes a bit complicated because of the cylindrical coordinates and the geometrical description of the Biot-Savart law. To calculate the induced velocities on the blades, it is necessary to sum up the integration over the radial axis of every blade and to perform an infinite integration in direction of the propeller along the free vortexes:

$$w_q = \sum_{q=0}^{R_q-1} \int \int \frac{\partial \Gamma (\varphi)}{\partial \varphi} \cdot R(\varphi) \cdot d\varphi \cdot dq$$

$$w_z = \sum_{q=0}^{R_q-1} \int \int \frac{\partial \Gamma (\varphi)}{\partial \varphi} \cdot R(\varphi) \cdot d\varphi \cdot dq$$

(6)
The approach for solving this set of equations was described in Zwick (1962). The solution used a numerical approximation of the circulation $\Gamma$ in the following form:

$$
\Gamma^n(\varphi, \kappa) = \sum_{r=1}^{\nu} A_r(\varphi) \cdot e^{i\kappa r} 
$$

$$
A_r(\varphi) = \sum_{l=1}^{\nu} A_{r,l} \cdot f_{\text{func}}(s) 
$$

$$
\varphi = \frac{r}{R_a}; \quad \kappa = -\beta - \omega t 
$$

The coefficients $A_{r,l}$ can be calculated by a system of linear equations:

$$
\sum_{l=1}^{\nu} B_{r,l} \cdot A_{r,l} = C_{r,l} 
$$

The matrix $B_r$ symbolizes the effects of the free vortexes and $C_r$ represents the effects of the wake. Once these coefficients are calculated, the system of equations is rapidly solved, because the relevant parameters $\varphi$ and $\kappa$ are small, typically less than 10. Thus, the numerical effort to solve the linear equations is nearly irrelevant.

In Appendix A, the definition of Matrix $B_r$ is shown exemplarily. Its derivation is in the original paper and not part of this investigation. Important is rather how the mathematical model is transformed into a design tool.

3. Handling analytic models with symbolic mathematical tools

Many sophisticated models were developed in the past. But often they are not easy to handle. Much know-how has been lost over the years and often nobody can explain their background anymore. In addition, our approach to analytical theories has changed over the years. Fifty years ago, we used slide-rulers or analog integrators. Today it is important to bring analytical models into computer form.

The numerical verification of a complex analytic theory is an ambitious task. The original paper shows only the input data, Fig.2, and the results as graphics, Fig.3. Therefore, an analytical mathematical model was developed and implemented using the symbolic solver MAPLE. A symbolic solver has the advantage to describe the mathematical model directly by the needed equations. Integration and derivation can be handled analytically or numerically. Problems of numerical modeling are not important. Discretization and definition of usable data structures for a specific computer language are not necessary. It will use an own solving mechanism without much user interaction.
The numerical performance of such a symbolic solver is slow. The great benefit lies in rapid prototyping. After building such a mathematical model, it can be used as a reference model for further developments and for testing. This task is not easy to handle, because the numerical implementation of a complex mathematical model has many other constraints, not just mapping mathematical equations to a computer language. A numerical method, to be efficient, has to take into account constraints of the processing unit and the usable data structures of the preferred computer language.

In any case, accurate calculation results have to be guaranteed. A rapid prototype model is very useful for this purpose. During the software implementation, the engineer needs support to control and debug the software. In principle, such an implementation is not very complicated. All needed equations are available, but then human beings tend to make mistakes when copying long mathematical structures. In this step, the main problem was that there were no interim values. The mathematical description could be defined nearly in the same way as listed in Appendix A. Between the input data of the wake and the resulting unsteady circulation, there are many mathematical calculations with many possibilities to make mistakes. And statistically it is sure to do some. Already much time was spent to develop the mathematical prototype to reproduce the published circulations. The subsequent implementation in a computer language is much more complicate. Now, numerical problems became the focus of the development. But a powerful reference system was available for a stepwise implementation in FORTRAN. The prototype model allowed generating all needed interim values. This supported a clear and controllable implementation process during the whole time.

4. Transferring the model from analytic to a numerical description

The numerical implementation of such an analytic theory has aspects which differ from a pure analytic view, as illustrated in two examples. In both cases, the problem results from digital nature of computers. The positioning of supporting points can be problematic if they coincide with discontinuities. Even if it is clear that such a discontinuity could be canceled, it has to be done in the correct way. A correct mathematical procedure has to be found to handle them, before it can be implemented on a computer.

4.1. A case study with no support for the equality of to real values

The task of the unsteady method is to calculate the vector $A_k$ which describes the solution of the unsteady circulation on the blades. Thus, matrix $B_k$ and vector $C_k$ of Eq. (8) have to be calculated. Matrix $B_k$ is defined in Eq. (21). But during the calculation of the sum over $K^\tau_{kl}(x_1,\alpha_\mu)$ a problem occurs if $x_1 = \alpha_\mu$. The function is defined for $x_1 < \alpha$ and for $x_1 > \alpha$. But the discrete sum causes that $x_1$ becomes equal to $\alpha_\mu$ for $l = \mu$. This discontinuity has to be handled before the function could be calculated. To do this, the relevant part of Eq. (21) has to analyzed:

$$
\frac{2\pi}{(k+1)^2} \sum_{x=0}^{k} K^\tau_{kl}(x_1,\alpha_\mu) \sum_{x=1}^{k} \lambda \sin\left(\frac{\lambda \pi x}{k+1}\right) \cos\left(\frac{\lambda \pi x}{k+1}\right)
$$

(9)

To cancel this discontinuity at $x_1 = \alpha$, the neighborhood has to be taken into account. The sum $\sum_{x=0}^{k} K^\tau_{kl}(x_1,\alpha_\mu)$ is an approximation of an integration over $\alpha$. This mean the discontinuity could be canceled by solving the integration over this discontinuity:

$$
\int_{\alpha = \frac{\Delta x}{2}}^{\alpha + \frac{\Delta x}{2}} K^\tau_{kl}(x_1,\alpha) \, d\alpha = \lim_{\alpha \to 0} \int_{\alpha - \frac{\Delta x}{2}}^{\alpha + \frac{\Delta x}{2}} K^\tau_{kl}(x_1,\alpha) \, d\alpha + \int_{\alpha - \frac{\Delta x}{2}}^{\alpha + \frac{\Delta x}{2}} K^\tau_{kl}(x_1,\alpha) \, d\alpha
$$

$$
= \lim_{\alpha \to 0} \frac{\Delta x}{2} K^\tau_{kl}(x_1,\alpha - \frac{\Delta x}{2}) + \frac{\Delta x}{2} K^\tau_{kl}(x_1,\alpha + \frac{\Delta x}{2})
$$

(10)
This approach solves the discontinuity at $\sigma = s$ can be canceled in the following way:

$$K_i^\sigma(s,\tau) = \lim_{\mu \to 0} \frac{K_i^{\mu}(s, s - \mu) + K_i^{\mu}(s, s + \mu)}{2}$$  \hspace{1cm} (11)

Now the function $K_i^\sigma$ can be evaluated everywhere, by adding two new cases to Eq. (23):

$$K_i^\sigma(s, \alpha, \mu) = \cdots \quad \frac{q}{\mu \pi S^\sigma(s)} \left\{ \begin{array}{ll}
2 & \tau = 0 \quad s < \sigma \\
1 - \theta^\sigma(s) & \tau = 0 \quad s = \sigma \\
-2\theta^\sigma(s) & \tau = 0 \quad s > \sigma \\
[1 + \theta^\sigma(s)] \left( \frac{s}{\sigma} \right)^{|\tau|} & \tau \neq 0 \quad s < \sigma \\
0 & \tau \neq 0 \quad s = \sigma \\
-\left[1 + \theta^\sigma(s) \right] \left( \frac{s}{\sigma} \right)^{|\tau|} & \tau \neq 0 \quad s > \sigma
\end{array} \right.$$  \hspace{1cm} (12)

Further it has to be pointed out that for the function $K_i^\sigma$ does not have implemented with the real parameter $s_1$ and $\sigma$, but with the discrete parameter $l$ and $\mu$. This was necessary, because a computer cannot test the equality of two real values well. By matching the case study of (12) to the integer parameters $l$ and $\mu$ the problem does not occur any more.

4.2. Example of a numerical integration with a discontinuity

A main task during the calculation of $K_i^\sigma(s_1, \alpha)$ is to determine the integral

$$R_{1\pi}^{\mu}(l, \mu) = \int_0^{2\pi} \cos(\tau x) \left( s - \theta^\mu(s) \theta^\mu(\sigma) - [s_1 - \alpha \theta^\mu(s_1) \theta^\mu(\sigma)] \cos(x) \right) dx$$  \hspace{1cm} (13)

To do this, the following two functions are defined:

$$f_{K_{1\pi}}(l, \mu, x) = \left[ s - \alpha \theta^\mu(s_1) \theta^\mu(\sigma) - [s_1 - \alpha \theta^\mu(s_1) \theta^\mu(\sigma)] \cos(x) \right]$$

$$g_{K_{1\pi}}(l, \mu, x) = \left( s^2 - \alpha^2 - 2s_1 \alpha \cos(x) \right) \left[ s^2 + \alpha^2 + (\eta_\mu - \eta_\sigma)^2 - 2s \alpha \cos(x) \right]$$  \hspace{1cm} (14)

The integral is consequently:

$$R_{1\pi}^{\mu}(l, \mu) = \int_0^{2\pi} \cos(\tau x) \frac{f_{K_{1\pi}}(l, \mu, x)}{g_{K_{1\pi}}(l, \mu, x)} dx$$

$$= \frac{2\pi}{n} \sum_{i=0}^{n-1} \cos\left( \frac{\tau \cdot 2\pi i}{n} \right) \frac{f_{K_{1\pi}}(l, \mu, \frac{2\pi i}{n})}{g_{K_{1\pi}}(l, \mu, \frac{2\pi i}{n})}$$  \hspace{1cm} (15)

The advantage within a framework like MAPLE is that such a discretization is done automatically. Numerical problems are hidden from the user. To implement such a function manually means to implement a numerical procedure for all valid parameter combinations. In this case the fraction from Eq. (15) becomes undefined if $l = \mu$, $x = 0$:

$$f_{K_{1\pi}, \mu=0}(l, x) = (s - s_1 \theta^\mu(s_1))^2 (1 - \cos(x))$$

$$g_{K_{1\pi}, \mu=0}(l, x) = 2s_1^2 (1 - \cos(x)) \left[ 2s_1^2 (1 - \cos(x)) + (\eta_\mu - \eta_\sigma)^2 \right]$$  \hspace{1cm} (16)
These two equations become zero for \( \nu = 0 \). This means a direct calculation is impossible. But using L'Hôpital’s rule, this discontinuity can be canceled:

\[
\lim_{\nu \to 0} \frac{f_{k1 \pi \mu = 1}(l, \nu)}{g_{k1 \pi \mu = 1}(l, \nu)} = \lim_{\nu \to 0} \frac{\frac{d}{d\nu} f_{k1 \pi \mu = 1}(l, \nu)}{\frac{d}{d\nu} g_{k1 \pi \mu = 1}(l, \nu)}
\]

After solving these derivations:

\[
\frac{d}{d\nu} f_{k1 \pi \mu = 1}(l, \nu = 0) = (s_l - s_l \phi^* l)^2 \sinh(x) \\
\frac{d}{d\nu} g_{k1 \pi \mu = 1}(l, \nu = 0) = 2s_l^2 \sinh(x) \sqrt{2s_l^2 (1 - \cos(x)) + (y_p^* - y_p)^2} \\
+ \frac{2s_l^2 (1 - \cos(x)) \sinh(x)}{\sqrt{2s_l^2 (1 - \cos(x)) + (y_p^* - y_p)^2}}
\]

Now the result for \( \nu \to 0 \) is:

\[
\lim_{\nu \to 0} \frac{f_{k1 \pi \mu = 1}(l, \nu)}{g_{k1 \pi \mu = 1}(l, \nu)} = \frac{1 - \phi^* (s_l)^2}{2s_l (y_p^* - y_p)^2}
\]

This special case can be used now for the numerical integration of Eq. (15):

\[
K_{k1 \pi}(l, \mu) \approx \frac{1 - \phi^* (s_l)^2}{2s_l (y_p^* - y_p)^2} + \frac{2\pi}{n} \sum_{i=1}^{n-1} \cos \left( \frac{2\pi i}{n} \right) \frac{f_{k1 \pi}(l, \mu, \frac{2\pi i}{n})}{g_{k1 \pi}(l, \mu, \frac{2\pi i}{n})}
\]

This process of transferring the analytic equation into a form which can be implemented in a software code is not difficult from a mathematical point of view. But the handling of long mathematical terms is not easy for humans. At this point, an analytic mathematical tool is helpful, too. Especially, tasks like L'Hôpital’s rule can be handled easily. Complex function can be derived quickly and reliably. Such tools can be used as a mathematical “CAD-Tool”. Error-prone tasks (such as algebraic signs, substitution of variables, etc.) are easily done by the computer. The human is now able to concentrate on the mathematical description of a problem and no longer on writing error-free equations.

5. Evaluation of the computed unsteady circulation

The numerical model was implemented in standard FORTRAN as part of the ship design tool E4. The wake given by Zwick (1962) (Fig.4, page 166) was digitized and used as input for the evaluation. The results are shown in the Figs. 6 and 7. Theoretically the results should be the same, but in practice it is complicated to reproduce calculations done 50 years ago. Much information has been lost over the years. The used information has been taken from small pictures to be digitized. This procedure loses inherently some information and the data is only available at special radial points, Fig. 4. In between, everything has to be interpolated. The real wake used for the former calculation is unknown. The number of original supporting points is unknown, as is the interpolation method and the actual integration scheme.

The implemented method was evaluated with the help of the published circulation. The aim was to reproduce it as good as possible. The results are shown in Fig.6 and Fig.7. In principle, the results should be exactly the same, but the input data is only a small subset of the original wake and after 50 years it is not clear which kind of interpolation and integration scheme were used.
The Fig.5 shows the way of discretization. The original data from Fig. 4 was digitized at fixed supporting points and linearly interpolated, Fig.5 (left). Then a Fourier analysis was used to describe the wake. Fig. 5 shows the meaning of discretization for such a wake.

Especially the hard peak at $\psi = 0$ and $\psi = \pi$ significantly influence the circulation. If this peak is not discretized accurately enough, the results are useless. Fig.5 (right) shows further that the Fourier analysis of such an inhomogeneous wake needs enough higher harmonics. Otherwise the results are difficult. The calculations of the original paper used six harmonics to describe the wake whereas this investigation used eight harmonics.
Fig. 7: Examples of calculated circulation $\Gamma(\psi)$ at $s = 0.76$ (left); $s = 0.94$ (right).

Figs. 6 and 7 show a good reproduction of the original data. The existing differences can be explained with the uncertainties of the available data. It should be pointed out that the input wake was given in Fig. 4 on the supporting points $s = \{0.2; 0.3; 0.5; 0.7; 0.9\}$, but the unsteady circulation $\Gamma(\psi, s)$ for $s = \{0.28; 0.52; 0.76; 0.94\}$. The results were calculated for interpolated supporting points.

6. Conclusion

This paper has described an example how a classical analytic approach for a hydrodynamic theory can be used within a modern computer dominated environment. Classical theories have the great advantage of low requirements in computer power. This is very useful in early design.

The development and the numerical implementation are often not easy. Problems occur if a complex mathematical theory has to be transferred to a computer algorithm. Discontinuities have to be handled with the required accuracy. Issues of supporting points and the mapping of real values on digital floating-point arithmetic have to be handled adequately. Symbolic mathematical tools can support this task efficiently.

References


ISAY, W.-H. (1964), Propellertheorie, Springer Verlag

KRÜGER, S. (2003), The Role of IT in Shipbuilding, 2nd Conf. Computer and IT Applications in the Maritime Industries (COMPIT), Hamburg


Appendix A

Definition of matrix elements from Zwick (1962):

\[
B_{\tau \nu \lambda \kappa} = M(s) \theta_{1 \kappa} + \frac{2\pi}{(k+1)^2} \sum_{\mu=0}^{k} K_{\nu}^\tau(s, \sigma) \theta_{\mu \nu} \sum_{\lambda=1}^{k} \frac{\lambda \sin \left( \frac{\lambda \mu \pi}{k+1} \right) \cos \left( \frac{\lambda \mu \pi}{k+1} \right)}{2(k+1)}
\]

\[
+ \frac{1-s_0}{2(k+1)} \sin \left( \frac{\mu \pi}{k+1} \right) L_\nu(s, \sigma, \nu_\kappa, \lambda_\kappa)
\]

\[
s_1 = \frac{1}{2} \left[ 1 + s_0 - (1-s_0) \cos \left( \frac{\mu \pi}{k+1} \right) \right]
\]

\[
\sigma_\mu = \frac{1}{2} \left[ 1 + s_0 - (1-s_0) \cos \left( \frac{\mu \pi}{k+1} \right) \right]
\]

\[
\theta_{1 \kappa} \left[ \frac{1}{l \neq x}, \frac{1}{l = x} \right]
\]

Further we define:

\[
K_{\nu}^\tau(x, \sigma) = \frac{q}{8\pi^2 \theta^{\nu}(s)} \int_0^{2\pi} \cos(\tau x) \left[ \sigma - s \theta^{\nu}(s) \theta^{\nu}(s) - [s - \sigma \theta^{\nu}(s) \theta^{\nu}(s)] \cos(x) \right] \frac{\cos(x)}{(s^2 - \sigma^2 - 2s\sigma \cos(x))} \sqrt{s^2 + \sigma^2 + (y_\sigma - y_\beta)^2 - 2s\sigma \cos(x)}
\]

\[
- \frac{q}{8\pi \theta^{\nu}(s)} \left\{ \begin{array}{ll}
2 & \tau = 0, \ s < \sigma \\
[1 + \theta^{\nu^2}(s)] \left( \frac{s}{\sigma} \right)^{ |\tau| } & \tau \neq 0, \ s < \sigma \\
-\left[1 + \theta^{\nu^2}(s)\right] \left( \frac{s}{\sigma} \right)^{ |\tau| } & \tau \neq 0, \ s > \sigma
\end{array} \right.
\]

\[
L_{\nu}^\tau(s, \sigma) = \frac{1}{8\pi^2 \theta^{\nu}(s)} \int_0^{2\pi} \frac{\sin(\tau x) \sin(x)}{(s^2 - \sigma^2 - 2s\sigma \cos(x))} \sqrt{s^2 + \sigma^2 + (y_\sigma - y_\beta)^2 - 2s\sigma \cos(x)}
\]

\[
+ \frac{1}{8\pi^2 \sigma^2} \int_0^{2\pi} \frac{\sigma \cos(\tau x)}{\sqrt{s^2 + \sigma^2 + (y_\sigma - y_\beta)^2 - 2s\sigma \cos(x)}} \sin(x)
\]

\[
+ \frac{|\tau| q \left(1 + 2\theta^{\nu^2}(s)\right)}{8\pi \sigma \theta^{\nu^2}(s)} \left\{ \begin{array}{ll}
\left( \frac{s}{\sigma} \right)^{ |\tau| } & s < \sigma \\
\left( \frac{s}{\sigma} \right)^{ |\tau| } & s > \sigma
\end{array} \right.
\]

\[
+ \frac{1}{4\pi} \sum_{\nu = 1}^{q-1} q \sin \left( \frac{2\pi}{q} \nu - 2\beta \right) \theta^{\nu}(s) \frac{s \sin \left( \frac{2\pi}{q} \nu - 2\beta \right)}{\sqrt{s^2 + \sigma^2 + (y_\sigma - y_\beta)^2 - 2s\sigma \cos \left( \frac{2\pi}{q} \nu - 2\beta \right)}}
\]

\[
M(s) = -\frac{q \left(1 + \theta^{\nu^2}(s)\right)}{2\pi \theta^{\nu}(s)}
\]
Simulating the Production of Future Marine Products

Thomas Koch, Atlantec Enterprise Solutions GmbH, Hamburg/Germany, thomas.koch@atlantec-es.com

Abstract

Discrete Event Simulation has recently gained broader interest to improve production analysis, planning, and scheduling and is already applied regularly by some shipyards. However, various potentially interested users consider the heavy-weight modelling effort and experience required a major obstacle to build useful simulation scenarios. This paper describes a domain oriented and knowledge based approach that creates simulation models in a straightforward, descriptive way. It uses real-world objects recognized by engineers and planners. The approach is applied to production facilities as well as to products. Another challenge addressed is the simulation of incompletely designed products for future new-building programs.

1. Introduction

Shipbuilding production usually is a complicated process that requires a lot of individual planning due to its one-of-a-kind nature. Traditionally the planning activity is mostly an empirical procedure, but with the introduction of computerized systems such as planning tools and ERP systems particularly the administrational aspects have been covered increasingly well in an automated or semi-automated way.

Following the full-scale use of CAD systems a trend has developed towards using simulation systems that can model the physical, dynamical behaviour of products being designed. At the same time, various approaches have been made to apply discrete event simulation techniques to production planning and factory design problems. Many of these systems focus on the generic description of processes or, more specifically, on logistics, manufacturing processes, or material flows through factories or warehouses.

In principle it is perfectly possible to model shipyard production operations based on generic principles as has been successfully done in the past. But the one-of-a-kind nature of shipbuilding processes causes various issues. Creating a simulation model based on generic process descriptions and properties takes a considerable effort and includes a wide range of potential configurations. This is acceptable for factory design and layout simulations for example, where the cost of creating the model accounts only for a small part of the total investment and where the involvement of trained experts can be easily afforded. However, when it comes to reflecting actual shipyard configurations and processes on a day to day basis (including capturing changes over time) this effort has been found to be quite high for production planners and engineers. Few shipyards are able to afford specialists focusing on these tasks. However, this issue can be addressed by simplifying the creation of shipyard production models.

Another major problem for tasks related to planning is the lack of precise product information required to execute reasonably reliable simulation runs at early project stages. Trying to forecast production of a future project at an early point in time poses a problem due to unavailable or unstable design information. It would therefore help to be able to make reasonable assumptions about the product as soon as possible.

2. Discrete Event Simulation

The discrete event simulation method is a time-based approach to simulation. A certain section of a real world set of objects is represented in a simulation model in the form of individual entities. They are each represented as set of properties that form the state of an entity at a certain point in time.
During a simulation run, events will cause changes to occur to those states in a chronological order. The change of state in turn may trigger the occurrence of new events. The fact that state changes will only happen in reaction to events taking place is a characteristic of a discrete approach. For example, continuous changes like movements are not represented, but the change of position will be established at some future point in time, e.g. by an arrival event. While this might not be a good approach to study kinematics at a high resolution, it is well suited for problems, in which the focus is on the resulting state of an operation and the related duration of actions.

There exist a number of different ways to implement discrete simulation methods. Commonly described in literature are the event-based approach, the process-based approach, the activity-based approach, and the three-phase approach Kreuzer (1986), Pidd (2004).

The fundamental components of a simulation model to be used for a discrete event environment are entities, processes/activities, and events. Eventually everything in the real world needs to be mapped to these generic fundamental items. Once this is done, a broad range of commercial and open source simulation engines is available to execute simulation runs quite effectively. The continued advances in both hardware and software technology are allowing simulation users to reach even higher levels of performance with regard to speed and execution. Parallelization is just one example, which has gained a lot of interest since the wide-scale introduction of multi-core CPUs, Fujimoto (2000). It thus becomes more feasible to apply simulation technology to a broader range of tasks, to handle larger models, and to investigate many more alternatives.

3. Typical simulation tasks for Production

The range of simulation tasks applicable to shipbuilding production is large. The typical goals that a shipyard might have in performing simulations include:

- Maximize production throughput
- Minimize production times
- Alleviate bottlenecks in workflows
- Maintain and level loads on facilities and resources
- Determine optimal production sequences
- Minimize production risks particularly for new products or new production technologies
- Evaluate impact of changes (design, planning, material) on production
- Compare design or production alternatives

Fig. 1 provides an overview of the typical application areas in which simulation can help to improve performance or to verify feasibility.
To support/improve the traditional planning activities, typical applications for the planning & scheduling area would include: validation of proposed schedules; determination of requirements to meet given schedules; or identification of changes in resource requirements to allow level loading of facilities or balancing of resource levels. Another dimension is added by considering short-term, mid-term, and long-term perspectives which will lead to different levels of detail both in input and observed output data.

Another aspect of planning involves capacity analysis. Simulation can help to optimize utilization of facilities and resources either by developing a specific forecast of resource and transportation requirements, or by identifying bottlenecks.

A quite different approach must be employed for bids and proposals. Detailed planning activities typically utilize detailed design and production preparation information. However when simulation is applied to an area like bids and proposals, the analysis will have the characteristics of a long-term forecast. At this stage usually only incomplete information is available, many alternatives may need to be investigated, and the cost of producing a simulation result must be quite low. In this phase shipyards would like to be able to validate and refine work content and resource requirements. Often, they would also like to evaluate configurations to minimize impacts on their current build program, *LSMS* (2010).

Process analysis and optimization applications have yet another set of characteristics. They may range from complete investigations concerning full or partial factory or facility reorganization to analysis of individual machine operation. They may also deal with analysis of detailed processing sequences that may be optimized for existing equipment or require only minimal modifications. This can result in quite detailed descriptions of production equipment and related processes. When experimenting with new processes or layouts, simulation can also help to identify potential deadlocks or analyse why certain throughput problems have occurred in the past and how to resolve them.

Closely related to that are questions about manufacturability which will validate task sequences for production or material ordering and delivery. It often involves analysis of material flows or determination of transportation and storage requirements. The study of transportation is a whole area in its own right. For example, the transportation of large assembly units requires careful planning and a lot of experience in both scheduling and technical matters.

![Fig. 2: Simulation data model](image-url)
When looking at offshore technology, new ship types, and other areas of a rapid development, simulation can also be a powerful tool to mitigate the risks of working with unusual/innovative products.

4. Components of the Simulation Model

In section 0 we discussed the fundamental structure of a generic simulation model. It is quite clear that such a model is difficult to handle by users who are experts in their application domain but who do not have the time to investigate those theoretical concepts and would need to spend a considerable amount of time thinking about a mapping between those two worlds. As importantly, the time needed to create such models is often prohibitive. This has been recognised by a number of project teams and has triggered development activities to bring modelling to a higher level, see e.g. Steinhauer (2010).

The approach that we decided to adopt was to go back and consider the problem from the pure “end-user” perspective. What is the equivalent high-level description that is easier to describe and maintain? The most natural way of describing a shipyard is, of course, to describe the shipyard in real world terms. The goal should be to get as concrete as possible when defining the production facilities. The same principle applies to the product(s) to be manufactured and the processes applied during production. At the same time, the required level of detail should match the requirements of the simulation task at hand. Configuring actual simulation scenarios and simulation cases can then be performed by “cherry picking” from those model elements and adjusting the desired parameters, Fig. 3.

Fig. 3: Example of machine properties in the facility model

5. Setting up a Facility Model

To accomplish these goals for the shipyard model definition, we use a facility modelling application, which enables users to define a shipyard layout and facility inventory in an expedient way. The model that is created by using this application is constructed from these entities:
This approach is quite straightforward when it comes to “shrink-wrapped” types of machines, work centres, or transport equipment items. By providing a broad range of shipyard production equipment items, the user just plugs the right equipment type into the model at the desired location and configures any non-standard performance parameters (some which are specific to the equipment type). Fig. 4 shows an example of an input for plate cutting equipment.

To further simplify input of the physical facility model – whether it is a complete shipyard, a factory, or a production line – a geographic layout can be used as a starting point. This may be retrieved from a map web service, an aerial view, or a drawing and gets loaded into an interactive editor component as a scale-observing background image. Fig. 4 shows a sample shipyard layout created using this editor component.

For every facility, the applicable work flow can either be taken from a library of standard work-flows or is defined by means of a process description to the desired level of detail configured from individual activities. For straightforward activity sequences, a tabular input is sufficient and allows rapid input; however more complex processing is easier to model by means of a graph structure. A key concept in this context is the definition of so-called fabrication methods, which follows the concept of production methods described in Hildebrandt and Koch (2006). These method descriptions
derive – via a parametric rule set – production performance data expected for a certain production method or activity from their operational parameters and/or product properties. For example, the work content of some item to be produced can be derived from a set of individual product parameters like welding length, surface area, weight, volume, material input, data from part lists, or properties of the resulting product; furthermore they may use any attributes related to the location, facility, transport system, or staff resources being involved, up to the shipyard level. Apart from database queries, rules usually also involve deterministic and/or stochastic functions. Depending on the referenced properties, evaluation of these rules may occur either at model configuration time (providing data that is static) or during simulation runs (for dynamic data). It is evident that the content of such rule sets can get somewhat sophisticated, but for many tasks quite simple calculations will be sufficient and yet powerful and the purpose of calculation is always explicit. Other projects have used a similar concept, for example to determine process durations, Wanner et al. (2010).

6. Providing Product Data

Product data is the second major group of entities that must be represented in a model. The level of precision depends on the goal of the actual simulation experiment. However, a common characteristic is the need for property information of parts, intermediate production results, and final products. This may include dimensions ($l \times b \times h$, mass), more or less detailed geometric shape, physical properties (material, centre of gravity), and a broad range of production related parameters like area, volume, welding lengths, no. of connections and type, etc. On a more detailed level, the actual 3D shape of items may be required but also data about individual design features may be useful. The more complete this information will be, the better e.g. estimates of actual work content will be.

![Fig. 5: processing imported CAD design data](image)

All product related data, in an ideal scenario, is readily available from detail design systems concerning the geometry or physical properties. Nevertheless this requires good integration with legacy – generally CAD systems – in many cases a mix of these systems will contribute to the overall model. In our application environment we can rely on the Topgallant® Infrastructure adapters to connect to commonly used CAD systems and to deliver high quality technical content for further processing, AES (2009), Fig. 5.
For large scale simulations, interaction between production activities related to different products is relevant. For example, to investigate a 5 year build programme of a sizable shipyard operating at two major locations and with a number of subcontractors can easily involve between 10-20+ ships.

To address product data availability issues, we have developed an application that is supporting the generation of high fidelity product data: the virtual product generator. This tool generates “virtual” product data of the same kind and structure as “real” product data used for simulation purposes, but the data will be adjusted to the characteristics of the new future product. The generator is configured by using a template library for generation of parts and intermediate products. The templates hold fully parameterised rule sets containing specifications about physical and administrational properties of the desired product items. These rules can be defined using deterministic values or stochastic distributions which enable randomisation of the data within defined value ranges, Fig. 6.

New types of ships may be supported by using existing templates for portions of the product to which they are applicable, while new types of intermediate products can be introduced by defining the appropriate new templates. Due to the nature of the rules, defining a template is a descriptive activity providing data about expected counts and properties of parts and intermediate products. If desired, the generator can execute actions to issue the corresponding work breakdown structure.

As detailed design progresses on a new type of product, virtual data can be replaced by real data. Comparison of simulation results using this data will provide feedback about the reliability of the initial assumptions.

Fig. 6: Example of a template for virtual product data

7. Plugging it together via Processes

Finally, a work breakdown structure and the associated work content are needed for a full specification of the model. For products that are already fully planned, this information is available from planning and scheduling systems or production engineering applications as schedules and work orders. Thus the minimum requirement is to be able to pick up such data from existing data bases or system interfaces.
As is the case for product data, complications arise when the product design and planning has not yet advanced to these stages, at least partially. Due to the highly compressed design schedule, simulation may need to be performed while a lot of such planning data cannot be provided.

Unfortunately, for various types of simulation experiments, particularly for long-term projection cases, design will not have advanced very much (yet). This forces the user to either use extremely simplified and generalised estimated values on a very coarse level, or it would theoretically require a lot of input, which is unrealistic. As a compromise, data may be copied from previously used product data sources (e.g. for sister ships), but this still causes a lot of work to be spent in making adjustments (particularly when trying to use the data for products that differ in type and/or general particulars) and in the end the quality of results will be unsatisfactory and conclusions must be considered unreliable.

We are employing the Topgallant® Assembly Production application in four task areas to generate missing scheduling and sequencing data based on the use of custom strategy rules sets:

- Work breakdown structure generation
- Assembly sequence generation
- Work content estimation
- Hull erection schedule generation

Any of these steps might be skipped during data preparation if the appropriate information is directly available from an external data source. In that case a corresponding data import will be performed via an appropriate adapter.

The process data preparation starts with the automated generation of a work breakdown structure (WBS). Utilising a fully configurable set of decision rules, assemblies (representing intermediate production results) are defined from existing product parts. Any property for a part (and its environment: functional structures, compartments, zones, connectivity, etc.) can be used in this process to determine the resulting structure. The method can be applied top-down, bottom-up, or as a combination of both. In many cases, the set of the topmost assembly units will be available as a fundamental planning input, which will provide further guidance in this process. At the end, a completely defined WBS will be available. For each assembly item, this will provide access to typical parts list information, total weight, centre of gravity, dimensions, footprint, and similar data. Fig. 7 shows the resulting WBS produced for an imported design structure.
Additionally, production sequences like mounting sequences within assemblies are required. This information is also often missing. Using a selectable strategy-based approach, the default assembly sequence can be automatically proposed by the system. The strategy rules have access to part and assembly information and can thus consider important factors like type of item, weight, mounting or welding requirements, need for fitting work etc.

Work content data is an essential input for most simulation experiments. Pre-calculated work content will be used for activities that are not further broken down into individual (sub-)activities are therefore closely related to the level of detail to be applied. In other cases, work content will be evaluated during the simulation, possibly taking into account actual conditions such as environmental information or introducing stochastic behaviour.

![Image of generated hull erection schedule based on WBS and work content](image)

**Fig. 8**: Generated hull erection schedule based on WBS and work content

Again, the main issue here is early availability and sufficient accuracy of such data. Using the fabrication method approach mentioned before, it is possible to estimate work content precisely based on product properties and production method used, either as deterministic static values before or dynamic, possibly stochastic values during simulation.

The final step consists of defining the hull erection schedule. Fig. 8 shows a grand block structure and the assembly erection sequence derived from it. The actual conditions for stacking or attaching blocks are fully configurable and can be made to consider the function and other properties of the intermediate grand block product. The sequence also provides target milestone dates for preceding production levels based on the estimated work content. The result can be used as a “skeleton” target schedule to drive simulations, e.g. by issuing the corresponding work orders.
8. Configuring and Running Simulation Cases

A simulation scenario is considered to be the high level configuration for a simulation task. It involves selecting all the data stores to be considered. In our simulation modelling environment, a scenario effectively acts like an inventory of all model items that may play a role in the simulation. The detailed configuration occurs on the simulation case level. This includes selecting /deselecting individual items or assigning life cycle information to those items (for example, if some machine shall be phased in at some point in time).

Simulation can only be initiated with some stimulus data. For production type simulations e.g. a set of work orders related to manufacture of products (at any level) or delivery events for material deliveries may be used. During case configuration, a variety of stimuli generators can be selected and configured.

Finally, some fundamental execution parameters will be set, e.g. calendar time, start and stop time, deadlock detection rules, etc.

The simulation runtime architecture includes a number of “administrational” entities representing planning and decision-making tasks. Jobs need to be scheduled, transports need to be organized, material is being requested, and process execution must be monitored. The chosen architecture reflects the necessary administrational structures. All these planning and decision-making tasks however are built on some sort of strategy. For example, choosing the next available work place for some task could be performed on one extreme as a pure random selection among the available items or it could on the other end be done in an entirely deterministic way. Between those extremes a number of other mechanisms might be applicable. This requirement is addressed by providing different strategies as pluggable modules that can be selected at model configuration time. Strategies can reflect anything from the classical theoretical concepts like first-come/first-served, last-in/first-out for handling of queuing problems, or random assignments using distributions to dispatching based on using weighted preferences and/or decision rules.

To configure the strategies, the desired strategy type is selected for the target controller function. Depending on the strategy chosen, a corresponding selection of configurable parameters can be adjusted according to the individual needs.

Pluggable strategies offer a lot of features for experimentation. When working with technically innovative products, it is often somewhat unclear whether traditional approaches will work as expected. Running various simulations based on different or modified strategies will provide a lot of what-if analysis capabilities.

Executing a single simulation is only one single step in searching for a solution of a problem. Under normal conditions, analysis of results will lead to a modification of some input parameters to create a new experiment. This can typically result in a large number of different runs, some of which may only differ by small details while other runs will reflect substantial modifications e.g. of the production facilities, resource constraints, scheduling, etc. In order to help users manage the variety of possibilities, hierarchical experiment storage is being used. By grouping runs into scenarios, experiments, and multiple levels of cases, some organizational guidance is provided. For every case, references to all input data, configuration data, and output data are persistently stored. Any input configuration can be visualized again and re-run or be used as the starting point for another variation. Any output data set can be re-analysed again and used for comparative studies.

9. Conclusions and possible future directions

Our recent work and experience gained with simulation projects has led to the development of a real-world modelling environment for one-of-a-kind production applications. By enabling users to work with representations of concrete objects, the requirement modelling effort has become quite
manageable. However, when it comes to new types of products or long-term projects of products not yet designed in detail, the automated generation of input data becomes the real challenge. The approach chosen for our solution seems to be very promising in this respect.

For future work, we see a number of interesting directions. For example, advanced support methods for the definition of templates for virtual products will further expand this capability. Or, massively parallelising simulation execution will further progress with the goal to enable users to investigate much broader variations of cases. This of course will make optimisation tasks more practical even for day to day operations.

References


Cost Effective Autonomous Robots for Ballast Water Tank Inspection

Leif Christensen, DFKI GmbH, Bremen/Germany, leif.christensen@dfki.de
Niklas Fischer, BALance TC GmbH, Bremen/Germany, niklas.fischer@bal.eu
Sven Kroffke, DFKI GmbH, Bremen/Germany, sven.kroffke@dfki.de
Johannes Lemburg, DFKI GmbH, Bremen/Germany, johannes.lemburg@dfki.de
Reinhard Ahlers, BALance TC GmbH, Bremen/Germany, reinhard.ahlers@bal.eu

Abstract

In this paper we describe the technical concept and first implementation of an autonomous rail guided robot for inspection and maintenance tasks in ballast water tanks (BWTs), especially for double bottom BWTs. We evaluate different locomotion techniques and other robotic principles potentially suited for such application scenarios and propose a modular concept that uses a simple corrosion-free thermoplastic rail which runs ‘through’ the robot and thus allowing for high payloads and upside-down operation. As a second aspect the paper describes the assessment of life cycle benefits for the developed system with cost parameters like scaffolding costs, tank preparation-, in-tank inspection- and vessel out-of-service time.

1. Introduction

Ballast water is used to stabilize partially loaded or empty ships on the open sea. When needed, ocean or port water is pumped into special ballast water tanks to increase the mass of a ship. Because ocean water is typically contaminated with algae, plankton and other organisms, and due to the aggressive nature of salt water, BWTs are often subject to serious fouling and corrosion. The ballast tanks represent more than 40% of the entire coated area on a vessel. They therefore need frequent inspection, cleaning, and repair. Until now, the maintenance of BWTs cannot be performed under operating conditions. Consequently, ship owners are forced to send their ships to dock inspection every 2.5 or 5 years, causing high costs in labour and ship-downtime. On cruise vessels BWTs are typically built in those spaces of a ship that cannot be used for other purposes. As a consequence, they are narrow, irregularly shaped, and badly ventilated. They are designed to be accessible to human workers, but are surely a dirty, unhealthy, and unpleasant workplace, Fig. 1 and Fig. 2.

Fig. 1: Double-bottom BWT of a cruise liner at Meyer Werft Shipyards (construction phase)

Manual work in a BWT is a tedious task that carries potential short- and long-term health risks for the workers involved. Nevertheless, coating, cleaning, inspection and repair of ballast water tanks is still done manually. An automation of these processes is not feasible so-far due to the complexity and the
variability of current tank-design, as well as the limited flexibility of the available robotic systems. If robots were to take over the inspection, cleaning, coating, and repair of ballast water and other tanks and narrow spaces, this would greatly speed up the process and reduce the health risks for workers and ship crews. Using robots, ship builders, ship owners and class societies alike would be able to considerably reduce the cost for inspection and maintenance of BWTs.

The international R&D project ROT (RObots in Tanks) addresses these challenges by evaluating available robotic principles potentially suited for BWT inspection (described in section 2), developing a concept for a BWT inspection robot and implement the concept in a robotic demonstrator (section 3). The whole development is accompanied in the project with a life cycle cost analysis of such a robotic system and the first findings on that are shown in section 4. Section 5 concludes the work done so far and gives an outlook for next possible steps.

2. General robotic concepts potentially suited for BWT inspection

Although the inspection and maintenance of BWTs today is done manually, there are generic robotic construction principles, systems and system components, which have to be evaluated to find a starting point for a design of a robot potentially suited for the specific application scenario in BWTs. Main robotic design parameters (locomotion, sensors, payload, communication, autonomy / behavior architecture and power supply were examined with regard to the user requirements, the design of ballast water tanks in general and in particular (crude carrier BWTs, double bottom BWTs), their typical state during/after construction as well as their typical state in use. Since none of the robotic design parameters mentioned above can be considered isolated, we show the cross-connections between and the impact on them, when choosing one technology over the other.

2.1 Locomotion and adhesion concepts

Apart from wheel-driving robots in a perfectly clean and unobstructed office- or laboratory-environment, the task of moving in a certain direction or (more complex, because it has a navigational part) getting from point A to point B is a hard task. This holds true even more if the robot has to move in 3 dimensions, often resulting in up to 6 degrees of freedom. The complexity of the movement (and therefore control) highly depends on the layout of the robot, for example thinking of a six-legged robot with 4 joints for each leg like the robot “SpaceClimber”, Bartsch et al. (2010), trying to coordinate these 24 joints to do a smooth seemingly simple right hand forward turn walking pattern. Nearly all classical locomotion methods like walking, frame walking, sliding, wheeled, hybrid, tracked and brachiating (arms with grippers) methodologies have been proposed for climbing robots, but there usability highly depends on the typology and the application at hand. For a more detailed overview see Longo et al. (2008) and Silva et al. (2008). In the very demanding confined space of BWTs, usually often cluttered with stiffeners and frames, movement requires adhesion concepts adequate for the movement task. Flying or diving in a flooded environment would not need an adhesion concept, but due to the very narrow confined spaces in double bottom ballast water tanks and to payload limitations both concepts obviously can’t satisfy user requirements and therefore are left out in the following examination. Adhesion techniques for climbing robots can be classified with respect to the nature of the forces needed to remain attached to the wall:
2.1.1 Pneumatic

A frequent approach is to utilize suction force. Usually, more than one vacuum cup is used in each foot in order to prevent loss of pressure due to surface irregularities (Longo et al. (2008)). Since pneumatic techniques do not only have a very high power consumption, but also need a smooth and clean surface, they would have been a minor option (considering other drawbacks like the slowness due to the time it takes to develop enough vacuum) for newly build ballast water tanks, but are out of the question for a robot that has to inspect ballast water tanks that were in use for some time. Please note that this conclusion may change vastly in the future, due to new ballast water exchange regulations like the IMO Convention “International Convention for the Control and Management of Ships Ballast Water & Sediments”, which will reduce the amount of sediments in ballast water tanks significantly, IMO (2005).

2.1.2 Magnetic

Nearly all surfaces in ballast water tanks are made of ferromagnetic steel, with very rare exceptions, where stainless steel is used. Magnetic adhesion can easily generate really strong forces using rare-earth permanent magnets (e.g. aluminum-nickel-cobalt, samarium-cobalt or neodymium-iron-boron) or electro-magnets, so this adhesion concept looks quite promising. Advantages of the electromagnets are mainly the flexibility to switch them on and off as needed, for example when used in a foot, which has to change from stance to swing-phase. The main disadvantages of electromagnets compared to permanent magnets are the high currents needed as well as the bigger construction space. The main challenge when using permanent magnets is to get them off the surface again when needed. For example, the “Magnetbike” robot, Tache et al. (2009), uses small extra actuated wheels for that purpose, whereas the DFKI magnet crawler uses a flexible passive peeling mechanism to get the small permanent magnets of the wall again, Vögele et al. (2010).

Fig. 3: MINOAS Crawler, Robot Magnetbike

Despite of the advantages of strong generated forces, the magnetic forces have roughly quadratic loss with increasing distance to the ferromagnetic surface, so coating, rust or dents in the surface lead very fast to the robot loosing the adhesion. Another problem in a long term application is the accumulation of ferrous dust, which again leads to a higher distance to the ferromagnetic surface and thus a significant loss in adhesion. Nonetheless, depending on the robotic task and the surface at hand, magnetic adhesion can be considered a valid adhesion concept option for ship inspection purposes. A magnetic crawler may be a solution for remotely operated spot inspection, where a surveyor uses the robot as a tool to be able to take local measurements, where means of access are insufficient and he could not reach. However, in our opinion this solution fits better to large ballast water tanks or loading areas, as they occur in large bulk carriers, tankers or container ships than to the confined small ballast water tanks of cruise liners.

2.1.3 Mechanical adhesion

Mechanical adhesion is a low-power-consuming possibility to remain attached to a surface in the case of structured surfaces and specialized grippers. The main disadvantage of those solutions is the challenging task of getting to the point, where a grip is possible, both from a locomotion and...
navigational point of view. These solutions tend to be very slow while consuming a lot of power due to high lever arm forces. In the situation at hand, the need to have a kind of small robot to fit through manholes combined with the large distance between possible gripping points (e.g. “Holland” profile stiffeners) without changing the environment renders this solution useless for our inspection task.

2.1.4 Chemical and electrostatic adhesion

Other adhesion forces, that were proposed for climbing robots are chemical and electrostatic forces, however the adhesive force tends to decrease rapidly after a few sequences in case of the chemical (glue) technique, due to the unavoidable presence of dust, while the utilization of van-der-waals electrostatic forces like in the robot “Stickybot”, Kim et al. (2007), looks promising, but seems today quite far away from actual applications in real world scenarios like the BWT inspection.

2.1.5 Rail guided system

A rail-based solution has two main functions: on the one hand the robots movement between two locations (function 1) and on the other hand the positioning at pre-defined locations (function 2). To fulfill function 1, three sub functions have to be considered: guidance, drive and navigation. The confined and fragmented space within the ballast water tank influences the positioning of the robot (function 2) in that way, that a simple two-dimensional rail system is not able to make the robot see or reach every region of interest. Payload items as touching measurement devices restrict the positioning even more to a fully three-dimensional positioning. Rail-systems to fulfill these demands have to be of form-fit, because the expected weight of the robotic system cannot be handled safe in an upside-down or vertical position by mere adhesion. The precision of the positioning at a given point is less critical for visual inspection of the scenery than for manipulation tasks. The positioning along the rail is dependent on the sub function navigation, whilst the radial positioning tolerances just depend on the precision of the rail and the carriage of the robot. The systematic evaluation of different cross section shapes, which are varied from simple circular up to free form shapes including already used Holland profiles, Fig. 4, leads to the choice of a rectangular cross section for the rail. This shape combines the best combination of affordable manufacturing, the ability to be bend and twisted and the feature to transmit the forces and torques of the robot. The special shape of the Holland profile that is used as stiffeners within the BWT is in principle sufficient for carrying the robot, but needs larger bending and twisting radii than it is required by the given trajectory. To use the straight stiffeners within the BWT as part of the rail system is not feasible, because their openings in the bulk heads are too small for the robot and must not be enlarged due to strength considerations and existing classification rules. The autonomous change of the robot from one stiffener to another would cause uneconomic demands to its morphology.

![Different possible rail shapes including already installed “Holland” stiffener profiles](image)

Main function 1, the movement of the robot, is also mainly influenced by the design of the ballast water tank. To reach all points of interest within the tanks, a three dimensional trajectory has to be
followed. As the rail has to channel the load of the robot to the ship's structure, it has to be mounted to the BWT's wall. To follow a 3D trajectory and to be statically defined within the normal plane of the rail for positioning purposes implies to twist and bend the rail with a preferable small bending radius, as shown in Fig. 10. To drive the robot along the rail, their contact has to transmit the driving torque. With an affordable manufacturing in mind a form-fit as used in rack-and-pinion railways is less efficient as a frictional-connection with an extrudable rail cross-section. The friction coefficient of the paired contact surfaces shall be as high as possible, because it will in use be reduced by moisture, dirt and surface wear. The question of form-fit versus friction has also to be answered for the function of navigation, that imposes the requirement of less slippage and as high as possible precision for the dead reckoning along the track. With the manufacturing cost as an overall requirement, the rail has to feature a simple geometry and therefore a constant cross-section as used in extruded plastic parts. This choice of material also fits the need of corrosion resistance, even under influence of wear and ageing of the material.

Summarizing, a rail guided inspection robot allows for high payloads, is expected to be much faster than other locomotion principles and reduces the navigational complexity to a minimum of 1 degree of freedom, allowing for complete autonomous task execution in the first place, with all the implications stated above. Among others, the reduced DOF results in a clear definition of the location of the robot inside the tank. Aside from that, this option fulfills security requirements concerning the reliability and danger of falling off, that other solutions don’t offer (no security ropes needed, etc.). The main point against this solution is the necessity to install a 3D rail system in the ballast water tanks, but this is an effort that has to be taken only once. Furthermore, means of access had to be installed in the past, too, to allow human surveyors the inspection on site. Nonetheless, the financial aspect is further investigated in section 4. It has also to be taken into account that the rail guided solution has the highest potential in comparison to other solutions to be enhanced for underwater or partly flooded operation inside ballast water tanks while the ship is at sea.

![Fig. 5: Virtual rail guided inspection system in operation inside a double bottom BWT](image)

### 2.2 Sensors

Possible sensors that have been evaluated for the ballast water tank inspection task can be divided in external as well as internal sensors. Internal sensors give feedback about the internal state of the robot or its components and are necessary for the basic operation of the robot. To this group belong the following sensors: battery management sensors, temperature sensors, motor current sensors, force / torque sensors to monitor load on manipulators and motors, brown-out detection and status sensors for different components (heartbeat-signals, watchdog). These sensors are more or less independent of the robots environment and task. The external sensors can again be subdivided into sensors, that are required for the robot itself, for example localization sensors to be able to navigate (to get to the point, *where* a task has to be executed), and those external sensors, that are purely related to the task that should be fulfilled (*what* to do there). Both groups highly depend on the robots mission environment and, naturally, the task to fulfill.
Since the environment in which our ballast water tank inspection robot will navigate is known from CAD data, the distance data provided by a TOF camera could be used to estimate the position of the robot via certain filtering techniques (e.g. particle filters, Kalman filter, self localization and mapping (SLAM)-procedures). While testing a SwissRanger SR4000 PMD/TOF camera we noticed 2 constellations, where distance measurement artifacts appeared: a) when measuring through manholes, the edges with clear space behind are giving false distance information and b) at a distance closer than 50 cm in front of flat surfaces there is extinction in the center, Fig. 5. Also, such a camera consumes a noticeable amount of power (~ 10W) and needs cooling when used for a longer period of time.

Like the PMD camera, a laser scanner produces distance data. This distance data is generally more accurate than the data generated by the PMD camera sensors, but while the PMD camera puts out distance data for the whole field of view, the laser scanner just emits one beam. In most types of laser scanners in use in robotic systems, this beam is rotating, thus generating distance data for a line, which is also the modus operandi of the Hokuyo UTM-30LX we tested inside our testing tank. In these tests we noticed no artifacts specially related to BWTs compared to other environments, but this may change with different coatings. If there is the need for distance date in two dimensions, the laser usually is mounted on top of a pan-unit to do a raster measurement. Due to these rotating or moving parts which also have to be actuated, the higher accuracy comes at the price of higher complexity and size compared to the TOF cameras.

To provide orientation data of the robot pose in our robotic demonstrator, a miniature IMU XSENS MTi was integrated. The MTi is a miniature, gyro-enhanced Attitude and Heading Reference System (AHRS). Its internal low-power signal processor provides drift-free 3D orientation as well as calibrated 3D acceleration, 3D rate of turn and 3D earth-magnetic field data. The magnetic field values showed strong anomalies as expected due to magnetic shipbuilding steel the BWT consists of. Nonetheless, these anomalies are static, and therefore can be compensated and furthermore be used to build a magnetic map of the rail trajectory for localization purposes. In most application scenarios, the need to know the robots orientation and accelerations will always be existent (even in a rail guided system), so such a sensor will necessarily be incorporated in a possible inspection robot.

Regarding task driven sensors, there are a variety of sensors related to ballast water tank inspection during the building and maintenance phase, with large difference concerning size, power consumption and manageability. Most of these sensors need a manipulator to place the sensor at the measurement point, which leads to very high power consumptions in comparison with the rest of the robot. To demonstrate task driven measurements with our BWT Inspection robot demonstrator, we therefore focused on a thickness- and a roughness measurement tool as well as an oxygen monitoring sensor as shown in Fig. 10. For visual inspection, a high resolution camera combined with a pan tilt unit and external flash will be used. A wide angle camera lens will be used to generate full coverage panorama pictures of the ballast water tank surfaces to be inspected or to extract wall-specific textures from them, which later can be mapped onto CAD data.
2.3 Communication

If the ballast water tank inspection robot cannot act totally autonomously, there is the need for at least punctual communication for teleoperation or sensor data exchange. Even if the robot acts on its own, depending on the degree of autonomy, there is often the need to get status information from the robot to a supervisor. Some of the evaluated sensors require high bandwidth for their real time data. For example the Prosilica GE2450C can generate images with a resolution of 2448 x 2050 pixels with 15.1 frames per second, resp. 1900 x 1080 (Full HD) with 24.6 frames per second. This results in a 90% load factor for 1000BASE-T (Gigabit-Ethernet). Anyhow, the main drawback of utilizing Ethernet-based communication is always the need for a cable from the master control station to the robot. At least for the full coverage of a BWT, the use of an umbilical (communication and power supply in one cable) is not an option, due to corners, entwining loops, etc. Regarding the narrow and confined spaces of ballast water tanks, wireless communication is an option to take into account, mainly Bluetooth and WLAN Standards due to their relatively high data rates (for example up to 450 MBit/s for IEEE 802.11 draft n). But even in perfect transmission circumstances, the video data would have to be scaled down, but this would contradict the wish to get the best possible visual impression of the ballast water tank state for the surveyor. Additionally, since the ballast water tanks are made of ship construction steel and very confined, we expect wireless communication showing very high error rates, reducing the bandwidth even more. Regarding the limited possibilities using high bandwidth communication technologies, this leads to the conclusion, that the robot has to be able to collect the data autonomously and store it locally during an inspection mission, or limit the data to be transferred to very basic status and control messages (e.g. robot stuck, percent of task finished, start or stop task).

2.4 Autonomy / Behavior Architecture

As stated above, the limitations concerning broadband communication channels either implies a radio- or cable-controlled ballast water tank inspection robot operating locally only or a robot, that can execute its tasks on its own and therefore has no need of communicating with the outside. The latter naturally implies a lot of artificial intelligence going into such a system, depending on the complexity of the task to fulfill and the complexity of locomotion and navigation. Normally, the high complexity of ballast water tanks, which are very obstructed and confined as well as polluted with partly dried sediments, would make it nearly impossible to design an autonomous system, which can move through a complete ballast water tank on its own while executing its task. But since we reduce the complexity of the navigational task in case of the rail guided system, we can design an autonomous robotic demonstrator, which only needs human interaction at the beginning and the end of an inspection task, while at the same time reducing the amount of sensors needed to navigate.

Usually autonomous mobile robotic systems are classified in two groups: in deliberative / planning robots and those robots, which utilize mainly reactive behaviors. However, the task at hand, the inspection of a ballast water tank with the necessity to take different measurements at predefined spots require in our situation a hybrid solution, where human expertise of ship surveyors is used to define points of interest for the measurements, while the robot has to have certain capabilities to react to unforeseen hindrance like a blocked rail by dried mud. The classification of the state of the tank is again in the hand of a human.
2.5 Power Supply

As stated before, the use of an umbilical for power supply is not an option due to corners, entwining loops, etc., so an on board power supply has to be chosen. Considering available rechargeable battery technologies, the highest gravimetrical energy density is offered by lithium polymer (LiPo) or lithium iron polymer (LiFePo) batteries, Fig. 8, Tarascon et al. (2001).

Aside of their key advantage (high energy density), lithium-ion batteries come in a wide variety of shapes and sizes so as to efficiently fit the devices they power. Furthermore, lithium-ion batteries do not suffer from the memory effect. They also have a low self-discharge rate compared to common nickel metal hydride and NiMH batteries. Lithium ion batteries however are more sensitive against mistreatment (deep discharge, overcharging), which can render the battery pack non-functional, so additional specialized equipment has to be used (e.g. cell balancer during charging). Most of the risks that were associated with lithium based batteries caused by mechanical, chemical or thermal damage possibly leading to ignition are eliminated in recent developments using ceramic separators. Also the fact, that most of today’s consumer electronics use LiPos underlines the fact, that this technology is well understood and usable. Given the rough estimation of the robots power consumption of 45 W, this would lead to an operational period of ~2.9 h using a 5 Ah LiPo battery pack, highly depending on the route of the rail the robot operates on.
3. First implementation of proposed concept

Summarizing the findings of section 2, we propose a rail guided solution for a ballast water tank inspection robot, with the following characteristics:

- **Locomotion**: rail guided system
- **Sensors**:
  - Internal: force / current sensor, battery monitor
  - Navigational: just IMU, odometer
  - Task driven: thickness measurement and high resolution camera with flash
- **Communication**: punctual high bandwidth (start, end), permanent low bandwidth
- **Behavior**: use pre-existing human knowledge about BWT, autonomous onboard plan execution with reactive components
- **Power Supply**: rechargeable battery LiPo or LiFePo sealed

3.1. Hardware

The first carriage design for a rail guided inspection demonstrator consisted of two pulleys and a T-shaped stainless steel profile. But since the operation of a robot on stainless steel could generate small amounts of abrasion that could lead to contact corrosion, we then decided to go for composite plastic as material for the rails. However, this solution would require forming the rail parts externally before installation and thus adding a huge amount to the one-time installation costs, so the decision was made to use thermoplastics. The finally chosen material can be adapted in-place using thermal techniques, but is less sturdy, so the dimensions of the rail had to be expanded. Based on the possible rail material and the estimated size of the robot, the maximum installation space, bending radius and torsion were computed as shown in Fig. 10.

![Fig. 10: Maximum installation space, bending radius and torsion](image)

To avoid big clearance of the demonstrator in conjunction with a rotation point above the rail, we developed a modular frame based concept, where each functional unit (gear train, sensor module, power supply) has its own carriage with the frame allowing for a rotational point in the middle of the rail. The modular concept has the advantage, that more modules can easily be added, if the application requires it, e.g. a second motor-module if the payload gets heavier. Additionally, the drive train modules of the demonstrator where equipped with a snap on mechanism to allow the inspection robot to be placed anywhere on the rail. The propelling wheel is mounted onto a motion link and pressed onto the rail with two compression springs of max. 65 N, typical 50 N spring force each, Fig. 11.

![Fig. 11: Propelling wheel](image)

For the drive-train module, several wheels were manufactured with different material and structure as shown in Fig. 12. The water-cut rubber wheel (50° Shore A) showed good friction on the rail allowing the demonstrator to climb a vertical rail, but the material showed first disintegration effects after about 20 test runs. The PU wheels (60° Shore A) showed no such effects until now, but further tests including friction tests with different rail surface conditions have to be carried out in the future.
To allow for seamless image stitching without, a pan tilt unit consisting of a Prosilica GC-2450 high resolution camera using a Pentax 418 lens together with Dynamixel RX-28 servos for actuation has been developed, which rotates around the point of no parallax-error (sometimes called “nodal point”). For lightning purposes in the pitch black BWTs, a high power LED lightning system was developed, which allows for flash operation via an Atmel ATmega microcontroller that controls light intensity, length and color distribution, is triggered by the camera and monitors flash board temperature. LEDs were chosen over xenon flashes for security reasons (potential of fermentation gas in BWTs, high voltage of ~400 V needed for xenon flashes), the shorter period between flashes and lower power consumption.
Table I: Demonstrator implementation parameters

<table>
<thead>
<tr>
<th>Control</th>
<th>Fully autonomous or remote controlled</th>
</tr>
</thead>
<tbody>
<tr>
<td>Locomotion</td>
<td>3D rail guided</td>
</tr>
<tr>
<td>Width</td>
<td>244 mm</td>
</tr>
<tr>
<td>Height</td>
<td>293 mm</td>
</tr>
<tr>
<td>Length</td>
<td>~ 900 mm (depending on amount of modules)</td>
</tr>
<tr>
<td>Weight</td>
<td>~ 9 kg</td>
</tr>
<tr>
<td>Max. Speed</td>
<td>0.35 m/s</td>
</tr>
<tr>
<td>Est. Power consumption</td>
<td>45W (depending on tank design)</td>
</tr>
<tr>
<td>Power supply</td>
<td>25.6V 5000 mAh LiPo battery</td>
</tr>
<tr>
<td>Motor</td>
<td>Brushed DC, 110 mNm, 36:1 planetary gear</td>
</tr>
<tr>
<td>Torque applied to rail</td>
<td>198 Nm</td>
</tr>
<tr>
<td>Inspection camera</td>
<td>2448x2050 px @ 15Hz, Prosilica GC2450C</td>
</tr>
<tr>
<td>Lens</td>
<td>Pentax C418DX, 100° opening angle, max aperture ratio 1:1.8</td>
</tr>
<tr>
<td>IMU</td>
<td>XSens MTi</td>
</tr>
<tr>
<td>Thickness measurement</td>
<td>ATP TE 1250-FN</td>
</tr>
</tbody>
</table>

3.2. Software

As a software framework, ROS, Quigley et al. (2009) was chosen due to its non-monolithic powerful yet easy to setup node distribution capabilities which fit well with our modular concept. Every node can register with a freely distributable master core. Node drivers where written for the XSens MTi IMU, the Faulhaber motor board MCDC 3006/S, the GPIOs of the LS-373 industrial PC board and the RS-485 driven servos in the pan tilt camera unit using C++ as programming language. For implementing single-shot capabilities of the GC2450C camera together with an exposure-dependent trigger of the LED circuit board, the AVT GigE SDK was wrapped in a ROS node. Since we wanted to keep the sensor overhead for localisation purposes to a minimum, we implemented a probabilistic estimator (particle filter) for the position on the rail with sensory input coming only from the IMU and the motor-encoder with the IMU using the position-variant magnetic anomalies of the test track and the gravitational field to determine the most realistic position and compensating for possible slippage decreasing the accuracy of the odometry. To generate the needed magnetic and gravitational “map” of the rail, we moved the IMU manually on the rail and took static measurements every 5 cm on the rail.

![Fig. 15: Generation of IMU based gravitational field map](image)

A simple autonomous behaviour was implemented as a finite state machine, which made the robot drive to certain predefined points, take 360° panorama pictures using the flash pan tilt camera unit and automatically stitch them together onboard. As a first “reactive” behaviour, the robot monitors motor current, stops and returns to the starting point if it gets higher than 2.5 A, indicating an obstacle, that it cannot overcome. The accuracy of the localisation may have to be enhanced when touching sensors have to be placed at distinct predefined points with a small manipulator, which is in development at the moment. A solution to enhance the accuracy without adding more sensors would be to use the camera to detect landmarks distributed on the rail. The gathered images where stitched together onboard to a 360° spherical panorama of each compartment of the test tank using the HUGIN/panotools scripting possibilities, Panotools.org (2011).
4. Demonstrator based Life Cycle Costs Calculation (LCCC)

4.1 Demonstrator case

The ROT project has designed and assembled a test bed to proof the robot concept. For realistic testing conditions a test tank with the following dimensions was built by the Meyer shipyard inside a 20” foot container. The structure found inside the test bed is similar to a typical double bottom tank on a cruise vessel with high complexity. The compartments are rather small to prove the concept in an environment with limited space that requires endeavours of the surveyors to access.

<table>
<thead>
<tr>
<th>Table II: Demonstrator dimensions</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Main dimensions</strong></td>
</tr>
<tr>
<td><strong>Two levels</strong></td>
</tr>
<tr>
<td><strong>Material thickness</strong></td>
</tr>
<tr>
<td><strong>Frame distance</strong></td>
</tr>
<tr>
<td><strong>Manholes</strong></td>
</tr>
<tr>
<td><strong>Upper level</strong></td>
</tr>
<tr>
<td><strong>Lower level</strong></td>
</tr>
<tr>
<td><strong>Rail length</strong></td>
</tr>
<tr>
<td><strong>Track/tank-volume</strong></td>
</tr>
</tbody>
</table>

The tracks are connected via screws to a supporting structure that is welded into the tank, comparable to the one used for cable ducts. The rails were adapted at the shipyard by heating them up with heater fans and bending them to the appropriate position. This manual method showed to be time extensive and not appropriate for an industrial application. It was proved that the robot is able to run over the rail without limitations. First tests on a vertical track were furthermore satisfactory. With a travel speed of 0.35 m/s and 20 s for a 360° degree picture the total time amounts to 10 min (shipyard personnel required 20 min) for recording the tank.

4.2 Cruise Vessel

Although the payload of cruise vessels is small, ballasting is carried out on a daily basis as the consumption of fresh water and supplies has to be continuously compensated. The ballast water tanks in cruise ships are to be found in the double bottom, the forepeak and the aft peak. In contrary to other ships types like bulkers these tanks are fairly small for the low payload but show a high structural complexity. Especially the fore and aft peak tanks are challenging to inspect with almost 150% of the surface of a double bottom tank and confined spaces difficult to access. For a cruise ship of 70,000 GT (2400 PAX) the typical ballast water tank capacity can be estimated with a total of 2500 m³ and an inside tank surface of roughly 10500 m². Based on the track/tank-volume ratio of the test bed, the overall length of the rail system would encounter for 3300 m distributed over 10 tanks. Tank surveys by class inspectors are carried out every 2.5 years in a dry dock. Nevertheless owner personnel should check the tanks independently on a more frequent basis. Depending on the conditions found inside a tank, inspections can furthermore be scheduled during the annual survey, DNV (2010).
4.3 Life cycle costs

Investment costs are composed of the robot and the installation of the rail system inside the tanks. For the first rail demonstrator, 80% of the total efforts spend - including material costs, drawings and other preparation works – where just focussed on the manual bending procedure due to the lack of automated tools for such process. The following table shows the results obtained for the prototype building scaled onto a 70.000 GT cruise vessel set in comparison to a calculation, which takes all effects of an industrial production into account:

<table>
<thead>
<tr>
<th></th>
<th>Prototype (scaled up)</th>
<th>Industrial Product</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cruise vessel</td>
<td>70.000 GT</td>
<td>70.000</td>
</tr>
<tr>
<td>Tank volume</td>
<td>2500</td>
<td>2500 m³</td>
</tr>
<tr>
<td>Required rails</td>
<td>3300</td>
<td>2950 m</td>
</tr>
<tr>
<td>assumed hourly rate</td>
<td>30</td>
<td>30 €/h</td>
</tr>
<tr>
<td>Installation*</td>
<td>1.100.000</td>
<td>90.000 €</td>
</tr>
<tr>
<td>Other</td>
<td>284.000</td>
<td>14.000 €</td>
</tr>
<tr>
<td>Robot</td>
<td>16.000</td>
<td>16.000</td>
</tr>
<tr>
<td>Total costs</td>
<td>1.400.000</td>
<td>120.000 €</td>
</tr>
</tbody>
</table>

* Material costs, preparation of substructure, installation of substructure, bending of rails, installation of rails

Assuming the availability of appropriate bending tools as they are already used for pipes, the installation costs can be reduced drastically. It is also expected that the track length and construction time can be reduced by introducing routines for rail layouts. In the calculation in table III, we therefore assumed a reduction from 2.6 to 0.1 hours per meter for construction drawings, from 8.3 to 0.1 h/m for rail preparation and from 1 to 0.5 h/m for rail installation. In comparison to the robotic inspection system costs, the shipyards are faced today with an amount of roughly 1500 man hours for inspecting the tanks, as identified in this project. Included in this number are several inspections after each production step (pre cleaning, full cleaning, 1st coating, 1st stripe coating, 2nd stripe coating and final coating) carried out by shipyard personnel and sub suppliers (mainly responsible for thickness measurements) and the final inspections by class personnel (the final inspections are around 20% of the total share). Assuming an average hourly rate of 60 € for the shipyard personnel and for class personnel the total costs for inspections will settle around 90.000 € per ship. Even with the robot system installed, there will be remaining costs for the evaluation of data the robot transmits. Under these assumptions there is few or no financial benefit for a shipyard to install such a system. The benefits must therefore be found on operational side. Already in service, the inspection time spend on one tank highly depends on the age and the condition of the ship. According to classification rules, the condition has to be rated in the 3 categories “good”, “fair” and “poor”. The better the conditions are the less detailed inspections have to be carried out, in special thickness measurements.

<table>
<thead>
<tr>
<th></th>
<th>Only minor spot rusting</th>
</tr>
</thead>
<tbody>
<tr>
<td>good</td>
<td></td>
</tr>
<tr>
<td>fair</td>
<td>Condition with local breakdown at edges of stiffeners and weld connections and/or light</td>
</tr>
<tr>
<td></td>
<td>rusting over 20% or more of areas under consideration</td>
</tr>
<tr>
<td>poor</td>
<td>Condition with general breakdown of coating over 20% or more of areas or hard scale at</td>
</tr>
<tr>
<td></td>
<td>10% or more of areas under consideration</td>
</tr>
</tbody>
</table>

The inspection time spend on one tank ranges from an average of 30 min on new built ships to 2 h on vessels older than 10 years. With the ship age the number of tanks that require more than 2 h increases, cases of 10 h are reported occasionally.
Table V: Total tank survey times for ships in operation

<table>
<thead>
<tr>
<th>Age of ship [yr]</th>
<th>Tank inspection [h]</th>
</tr>
</thead>
<tbody>
<tr>
<td>0-5</td>
<td>4</td>
</tr>
<tr>
<td>6-10</td>
<td>15</td>
</tr>
<tr>
<td>&gt;11</td>
<td>20+</td>
</tr>
</tbody>
</table>

Typical costs for an inspection team (consisting of 2 surveyors, 3 technicians and 1 owner representative) add up to 11,000 € per day. Dock fees add another 20,000 € per day to the bill as interviewed repair yard experts stated. The additional net loss (because of inexistent income but ongoing upkeep costs e.g. crew) for the owner per dry dock day can be estimated roughly with 150k€, N.U. (2010).

The scenario, developed within the project, will allow to minimize the time spend inside tanks. During operation robots are placed inside the tanks even before the surveyors enter the ship and decisions whether or not entering the tanks is necessary can be made beforehand. Failures are marked on the pictures and repair teams are directed to the corresponding spots. Under the assumption that inspection costs could be halved this way it accounts for an estimated saving of 140,000 € (one day dock fees saved after 10+ years) for the first 20 years. This alone would compensate the system investment costs. If we also take into account an earlier return to service the effect is of large significance for the owner, because by saving only one dry dock day and the corresponding net loss, the investment for the robotic system already pays off.

5. Conclusion and outlook

In this paper, we evaluated different locomotion techniques and other robotic principles potentially suited for robotic tank inspection and maintenance, with a focus on especially confined tanks like double bottom ballast water tanks in the construction phase and in use. We described a suited technical concept for such tasks and showed the implementation of that concept in an autonomous rail guided robotic demonstrator. The modular concept uses a simple corrosion-free thermoplastic rail which runs through the flexible robot and thus allowing for high payloads and upside-down operation. By demonstrating the ability to drive through a specially designed test tank that directly resembles a double bottom BWT and generating high resolution panorama views of the inside, the robot proved that the chosen concept works in such environments. A future application that adds value might be the spot repair (blasting and painting) to maintain tanks in a good condition for a longer period of time. This will affect maintenance and survey costs and will reduce inspection times further. An additional requirement for the establishment of robotic systems to support controls and surveys inside tanks will be the approval of classification societies as a substitute for human in-tank inspections. In addition to further life cycle cost analysis, future work on the robotic system has to and will include the integration, control and testing of a manipulator for the demonstrator platform as well as intensive testing of different wheels, rail-materials and -surfaces under different conditions (moisture, dirt). The concept of a cog railway will be considered for future enhancements of the proposed concept.

Acknowledgments

This work has been funded by the German Ministry of Economics, grant number 03SX269A.

References


DNV (2010), Rules for Classification of ships, ships in operation; survey requirements, Part 7/1


N.U. (03.03.2010), *Norwegian reports*, Cruise Industry News 4/26


Patent Protection of Software-Based Inventions in the Maritime Industry

Rikard Mikalsen, Newcastle University, Newcastle upon Tyne/UK, rikard@mikalsen.eu

Abstract

This paper presents some legal and strategic aspects relating to international patent protection of software-based inventions in the maritime industry. First, a brief overview of patent systems in relevant countries is presented, and particular regulations relating to software-based inventions are discussed. Then, important aspects of enforceability are reviewed, including the situation in important developing economies and the enforcement of patent rights against ships in operation. Finally, protection strategies for software-based technologies are discussed in light of the above.

1. Introduction

It is widely accepted that innovation is key for the European shipbuilding industry to stay competitive, a fact which is also reflected by the recent initiatives by the European Commission to raise the awareness of intellectual property-related issues in the industry. European shipyards and suppliers are particularly involved in the development of ships with high technical complexity, such as passenger and specialist vessels. With the current high innovation rates, there is potential to obtain strong legal protection for new developments for these vessel types. A good understanding of the international framework for protection of intellectual property is, however, critical in order to fully utilise the legal instruments available and to avoid technology leakage.

Innovation in the maritime industry, as in many other industries, is increasingly related to software-based systems. For shipbuilders and suppliers, examples include:

- shipyard design methods, production methods and logistics;
- control systems for machinery and auxiliaries (such as exhaust gas cleaning and ballast water treatment systems);
- vessel condition monitoring, energy management and operational optimisation;
- navigation systems;
- cargo handling and management;
- control and integration of alternative energy sources, such as wind (e.g. Flettner rotors).

The fact that software and computer systems play an integral role in many new products means that additional awareness is required on the part of the IP owner. This is due to the fact that software is easily leaked (as source code; it can, however, be very difficult to reverse-engineer object code), a software-based system can in many cases be easily reproduced, and because patenting of business methods and software are excluded by many national patent offices.

1.1 Protection of intellectual property

All industrialised countries have laws giving protection for “creations of the mind”, i.e. intellectual property (IP). There are many different types of IP, and consequently a variety of legal instruments to protect it. The most common types of industrial intellectual property include copyright, trade marks, rights in designs, and patents. Closely related to this, and often also considered under the heading “intellectual property law”, are aspects of contract law, licensing, unfair competition, and trade secrets.

As in other industries, IP tools can be utilised to protect innovation in the marine industry. However, due to the highly international nature of the industry, there are some particular challenges associated with this. These relate to the fact that where the ship is built, where it is flagged, where the owner is based, and where it is operated from may all be different countries. For example, a typical ship in international traffic could be built in South Korea, flagged in Panama, owned by a Norwegian
company, and spend most of its time in international waters. Hence, the international aspect of IP protection becomes critical. Importantly, some of the largest shipbuilding nations have immature and unreliable IP protection systems, whereas many of the major flag states do not have functioning IP protection mechanisms at all, *Houthoff Buruma* (2007), *Mikalsen and Harlfinger* (2011).

1.2 Patents

Patents are commonly considered to be the strongest type of IP protection, since a patent will protect an idea or a concept, not just one specific implementation. A patentee can prevent others from using the invention commercially, even if the other person has come up with the same idea independently and without being aware of the patent. Unlike some of the other types of IP rights, patents are examined before grant in order to ensure that a monopoly not be granted for something that is already known in the technical field in question.

Although a number of international treaties exist to harmonise patent and IP laws, patentability and infringement are matters for national courts, and local variations therefore do exist. One of the most important differences between national laws is how software and business method inventions are treated. This will be discussed in more detail below.

The costs associated with getting and maintaining patents are not insignificant, in particular if protection is sought in a large number of countries. Since patent protection is national, a patent must be applied for in every country where protection is desired. (There is no such thing as a “world patent”, however some regional systems, such as the European Patent Office, allow the grant of a bundle of national patents in a single procedure.) Although the same application can, initially, be used in all countries, translations will usually be required, as well as the use of local attorneys to handle the application process.

As examples, the Australian patent office\(^1\) estimates the cost of obtaining an Australian patent, including attorney fees, to AUD 6,000 to AUD 10,000, depending on the complexity of the application, with maintenance fees over a 20 year term of a further AUD 8,600. The European Patent Office\(^2\) has estimated that a typical cost for obtaining and maintaining a European patent in 6 countries is €30,000. In a report, the US General Accounting Office (2003) estimated that the cost of a “relatively short and straightforward” US patent, maintained for 20 years, is about US$ 10,000, while additional protection in selected European countries (France, Germany, Italy, Ireland, Sweden, and the UK), as well as Canada, Japan, and South Korea, would costs between US$ 160,000 and US$ 330,000.

2. Protecting software-based inventions

Creating “layers” of IP protection around a product is usually advisable to obtain best possible protection, and this can include trademark, copyright, utility model, and design protection. However, for technology developers the core IP value usually lies the patent portfolio and trade secrets associated with the product design.

2.1 Alternatives and complementaries to patenting

For technical inventions, there are two main complementaries to patenting: copyright or design protection and trade secret protection. For software inventions, copyright prevents direct copying of the code and exists automatically and independent of whether the invention is additionally patented. However, copyright (and, similarly, rights in designs) only protects a specific expression of an idea, not the idea itself. It does not prevent someone from writing a similar piece of software which does exactly the same thing. In practice, copyright can prevent direct copying of software, design drawings, manuals, etc., but only has a minor role in the protection of new technical solutions.

\(^1\) [http://www.ipaustralia.gov.au](http://www.ipaustralia.gov.au)
\(^2\) [http://www.epo.org](http://www.epo.org)
More important, however, is trade secret protection. This is often seen as an alternative to patenting ("should we patent the invention or keep it as a trade secret?"). but it is important to note that information "around" a patented product can also be protected as a trade secret. Trade secret protection exists automatically for all business-sensitive information provided it is not in the public domain and that steps are taken to maintain secrecy. (The latter usually means having appropriate internal routines for handling the information.)

Trade secret protection can be maintained indefinitely as long as the information is kept secret, hence it can provide much longer protection than a patent can. However, once it has been leaked (intentionally or not), the protection is lost. A further advantage is that trade secrets protect anything, also incremental developments that would not be patentable, general know-how, or inventions excluded from patentability, such as business methods. Trade secret protection typically works well for complex products which are difficult to reverse-engineer (which may include many software-based inventions, which are typically supplied to the customer as machine code), as well as internal processes which cannot be discovered by inspecting a company's products.

Potential disadvantages of relying solely on trade secret protection include: (a) other people can come up with the same idea independently, or after studying your product; (b) leakage channels, such as:
- employees (people change jobs, and although non-compete clauses exist, these are limited in time and scope);
- end customers (after purchasing a ship, the shipowner decides who can inspect it);
- business partners (after installing one system from a supplier, a yard may decide to copy it and offer it directly themselves);
- suppliers who are involved in design processes but also work with competitors;
- classification societies, who require detailed knowledge of the technology; and
- universities and research centres hired for collaborative research or consultancy.

Despite the limitation and risks for leakage, trade secrets form a essential part of IP protection in all companies since it protects most of the business know-how and procedures. Importantly, trade secrets complement patents in that not all aspects of an invention must be published in the patent filing. Surrounding a patented product will be much information about e.g. optimum design solutions, which will not be public and therefore still subject to trade secret protection, even though the core concept of the product may be patented.

2.2 Software and business method patents

Patentability of software or business method inventions is one area where there are differences in the law internationally. In some jurisdictions, including the United States and Japan, business method and software patents are generally accepted; the requirement is merely that the invention, in addition to being new and inventive, have a practical application. In other countries, notably EPO and China, a technical effect is required, and software and business methods as such are not patentable.

<table>
<thead>
<tr>
<th></th>
<th>Software</th>
<th>Business methods</th>
</tr>
</thead>
<tbody>
<tr>
<td>Europe (EPO)</td>
<td>Pure software excluded; patentable if a technical effect is present</td>
<td>Excluded (except where a technical effect is present)</td>
</tr>
<tr>
<td>United States</td>
<td>Patentable</td>
<td>Patentable</td>
</tr>
<tr>
<td>China</td>
<td>Pure software excluded; computer-based system patentable if technical effect is present</td>
<td>Excluded (except where a technical effect is present).</td>
</tr>
<tr>
<td>Japan</td>
<td>Patentable</td>
<td>Patentable</td>
</tr>
<tr>
<td>Korea</td>
<td>Patentable</td>
<td>Patentable</td>
</tr>
</tbody>
</table>

Table I shows an overview of the law relating to software and business method patents in some relevant jurisdictions. It should be noted that this is a general overview and that nuances exist. In
many cases, business method inventions are implemented on a computer, hence there is an overlap between the two areas. A computer-implemented method of doing business may be patentable in Europe and China if it has a technical effect, and there are examples of such patents that have been granted. Conversely, a computer program that only solves a business problem, and not a technical one, (for example, providing financial analysis or merely automating standard business tasks) would normally not be patentable.

In Europe, in order to be patentable, a computer program must cause a “secondary” technical effect, beyond just the normal running of the software on a computer. Hence, if the invention as a whole is technical, it is not automatically excluded just because it is implemented using a computer program. This is of significant importance for engineers, since most new developments have some technical application. The statutory exclusion of software from patentability is therefore of less concern to technology developers in the maritime industry than it would be at first sight.

Some examples of evaluations relating to patentability of software-based and business method inventions will be given in Section 4.

3. Availability and enforcement of patent rights

A patent gives legal rights to prevent anyone from utilising an invention commercially without the permission of the patent owner. This includes importing, manufacturing, or selling, as well as commercial use of the invention. A patent can therefore be enforced against a number of entities, including manufacturers, importers, distributors, and end users.

Usually, a rights holder will first pursue importers, manufacturers, or sellers of the infringing article, since this will catch the unauthorised activities at its source. Although enforcement of patents against end users is rare in many industries (such as consumer goods), it may be a worthwhile—or indeed the only—option in others. For example, for patented processes, there may only be infringement in the form of use. For product patents, if the number of users are small and their financial power is adequate, then pursuing end users may be the best strategy. This may be the case in the marine industry, where enforcing a patent against a shipowner may have a greater chance of success than, for example, a small importer or distributor.

3.1 Temporary presence exception for foreign ships

The “temporary presence exception” is set out in the Paris Convention for the Protection of Intellectual Property Rights, an international treaty signed by practically all industrialised nations. It provides that a patent can not be enforced against a foreign vessel engaged in international transport, if the infringing article is used on-board only for the normal operation of the ship. The actual convention text is:

In any country of the Union the following shall not be considered as infringements of the rights of a patentee:

(i) the use on board vessels of other countries of the Union of devices forming the subject of his patent in the body of the vessel, in the machinery, tackle, gear and other accessories, when such vessels temporarily or accidentally enter the waters of the said country, provided that such devices are used there exclusively for the needs of the vessel;

[...] 

This provides security for shipowners, who don't have to deal with (potentially differing) patent rights in each port the ship enters, and ensures that international trade is not hindered by inventors taking legal action against visiting ships. However, in the current maritime marketplace, this provision presents a challenge for technology developers seeking broad protection of their inventions, in particular for software-based systems.

Some case law on the application of this exception exist, and the courts have generally interpreted the
exception broadly and not allowed enforcement of patent against foreign ships. (See Mikalsen and Harlfinger (2011) for a detailed analysis.) A key problem is that a large fraction of ships in international traffic will never enter the territorial waters of their flag state. Once in operation, such a ship will never infringe any patent: on the high seas there are no patent rights and when in port it will fall under the temporary presence exemption for foreign ships. Even if a ship does regularly or occasionally enter flag state territorial waters, there will be a question of enforceability: many of the major flag states do not have functioning legal system for granting and enforcing patent rights. Moreover, since the “competitiveness” of these open register shipping flags to a large degree depends on a lax regulatory framework, these countries probably have little incentive to change their systems.

For practical purposes, it would be very difficult for a technology developer to obtain protection against unauthorised use of an invention on ships in international traffic. It should, however, be noted that this only applies to ships in international transport. Vessels operating nationally, e.g. in operations associated with oil and gas exploration, will not fall under the exception.

3.2 Legal situation in Asia

The direct consequence of the temporary presence exception is that, for a large fraction of the world fleet, once a ship is in operation, it is free from patent infringement. Hence, the only practical method of protecting inventions used in international shipping is to seek protection in countries where the invention is manufactured or sold. The legal situation in the major shipbuilding countries therefore becomes critical, since any patent rights would have to be enforced before the ship enters operation.

Relevant countries in this respect include Japan and European countries, which have mature and well-established IP protection systems, and—importantly—China and South Korea. While both China and South Korea have functioning IP protection systems, including patent offices which are party to the main international treaties, the legal situation for enforcing IP rights in these countries somewhat more uncertain. Enforcing patent rights in these countries have been associated with problems such as lack of experience in the judicial system, different practice between courts, low damages awarded, local favouritism, and corruption (Muller-Stoy, 2007; Fabry, 2005). The recent years have, however, seen an improvement, with some high-profile cases where Western companies have successfully defended their rights. Foreign rights owners therefore have reasonable chances of enforcing their rights in China and South Korea, however there is clearly a higher level of uncertainty than in, for example, Europe or Japan.

3.3 Patent strategy

If patenting of an invention has been decided, which patent strategy will then give best possible protection for maritime inventions? Due to the high cost of patenting if protection in many countries is sought, a strategic selection must usually be made. Here, three strategies for patent protection of maritime inventions will be briefly summarised and evaluated in light of the temporary presence exception and the current legal situation:

- Protection in major shipbuilding countries.
- Protection in territories of operation and/or major ports.
- Protection in major flag states.

It is usually preferred to enforce patents against manufacturers and sellers of the patented invention, hence protection in shipbuilding countries would probably be the first choice in most cases. The largest shipbuilding countries in terms of tonnage are South Korea and China, and patent protection in these countries is possible but may be associated with enforcement difficulties, as noted above. For specialist and more technically advanced vessels and systems, Europe and Japan should probably be considered. A high level of protection is in theory available to prevent unauthorised manufacturing and sale through patenting in the major shipbuilding countries, however the uncertainty regarding enforceability in China and South Korea should be kept in mind. Due to this, many companies, in
particular SMEs, may not consider it worthwhile to pursue patent protection in Asia. However, if, for example, patent protection is only acquired in Europe, then a patent holder may have rights against local competitors but Asian companies would be free to build and sell the invention. (And how to do it would be described in the patent.) In such cases, keeping the invention as a trade secret should be carefully considered in order to minimise technology leakage.

Patent protection in countries with major ports would be the most intuitive strategy to cover the use of an invention on ships in international traffic, since most ships will regularly call in the major ports. However, the temporary presence exemption completely eliminates this as a viable option. Only ships sailing under the flag of the country in question would infringe a patent; foreign ships would be exempted. For example, in the port of Rotterdam, a Dutch patent could only be enforced against Dutch vessels.

Protection in territories of operation can, however, be a good strategy for inventions used on ships involved in national transport or any form of operations at sea. This includes, for example, vessels for oil and gas exploration or installation of offshore wind turbines. Such vessels are not solely involved in international transport, and can therefore not claim defence under the Paris Convention (irrespective of the nationality of the ship). Hence, specialist systems for such ships can be protected in the major areas in which they would operate. For offshore oil and gas vessels one could, for example, patent in USA, Brazil, Australia, Norway, and the UK to cover major exploitation areas.

Finally, protection in major flag states would be an option to consider. A ship can obviously not claim defence under the temporary presence exception when visiting a home port, hence a patent in the flag state could be enforced when the ship is in flag state territory. Protection in major flag states could therefore give a technology developer broad protection for his inventions. There are, however, two problems with this strategy. Firstly, the major flag states, in which the majority of the world tonnage is registered, have limited legal remedies available for awarding and enforcing intellectual property rights. Hence, even if a patent could be obtained and the ship regularly visits a home port, the patentee may not be able to enforce his rights. Secondly, many of the largest flag states do not lie on any major shipping route, and many ships in international traffic rarely or never enters flag state territory. Hence, a flag state patent would never be infringed.

Despite this, flag state patent protection could be an effective (and probably the only realistic) strategy to prevent unauthorised use on-board ships of countries with major ports. If one targets countries that (a) are large flag states, (b) have busy ports, and (c) have effective legal systems, then some protection could be obtained. For example, patenting in China, Singapore, and Hong Kong would cover a small but non-negligible fraction of the world fleet, and such patents could be enforced against ships visiting a home port.

4. Examples of invention analyses

For software-based maritime inventions, the situation becomes even more complicated than what the above indicates, since it can be particularly difficult to spot and prove infringement, and inventions can be more easily copied. In this final section, some examples of on-board and shore-based maritime inventions will be presented and briefly analysed in order to give an overview of the potential for effective patent protection for such inventions. Please note that this is not a formal legal analysis and that details have been omitted for clarity.

4.1 On-board systems

Numerous on-board systems rely heavily on software, and much development is ongoing in these areas, for example in relation to condition monitoring and operational optimisation. Such systems are often technical in nature, and therefore possibly patentable subject matter. However, enforcement may be a problem for systems used on ships in international traffic, in particular for inventions which are easily retrofitted on board operating ships.
Fig. 1 shows the Liner Guardian system developed by Portuguese company TecnoVeritas. The system monitors the onset of scuffing between the piston and cylinder liner, which is caused by a breakdown of the lubrication film in the cylinder and can cause significant engine damage. It utilises four sensors to measure surface roughness on the piston side surface and a computer algorithm to analyse the measurements and compare it to reference data. Although the mathematical algorithm and computer system used to analyse the readings are clearly key elements of the invention, the system as a whole is technical, and is therefore most probably patentable in most countries.

A similar situation will arise for on-board operational optimisation systems, which are currently being offered by an increasing number of companies. For example, a software-based system to obtain optimum vessel trim or optimise the operation of a machinery system, see e.g. Hansen and Freund (2010) could be patentable based on that it works on real operational data and produces a technical effect. (A change in the vessel trim.) It should, however, be noted that, in all cases, the invention must be new and inventive; a mere automation of, or incremental improvements to, known methods or algorithms will not suffice.

With regards to business methods, the European Patent Office is usually quite restrictive. However, for inventions which have a technical character, or where a problem is solved through new technical means, protection may be available.

Fig. 2 shows a system for distributing dry bulk, such as fertiliser and grain, in which, instead of being transported in bags, bulk transport is used with the material being bagged upon unloading from the ship. The bagging is carried out on the port quay by bagging units being modular and having the dimensions of a standard container, thereby allowing them to be easily transported and set up where required. A claim to a “method of material distribution” as described above was allowed by the European Patent Office, with the reasoning that it involved a new type of bagging plant, has a technical character, and uses technical equipment to achieve a technical result.

Although many systems used on-board will be technical, and therefore patentable, patents on software-based inventions for ships may be very difficult to enforce. Ships in international traffic can

---

3 A patent application has been filed for this system; see WO/2009/037661.
4 By the Technical Board of Appeal, upon the validity of the patent being challenged; see decision T 0636/88.
in practice not be pursued, due to the temporary presence exception. Moreover, software systems may not have to be supplied with the ship, but can be retrofitted on-board once the ship is in operation. In such a case, the system can be supplied from a country where protection has not been obtained, and the inventor has no viable method of preventing unauthorised manufacturing and sale. An exception to this will be for inventions used on ships involved in national traffic or marine operations, for which patent protection in major areas of operation may be a viable strategy.

4.2 Shore-based systems

Considering shore-based systems, one could differentiate between inventions related to yards or suppliers, and such used by shipowners in ship or fleet operation and management. From a shipowner's point of view, consider a new and innovative route planning and optimisation system for a single ship or a fleet of ships. As a (made-up) example, this could perhaps be a system evaluating the financial benefit of slow steaming by producing a 30-day forecast for bunker prices and combining this with route information and freight rates. Although this invention could be implemented in a technical system (a high-performance computer), it is fundamentally a business problem, as both the inputs and outputs are business data. Hence, patenting this invention in Europe would probably be challenging.

A somewhat similar system was proposed by Cerchione et al. (2007), Fig. 3. The invention allows shore-based databases with relevant information to be used for decision support on-board. Potential zones of concern in the planned route, for example due to terrorism, piracy, weather conditions, can be utilised ensure safe and efficient ship operation. Route planning for ships (as such) may be considered a business method and therefore receive objections by a European examiner. However, if implemented in a technical system and not purely used to make business decisions, the method may be patentable. For example, a claim to an on-board system in which the output is used directly in the navigation of the ship could be acceptable since a technical effect, in the form of a change of course of the ship, is produced.

In relation to infringement and enforcement of a patent for a shore-based system, things are somewhat clearer than for patents used on board ships. If an invention is operated from an office of a shipowner (or some company), then the patent laws in that country would apply as usual. This probably includes a situation where a software program is run on an external server and operated locally via a computer link. Hence, if an invention is operated locally, a potential infringer could not base a defence on that the computer actually running the software was located in a different country.
For yard processes, there is often a strong case for keeping inventions as trade secrets, particularly those which cannot be directly reverse-engineered from a product. Moreover, many yard inventions are for processes, making it more difficult to spot infringement of a potential patent. In this respect, it should, however, be noted that a direct product of a patented process is also an infringing article. For example, a patent for a new process for welding can also cover the welded product. This makes it easier to enforce process patent rights, particularly if the process used is identifiable from the product, and also means that an infringer could not simply move production abroad and re-import the product.

Yard or port logistics processes would probably in most cases be considered business methods and would therefore not be patentable in Europe. The situation would be similar for production planning tools; for example, a system for organising airline crew was considered unpatentable in a practice note by the UK Patent Office. However, a new method implemented in a technical system may not be excluded; for example, an automated container handling system for a port may be patentable, since it has a technical effect in handling the containers. Hence, a business method in itself is not patentable, however a technical system utilising that method may be. (This is somewhat equivalent to the fact that laws of nature and scientific discoveries are unpatentable, but machines that work based on such laws may be, if they are otherwise new and inventive.)

For new design algorithms, which would generally not be patentable in Europe, note that the product may be patentable even if the design algorithm is not. If a product is novel and inventive, it is irrelevant how it was created; the patent law does not differentiate between something created in a flash of genius by an engineer and that created through months of high performance computing time. Hence, e.g., if a new ship hull form was created by an evolutionary algorithm, it may be worthwhile looking at whether this can be patented. The design algorithm would not have to be disclosed.

5. Conclusions

Patent protection of maritime inventions remains a highly complex issue, and it is extremely difficult to give general guidelines. For developers of maritime technology it is, therefore, critical to have a basic understanding of the international legal framework in order to obtain acceptable protection for new developments and to avoid technology leakage. The key issues discussed in this paper were:

- The (potentially considerable) costs associated with patenting and the requirement that the invention be fully disclosed in the application.
- The advantage of trade secret protection if the invention can be kept secret, but the vulnerability of such protection in the form of a high number of potential leakage channels.
- The limitations in the patentability of software and business methods in Europe, which,
despite not being a big issue for most technical inventions, creates a more complex international legal framework which a technology developer must deal with.

- The “temporary presence exception” for foreign ships, which in practice frees ships in international traffic from patent infringement worldwide. (A situation which is unlikely to improve in the near future.)
- The potential difficulties and uncertainty associated with enforcing patent rights in major shipbuilding countries such as China and South Korea. (And, although the situation appears to be improving for these countries, the emergence of new major Asian shipbuilding nations with limited IP protection mechanisms may maintain this problem.)

In the current situation, it is understandable that the European shipbuilding industry does not make full use of the patent protection possibilities, but rather “keeps as much as possible secret and innovate faster than the competitors can copy”. Although these problems have long been recognised by both the industry and the European Commission, changing the international legal framework is a notoriously slow process, and it seems that the industry will have to continue to rely to a large degree on trade secret protection for the foreseeable future. This does not encourage the publication of new developments (which is one key purpose of the patent system) and the current legal framework clearly has improvement potential in order to better promote innovation in an industry in which continuing technology development is critical to meet future fuel efficiency, emissions, and safety requirements.

Disclaimer

This article represents the personal opinions of the author and is not legal advice. While every effort has been made to ensure the accuracy of the information presented, some aspects have been omitted for clarity of presentation. For comprehensive advice on intellectual property related matters in your jurisdiction, please consult a local IP professional.

References


HANSEN H.; FREUND M. (2010), Assistance tools for operational fuel efficiency. 9th Int. Conf. on Computer and IT Applications in the Maritime Industries (COMPIT 2010), pp. 356-366.


Integration of DB Oriented CAD Systems with Product Lifecycle Management

Roberto Penas, SENER Ingeniería y Sistemas S.A., Tres Cantos/Spain, roberto.penas@sener.es
Carlos González, SENER Ingeniería y Sistemas S.A., Tres Cantos/Spain, carlos.gonzalez@sener.es

Abstract

This paper describes the solution designed to integrate a DB oriented shipbuilding CAD system with product life cycle management systems. It is organized into three different phases: (1) Define Integration architecture through a neutral framework and first prototypes. (2) Solution release into a shipbuilding CAD: FORAN. Solution based on document management and on demand data publishing and synchronization between CAD and PLM, including metadata and 3D graphics. (3) Extended functionality by enabling automatic publishing and synchronization mechanisms, access control managed by PLM system and extended document management and 3D graphics support. The software solution described here is called FPLM.

1. Introduction

1.1. Nomenclature

A nomenclature is required for papers/technical notes using a large number of symbols, abbreviations and acronyms.

**FORAN**: Shipbuilding CAD application owned by SENER, Ingeniería y Sistemas.

**FPLM**: Software solution for connecting FORAN to a PLM system and based on the architecture described in this document.

**PLM**: Acronym for Product Lifecycle Management system.

**Java**: Software development platform owned by Sun Microsystems Inc.

**CORBA**: Common Object Request Broker Architecture

**IDL**: CORBA Interface Description Language.

1.2. Overview

FORAN system is used here as a sample shipbuilding DB oriented CAD to be connected with a generic PLM. With most file oriented CAD systems, integrations with PLM systems usually consist of implementing a series of tools to put CAD files under PLM control, by setting up some rules, procedures and user interfaces.

However, the challenge here is that in DB oriented CAD systems, this integration is performed in a different approach, based on two main blocks:

- A set of tools for publishing and synchronizing product data between CAD and PLM databases.
- A set of document management tools to enable putting CAD generated documents under PLM control within CAD environment.

1.3. Development phases

These are the phases to develop the integration between the shipbuilding CAD (FORAN) and different PLM systems:

1. **Initial**: Integration architecture through a neutral framework and first prototypes. This phase is now completed.
2. Second. First integration release into a shipbuilding CAD: FORAN. Solution based on document management and on demand data publishing and synchronization between CAD and PLM, including metadata and 3D graphics. This phase is completed.

3. Third. Extended functionality by enabling automatic publishing and synchronization mechanisms, access control managed by PLM system and extended document management and 3D graphics support. This phase is currently a work in progress.

2. PHASE I – Architecture definition and solution prototype design

2.1. Architecture overview

The connection solution consists of a number of components:

- A main Java process called FPLM Server which is the basic connection point between the PLM processes and FORAN functionality.
- A set of CORBA IDL’s that define the methods interface between FORAN and FPLM Server.
- A set of Java classes and methods that run embedded in FPLM Server that is the specific implementation for PLM system. This component is called FPLM Plug-In and consists of two parts:
  - A set of classes specific for each PLM system that runs embedded in FPLM Server, and are in charge of prepare the data and invoke the PLM functionality through Web Services
  - A package called FPLM FORAN Adapter embedded into PLM server that consists of:
    - A set of PLM Web Services that are invoked from FPLM Server included in PLM itself. Some Web Services are out-of-the-box (included in the standard PLM), while others are specific for FORAN integration.
    - A set of templates to define a marine entity definition package, including entity types, attributes for that entity types, lifecycles and workflows.
- A set of FORAN user interface client utilities, invoked from FORAN modules, to perform operations with PLM from FORAN, and that implement the CORBA IDL’s explained above. These client utilities are generically called FPLM Client.

2.2. Architecture diagram

![Fig. 1: FPLM architecture](image)
2.3. Architecture components

2.3.1. FPLM Client

*FPLM Client* consists of a set of utilities that will be integrated into different FORAN modules to interact with PLM server. *FPLM Client* is developed with the standard FORAN user interface tools, so it has exactly the same look and feel as other FORAN user interfaces. *FPLM Client* utilities are included in these big groups:

1. **PLM connection setup.** Basic tools to select a PLM server alias, perform user authentication and set local user parameters.
2. **Document Manager tools.** A set of document manager tools to be invoked from FORAN modules that generate documents. Tools to manage links between model and documents are included.
3. **Data publishing and synchronize tools.** A set of tools to publish and synchronize FORAN model tree data into PLM product structures.

2.3.2. FPLM Server

*FPLM Server* is a Java process in charge of managing FORAN business objects and their mapping to PLM objects. *FPLM Server* will include a set of functional libraries called *FPLM Plug-In*. *FPLM Server* relies on a couple of configuration files:

- *plmconfig.xml* registers the available PLM server aliases and their basic configuration.
- *entitytypes.xml* registers the entity definition types and their attributes that must exist in PLM entity type definition.

If either *plmconfig.xml* or *entitytypes.xml* do not contain the appropriate entries, *FPLM Server process* cannot start properly. FPLM Server also relies on a standard CORBA ORB server that will be the communication port between FORAN FPLM Client and Server.

2.3.3. FPLM Plug-in

A set of specific PLM class libraries that will run embedded into FPLM Server. *FPLM Plug-In* will manage the business logic and communicate it to PLM Server. This component includes two parts:

- A set of Java classes and methods that run embedded into FPLM Server. These methods are in charge of invoking the specific Web Service calls to the PLM server.
- The PLM Adapter is a set of Web Services and PLM templates that are embedded or run in the PLM server.

2.3.4. FPLM CORBA IDL Specification

A standard CORBA server is the communication interface between FPLM Client utilities and FPLM Server process. There is a CORBA IDL set of classes and methods that will be the specification interface to communicate FORAN with PLM. FPLM Client will make use of those IDL implementations in the client side. FPLM Server will make use of those IDL implementations in Java in the server side. This approach helps the client side applications to be fully independent on the kind of PLM system we are using.

3. PHASE II – FPLM Release

3.1. Overview

The first FPLM implemented solution has been released with latest FORAN version (v70R1.0) and
consists of a set of PLM utilities fully embedded into FORAN modules that are classified in the following groups:

- FPLM connection setup.
- Local document manager utility.
- Local document manager extension for drawing generation module (FDESIGN).
- Local document manager for diagram modules.
- Model tree manager utility.

The main goal of this first FPLM release was to provide a comprehensive set of document management tools integrated into FORAN modules, as well as to provide a on demand tree publishing and synchronization between FORAN and PLM systems. This has been fully achieved.

3.2. FPLM connection setup

3.2.1. Functional Overview

A new FPLM menu is enabled. To perform authentication, you can either:

- Go to FPLM Menu and select **Set connection parameters** option:

  ![Set connection parameters](image)

- Click the **Set connection parameters** icon:

  ![Set connection parameters](image)

The authentication window is shown in next figure:

![Connection parameters](image)

**Fig. 2: Set connection and authentication parameters**

Where:

- **PLM Alias** is the PLM server identifier according to the FPLM Server configuration. You must select the appropriate PLM from the drop down list to authenticate to.
- **User ID** is the PLM user Id. By default, the current FORAN user Id is used. You can reuse the current FORAN user Id if it also exists in PLM user definition schema.
- **Password** is the PLM password for the user. By default, the current FORAN user password is used.
- **Local directory** is the user’s local directory from which files will be transferred to and from PLM server. By default, the current FORAN user local directory is used.

Click OK button to authenticate in PLM system. If authentication is successful, you have then just opened your connection session.
3.3. **FPLM local document manager**

3.3.1. **Functional Overview**

This is a local file manager to interchange files and documents with PLM server, ready to be invoked from all FORAN modules and within the context of a concrete FORAN project. This utility will be aware of the FORAN module from which it is invoked and will manage the file types proper of that module only, but will have no direct interaction with the data opened in the module session.

The basic function of this utility is to provide a basic mechanism to publish into PLM all files generated in FORAN modules (data files, reports …). This module lists local files, showing their state with regard to PLM. If it is needed to add a file to the local list from PLM, then a search capability will be used to allow document selection, download and, also, check-out options.

The basic metadata for a file document is stored in a file with the same name and `.fplm` extension, placed in a directory called `.fplm` under working directory.

A specific IDL interface method is used to pass the appropriate data to FPLM Server.

This utility can be invoked from a FORAN module, and is prepared to manage the file types that can be generated or managed from this module.

The common operations that can be performed with local document manager are the following:

- Manage document content types contextualized for the FORAN module used.
- Lists all local files managed by FORAN user and shows their PLM status.
- Search documents of a specific content type in PLM and add them to local files list.
- A set of dynamically enabled/disabled operations with selected documents.
  - Download file content from PLM server.
  - Check-out a document for local modification.
  - Undo a previous check-out.
  - Check-in a document after a local modification.
  - Upload a new document to PLM server.
  - Create a new document version.
  - Remove local file. This operation removes the file from local disc and from users file list. It does not remove the document in PLM. If the user needs working with the file again, he/she can search it again in PLM and return it to local list.
- Upload a document from a FORAN module, and manage from a different one. This is possible for certain document content types.

There is an editable XML template to configure the content types and FORAN modules compatibility map.

Local document manager can be invoked by either:

- Go to FPLM Menu and select **Local document manager** option:
  - ![FPLM menu](fig.png)
  - Click the **Local document manager** icon:

Fig. 3 shows current FPLM local document manager tool.
Fig. 3: FPLM Local document manager

For searching a document in PLM system and add it to local list for user edition, we should click the Search Related Documents in PLM button, once a content type has been selected. The attributes are dynamically supported; this means that those attributes added to FORAN document types are automatically supported in FPLM interface. Once we have found the requested documents, we just need to select them and add them to the local list by clicking the File Download and Add to Local List button. This is shown in Fig. 4.

Fig. 4: Search documents in PLM system.

3.3.2. Extension for FDESIGN

When using FPLM Local Document Manager in FDESIGN module, we can also perform specific tasks that allow further interaction and eliminates potential errors in data manipulation:
• Open a document file into current FDESIGN module session, by selecting a file in the list and clicking the *Open in session* button.
• Upload current opened file into PLM as new document.
• Check-in the current opened document file.
• Manage drawing BOM. Parts in PLM will be linked to the drawing that describes them.

![Local Document Manager](image1)

**Fig. 5:** FPLM Local document manager in FDESIGN; Observe the *Open in session* button.

![FPLM icons](image2)

**Fig. 6:** FPLM icons in FDESIGN. Observe *Upload* and *Check-in* buttons.

In addition, a *Set product data* option is included to help PLM system to determine the appropriate part versions to be used for the applicability context. This concept is more widely explained in *Model tree manager* section.

### 3.3.3. Local Document Manager for Diagram Modules

In FORAN, schematic diagram graphics and metadata are stored in FORAN DB, therefore it has been implemented a special case for managing this type of documents in a PLM system.

In FORAN schematic design FORAN modules, the uniqueness diagram definition is through FORAN keys, like project Id, system Id, and sheet number, allowing the use of simplified diagrams published as PLM documents. Document Upload, Check-out and Check-in are automatic operations and user just need to invoke a generic *Publishing schema* operation that automatically detects whether the diagram document exists in PLM or not, and that performs those operations automatically when needed.

Additionally, a PDF snapshot is generated at publishing time, using that file as primary content for document.
Fig. 7: A FORAN P&ID and the existing FPLM options, including publishing schema

Fig. 8: A Publishing schema window
As well, a Set product data option is included to help PLM system to determine the appropriate part versions to be used for the applicability context. This concept is more widely explained in Model tree manager section.

3.4. Model tree manager

3.4.1. Functional Overview

This is a set of integrated tools into build strategy definition module (FBUILDS) to allow on demand tree publishing and synchronization capabilities between FORAN and PLM system, as well as enabling 3D mock-up visualization of FORAN ship model into a PLM system.

Two additional menu options are included in this module: Set product data and Publish data cleanup.

3.4.2. Set product data

This option helps to establish the appropriate environment for data to be created and/or updated as a result of publishing and synchronization operations. Samples of data to be provided here are:

- A default repository or folder in PLM to locate the new parts to be created.
- A configuration specification to select the appropriate part versions in PLM system to be used. Usually, two choices are available: use latest versions or use versions compliant with a specific construction number.
- The construction number to be used if applicability by construction number option is selected. The value of construction number is obtained by default from FORAN DB directly.
- An optional engineering change notice number to be added in the transaction.
3.4.3. Locking FBUILDS nodes

In order to enable publishing and synchronization between FORAN and PLM trees and product structures, FORAN tree node must be locked, and this operation will first ensure that PLM system authorizes the operation. According to PLM system response, FORAN node locking will be enabled (if PLM modification access on the tree items is granted) or disabled (otherwise). If, for instance, a product structure item in PLM is checked out, it means that it is under modification, so we will not be allowed to lock the tree in FORAN for the same assembly, in order to avoid simultaneous edition both in FORAN and PLM systems. If the locking is enabled, all PLM items in the tree will remain frozen meanwhile. As well, if locking is enabled, we can be in the situation of that the tree has been modified in PLM and those changes have not been applied to FORAN yet. This case is also managed and, if produced, FORAN will not enable tree publishing until it is first synchronized from PLM.

The objective of this FORAN-PLM locking device is to prevent simultaneous editions on the same tree structure in both systems:

- If a tree is locked in FORAN, it cannot be edited in PLM and vice-versa.
- If the tree is up to date in PLM, then synchronization from PLM is not required and this option will be automatically disabled.
- If a tree has been modified in PLM, the changes have to be first applied to FORAN tree before publishing to PLM is enabled. FPLM will inform about the items affected in PLM tree in a message, Fig. 11.
When the user unlocks a tree node in FBUILDS, an instruction is sent to PLM system to unfreeze the assembly, so it could then be edited again. When finished, the system will show the following message:

![Message that the publishing data is updated to FORAN tree nodes and that PLM assembly has been unfrozen](image)

Fig. 12: Message that the publishing data is updated to FORAN tree nodes and that PLM assembly has been unfrozen

### 3.4.4. Model tree manager invocation

To invoke model tree manager, we just need to select the locked FORAN node, and choose the right button option called FPLM -> Model tree manager. The operation window appears. This window has just three elements:

- **Publishing to PLM** button. It will be disabled if a synchronization operation is required.
- **Synchronization from PLM** button. Once data has been synchronized, the publishing to PLM button will be enabled.
- **Generation of 3D Viewables** checkbox. If clicked, FPLM will generate 3D files that will be converted to native visualization format for the PLM used, if this conversion option is available for that PLM. Publishing process will then associate the visualization files to the published items into PLM system.

![Model tree manager invocation](image)

Fig. 13: Model tree manager invocation
Once the publishing operation has been launch, a message is shown to the user. Depending on the particular PLM used, publishing operation will be performed on-line or off-line.

In PLMs that perform on-line publishing, the message shown in Fig. 15 means that the publishing has been successfully performed. In PLMs with off-line publishing, this message will indicate that all the data has been successfully sent to PLM to be published. In order to be informed about publishing results, the user must unlock FORAN node and check node attributes. If the operation is not finished, the user may then click the Publishing data cleanup option, which will update publishing status on the involved FORAN nodes.

The Synchronization is performed always in on-line mode and, when finished, it will show a message indicating that the operation is finished.

4. PHASE III – FPLM Next Releases

4.1. Overview

For next FPLM releases, the target is to reach further significant milestones on FORAN to PLM integration, by providing a tight interconnection between the two systems. The most relevant features will be:

- **FORAN-PLM support table.** A FORAN table to register relevant data from PLM for all FORAN items. Will be the main PLM data repository for FORAN entities, and will store the unique PLM identifiers for FORAN items and other PLM data like maturity indicators and access control flags.
- **New PLM-to-FORAN synchronization mechanism.** A new mechanism will be introduced to synchronize FORAN data according to changes in PLM. It will be called FSYNC and will work on a scheduled basis: every X minutes will proceed with data synchronization for entities created, modified or deleted in PLM.

![Fig. 17: Process diagram for FSYNC module](image1)

![Fig. 18: Process diagram for FPUBLISH module](image2)
• **New FORAN-to-PLM publishing mechanism.** A new device will be introduced to periodically publishing new, modified or deleted FORAN items into PLM. Depending on the PLM used, it will include 3D graphic support. Data to be published will come from FORAN model tables and FORAN-PLM one.

• **FORAN items creation in PLM.** Some FORAN items could be first created in PLM, according to certain rules, and then their FORAN keys be transferred to FORAN through FORAN-PLM support table through synchronization mechanism.

• **FORAN items access control driven by PLM.** The FORAN-PLM table will store data for allowing/disallowing access control on FORAN items. This means that, for example, if we put some items in read only state in PLM, we could not modify the equivalent items in FORAN.

• **New FORAN Product Tree.** A new FORAN product tree structure will be automatically created and updated in PLM system, to exactly reflect the current existing FORAN data. This tree will be automatically edited by the new FORAN-to-PLM publishing mechanism.

• **Local document manager extension for additional items.** Some specific drawing types will have a dedicated mechanism for managing their document files in PLM, including BOM links generation. Those are:
  - Spools
  - Isometrics
  - Nestings

**References**

Software Applications for Life-Cycle Management at a Shipyard

Rolf Nagel, Flensburger Schiffbau-Gesellschaft mbH, Flensburg/Germany, nagel@fsg-ship.de

Abstract

Life Cycle considerations play an increasing role in the design of ships under the light of further increasing fuel costs as well as future improvements of sustainability of ships in general. Therefore, life cycle costs, life cycle assessment and life cycle performance, which all together can be summarized as life cycle management, have been introduced at FSG in the field of R & D as well as in the daily practice. The theoretical background will be highlighted as well as software applications to cover the challenging part of life cycle management of the ship design from the very beginning to operation.

1. Introduction

Life Cycle Management (LCM) is an integrated concept to assist in business managing the total life cycle of products and services towards more sustainable consumption and production patterns\(^1\). It describes a holistic management approach in order to combine the different perspectives into a business case for sustainability.

A number of tools are commonly used in practice and most of them are supported by more or less sophisticated software applications. These tools are in particular:

Life Cycle Assessment (LCA) puts focus on the environmental impact for a product from raw material extraction to final disposal (cradle to grave concept). LCA can also address the environmental impacts of a specific production process or an enterprise as shown in Fig. 2.

---

\(^1\) EPA Victoria :http://www.epa.vic.gov.au/Lifecycle/default.asp
Life Cycle Costing (LCC) is used to assess the costs of a project over its life time (typical for government investments) or the tool can be used to assess the benefit of an investment.

Life Cycle Performance (LCP) describes a concept in order to assess the performance of certain equipment over its lifetime.

Cost Benefit Analysis (CBA) is mostly used by governmental organizations to assess the benefit of new regulations or environmental projects.

Other tools like “Material Input per Service Unit” (MIPS) have been developed to identify the usage of resources over life time. It is a KPI (key performance indicator) that can be measured through a standardized approach and gives an indication whether the performance (from manufacturing over consumption to disposal) of a certain product can be improved.

As shown in Fig. 1, Thinking, Corporate Environmental & Social Responsibility, Management Tools as well as Data & Information play important roles in the business case for sustainability.

2. Perspectives of Life Cycle Management

2.1. Investment decision of ship owners

This perspective is triggered through the economical success of an enterprise. Ship owners have to make their decisions for a new investment (e.g. a new ship or an important retrofitting, e.g. scrubber technology in Emission Controlled Areas) on a rational basis. Typical financial key indicators are e.g. the return on investment (ROI), the internal rate of return (IRR) or the net present value (NPV). The net present value concept is most commonly used and will be explained in more detail in Section 3.

2.2. Rule Development (and sustainability)

IMO uses FSA guidelines, *IMO (2002)*, as a supporting tool for the rule development. Through the FSA guidelines, Cost-Benefit Analysis for assessing the different options to mitigate the monetary consequences from the new rules is used. The basis for a cost benefit analysis is the assessment of life cycle costs and benefits using the NPV concept.
The methodology can also be applied, if environmental issues are under consideration, Eide et al. (2009). In this context, life cycle cost evaluations on NPV basis have been made in order to assess the cost effectiveness of measures for CO₂ abatement options.

In order to assess the benefits correctly, further methods are available especially to account for the use of environmental goods (mostly public goods like air, water of the oceans, biodiversity etc.). In this respect, the US government has introduced the Contingent Value Method in order to create a rational basis for putting into force the “Oil Pollution Act”, which was introduced after the Exxon Valdez disaster with heavy impact on the design of tankers.

2.3. Decision Making of Government projects

Different approaches have been seen, when a government project shall be assessed in order to decide for the best project. Beside a pure economical approach on basis of the NPV (currently executed in a big navy vessel project), other assessment concepts incl. environmental & social factors lead to a scoring system, balancing between the economical perspective as primary part of the scoring and incentives for the coverage of environmental & social contributions. Such evaluation schemes have been used e.g. for a German Research Vessel project and a big Canadian Vessel intended for the support of military operations.

3. Calculation Methods

3.1 Net Present Value

The principle formula for NPV-calculation is:

\[ NPV = -C_0 + \sum_{t=1}^{T} \left(C_t \cdot \left(1+i\right)^{-t} + C_T \cdot \left(1+i\right)^{-T}\right) \]

For continuous cash flows, the formula can be adjusted by substituting the term \( \sum_{t=1}^{T} \left(C_t \cdot \left(1+i\right)^{-t}\right) \)

by \( \int_{0}^{T} C_t(t) \cdot e^{-\lambda t} dt \) with \( \lambda = \ln(1+i) \).

3.2 Cost Benefit Analysis

The principles of cost-benefit analyses, especially under consideration of the price of nature, are taken from Hanley and Barbier (2009). The basic formula is presented on page 6 as:

\[ NPV = \sum_{t=1}^{T} B_t \cdot \left(1+i\right)^{-t} - \sum_{t=1}^{T} C_t \cdot \left(1+i\right)^{-t} \]
\[ NPV: \quad \text{Net Present Value} \]
\[ T_t: \quad \text{Life Time under investigation} \]
\[ B_t: \quad \text{Sum of all benefits per period \( t \)} \]
\[ C_t: \quad \text{Sum of all costs per period} \]
\[ i: \quad \text{Interest (Discount) rate} \]
\[ t: \quad \text{Current index of period} \]
\[ (1 + i)^{-t}: \quad \text{discount factor} \]

The decision criteria for a project to be cost effective is the following condition:

\[ NPV \geq 0 \quad (3) \]

Other decision criteria like the internal rate of return (IRR) or cost benefit ratio may be used, but have drawbacks, Hanley and Barbier (2009).

There is a hidden discussion about discount rates in the public, although the different perspectives are easy to follow. Investments for enterprises shall give a good return on investment in a short time and therefore, discount rates in the range of 6-10% are common, but sometimes even higher. From a national economy (and sustainability) point of view, much lower discount rates (2-3.5%) are used in practice, valuing revenues in the far future much higher than the short term profit.

4. Practical Software Applications

4.1 Excel

Excel applications are used to solve simple problems without the use of too much resources. This has been tested at FSG in conjunction with retrofitting of scrubbers on existing designs or when calculating CO\(_2\) Marginal Abatement Cost Curves\(^2\) for Ro-Ro vessels. This approach is acceptable for a quick start, but difficult to handle for more complex problems. Errors are likely to happen, because all input data must be carefully retrieved and stored separately from the calculation algorithm itself.

With respect to above mentioned perspectives, a paint supplier obviously has developed such an Excel-Tool to assess the eco-efficiency of innovative underwater hull paintings and results are further published in\(^3\). Eco-efficiency is a performance indicator, which is linking the value of a service to the -reduced- environmental impact of creating this service. Further information on this concept can be found in WBCSD (2007).

4.2 LCA-Software

The Life Cycle Assessment is standardized in ISO 14040, as shown in Fig. 3. The core of the concept is to establish the Life Cycle Inventory Analysis by using specialised software. There are some commonly used products on the market like SimaPro\(^4\), GiBa \(^5\), Umberto\(^6\).

Having started with the process definition and defining the boundaries of the system or product under consideration, a full inventory of material end energy flow has to be created. This task is supported in various ways.

A network with input sources, transitions, connection points and output sources will be created. Fig. 4 shows an example for demonstration purposes.

\(^2\) Some background information see: http://www.dnv.com/resources/reports/pathwaystolowcarbonshipping.asp
\(^4\) http://www.pre.nl/content/simapro-lca-software/
\(^5\) http://www.gabi-software.com
\(^6\) www.umberto.de
A reference flow has to be defined (in the example above the total energy flow of fuel to main engines) and based on the process definitions within the transitions, the input of a transition will be processed to several outputs.

The principle concept behind this material flow and energy analysis is that of a Petri net (also known as a place/transition net or P/T net). For the analysis of material end energy flows, the principles of a Petri net have been adapted to represent the logical interconnection of parallel processes and their complexity.

For a more detailed analysis within the process, subnets can be created and these subnets substitute the previous defined transition. Thus very detailed investigations can be performed.

http://www.informatik.uni-hamburg.de/TGI/PetriNets/
A number of modules are available in a library with standard process definitions. This releases from definition of many time consuming processes. In order to calculate a CO₂ footprint for goods, manufactured in Asia and consumed in Europe, the transport by ship is stored in the library and corresponding energy consumption and related emissions will be calculated for this transport.

The interpretation as part of the LCA acc. to ISO 14040 will be supported by standardized methods like ECO Indicator 99, Method of Critical Volume, MIPS (Material Input per Service Unit), “estimation of effect” acc. to German UBA (Umweltbundesamt) and others.

Finally the results provide a complete analysis of the input & output of materials & energy and their impact. It can be identified, where waste of resources occurs and improvements can be discussed on a rational basis. In addition, based on the cost categories of the financial accounting, a link between the material & energy flow and financial impacts can be provided. Thus, also negative financial impacts can be discovered and further improvements in this respect can also be investigated.

4.3 costfact

The life-cycle cost module of the costfact® software package for shipbuilding application was used in a navy project at FSG to assess different design aspects in the early design phase from a life cycle perspective. Especially, some features relevant to the impact of future scenarios have been implemented such as sensitivity analysis and risk analysis.

In the following example, the building costs as well as all running cost incl. disposal costs have been identified and provided as input data over the life time. The Net Present Value is then calculated following Eq.(1). By varying the future fuel oil price, the difference in Net Present Value can easily be calculated and the possible financial impact on the project can be assessed. This allows more qualitative discussion between owners and a shipyard to identify strategies to improve the design for even better life cycle performance.

In the following example, the building costs as well as all running cost incl. disposal costs have been identified and provided as input data over the life time. The Net Present Value is then calculated following Eq.(1). By varying the future fuel oil price, the difference in Net Present Value can easily be calculated and the possible financial impact on the project can be assessed. This allows more qualitative discussion between owners and a shipyard to identify strategies to improve the design for even better life cycle performance.

![Fig. 5: Comparison at 3% increase of fuel price](www.costfact.de/lifecycle.htm)
Fig. 6 shows an example that deals with the risk that above mentioned increase of fuel oil price varies.

![Sensitivity analysis of fuel oil price](image)

The variation has been displayed not only for the fuel price but also for all other cost categories used in the investigation. The costfact life cycle module provides valuable features to improve the design itself and the communication with owners.

### 4.4 BESST LCPA-software

The BESST⁹ (Break Through in European Ship and Shipbuilding Technology) project is an EU research project and focuses on the Life Cycle Performance Assessment (LCPA) with respect to cost, environmental, safety and societal impact. A number of innovative technology concepts shall be developed and assessed by application of the newly developed LCPA software tool.

The first task was to identify a number of Key Performance Indicators, which are relevant to the design of future cruise vessels and Ro-Ro Passenger ships. While targeting on these KPIs from a top-down perspective, a principle flow of data/information to be generated in the design process was established, Fig. 7.

For the LCPA, as many components as necessary have to be defined in order to describe the innovation (on system level or on ship level). Within the components, the technical parameters have to be defined. The approach is fully modular and after a certain time of application, a number of components are available in the component database and can be used in different projects.

The vessels life cycle can be divided in as many phases as necessary to define the intended operating of the ship. Global values need to be defined for those variables or fixed parameters, which are relevant from an owners perspective (e.g. fuel oil prices, future revenues for cabins or lane meters) or which are physically relevant like e.g. conversion factor for carbon dioxide. The software tool is

---

⁹ The research leading to these results has received funding from the European Community's 7th Framework Programme (FP7/2007-2013) under grant agreement n° 233980
currently under development. First results are expected for the second half of 2011.

Fig. 7: Implementation of LCPA software tool

References


IMO (2002), Guidelines for Formal Safety Assessment (FSA) for use in IMO rule making process, MSC/Circ.1023, MEPC/Circ.392

WBCSD (2007), A Business Guide to Sustainability, United Nations Environmental Program
The Role of Social Computing in Shipbuilding

David Thomson, AVEVA Group Ltd., Cambridge, United Kingdom, david.thomson@aveva.com

Abstract

Social Computing is a new term for the practice of using computer applications and networks to gather and distribute information across social groupings such as teams, communities, organizations, and markets. Shipbuilding is a highly collaborative activity, and moreover an activity that requires the collaborative use of expertise and knowledge. Are there lessons to learn and benefits to be had from the current explosion in growth of Social Computing technology? This paper will provide a comprehensive introduction to the topic of Social Computing in the context of engineering, and will highlight the role Social Computing will play in the areas of communication and knowledge management within the shipbuilding enterprise.

1. Introduction

Social Computing is a wide ranging term covering a variety of software applications web sites, and general fields. In its most basic form Social Computing is the application of computing technology to support social interactions, notably when physical social interactions are not possible, either due to distance or other constraints.

On that premise we can say that the internet, email and instant messaging have been the fundamental building blocks of social computing, enabling the basic social interactions over any distance and to anybody with an internet connection. While this technology gave us a major leap forward in terms of one to one communication we all know that these technologies are limited in their application towards collaborative activities involving more than 2 people.

The Web 2.0 phenomenon has built on the basic communication building blocks making the step change away from relatively static pages presenting information to dynamic interactive sites where users not only contribute but can customise their whole experience according to their needs. Digital media, file sharing, forums, news feeds and the overall application of XML played major roles in this transformation. This paradigm shift engaged the public with the web in new ways; they were no longer just consumers but engaged users of sites and their own customised applications. With this came trust and openness and previously unheard of things such as buying food online, reviewing holidays, choosing music, making friends, organising events and even meeting a lifelong partner became possible.

The recent growth of social networking sites has cemented the shift in the way we communicate, Social Networks combined key web 2.0 capabilities in an easy to use interface and most importantly provided a large enough pool of users that you could be sure that a significant proportion of your friends, family and even colleagues would participate in the sharing, posting, poking, messaging and general socialising. The combination of access to a network of real people and the basic elements of communication such as messaging, instant messaging, feeds and groups allowed communities to naturally form within the networks and these communities are key to the power of Social Computing.

Shipbuilding is a highly collaborative activity, relying on a high number of discipline specialists to work together to produce highly complex one-off assets in extremely short time frames. Due to global economic pressures and the subsequent changes to the way shipyards work, the total number of engineers working on the design and build may grow, but each specialist may have to work with ever more, less experienced and less knowledgeable workers in ever more remote locations thus increasing dramatically the need to communicate, ask questions, inquire and gain understanding as one works.

Enterprise IT solutions have historically strived to build expert systems by capturing the ‘knowledge’
of workers to streamline processes and save time and money. This model has significantly improved efficiency in the shipbuilding process by automating time consuming, tedious and error prone tasks such as the production of drawings and parts lists, however there are some activities in the shipbuilding process that cannot be replicated by rule or algorithm and will always rely on the human factors of experience and intrinsic knowledge. Social Computing principles can be applied here, not to replace the knowledge of the worker but to enable the more efficient communication of knowledge and provide better overviews of the design, procurement and building processes.

2. Real world applications of social computing

The Wikipedia defines Social Computing as follows:

"Social Computing" refers to systems that support the gathering, representation, processing, use, and dissemination of information that is distributed across social collectives such as teams, communities, organizations, and markets. Moreover, the information is not "anonymous" but is significant precisely because it is linked to people, who are in turn linked to other people.

The following examples of Social Computing will demonstrate how Social Computing is being used extensively in today’s consumer world:

2.1 Social Proof

We know that humans are remarkably sensitive to the behavior of those around us, and in fact make countless decisions based on the current social context. Social Computing has enabled the extension of this concept to cover other forms of social interaction, for example decision making.

Unlike face to face interactions online interactions have historically deprived us of the signals that we instinctively use to make a judgment of trustworthiness. However Social Computing and the Web 2.0 mean that internet users leave such a visible trail of their activities that we can now use Google to build up a better picture of that individual, or establishment.

For example when choosing a hotel on booking.com we use the reviews of other users to help us decide if the hotel is really as good as the marketing makes it out to be. Or when we buy something from an eBay auction we scrutinize the ratings of the sellers to give us the feeling of security that they will deliver on time and a product that fits the description.

2.2 Mash-Ups

Essentially a Web 2.0 phenomenon, mash ups allow users to combine data from various specialist sources to provide better overviews of complex subjects and media. For example why write your own mapping application when other providers have specialised in this. Social Computing applications and websites usually provide APIs for the purpose of this kind of integration, the openness of participating applications being an underlying principle enabling this. The photo sharing site flickr, and the iGoogle home page are good examples of mash-ups.

2.3 Social Networking

Networking and communities are a key component of all Social Computing applications, therefore it comes as no surprise that the most prolific and successful Social Computing applications have been dedicated to creating social networks.

Social network sites provide various mechanisms to make connections, the most effective is to match certain attributes of their users, e.g. the name of the school they attended, their location or the brand of motorcycle they ride. This connection ability is enhanced on these sites by suggesting connections based on these common attributes. Connections can also be suggested based on the commonality of
contacts thus having a viral effect on the growth of a network. Social Networks play on our human inquisitiveness to compare with those around us, with profile pages providing a way of showing off our skills and achievements, which provides intriguing content for others to browse and consume.

Online Social Networks are one of the most useful and tangible forms of Social Computing as they reflect the way humans behave in the physical world, and simply extend this into the online community. Just as we differentiate our Social Networks in the physical world based on the closeness and nature of our connections, Online Social Networks are attempting to replicate this by allowing us to group, sort and filter our social networking behaviour according to its intended audience. The easiest way we can do this is to completely separate our networks by using different online networking providers, although Facebook is by far the most widely used Social network online, networks such as LinkedIn and XING are very popular and active in the business community.

2.5 Forming Communities

As well as providing a place to get connected online, Social Networks provide a convenient place to form communities to serve the purposes of their users. For example a group of people who are planning a holiday together, a team wishing to share photos from an event, users of a particular consumer product wanting to share experiences with it. Social Networks provide users with social objects, which act as a focal point to form communities around;

Groups can be open or subject to entry approval and usually provide at a minimum, a forum based discussion functionality. Events may offer up tools for planning, as well as a place for discussion and a place to share media related to the event.

Online communities have proved extremely powerful, with groups formed in Facebook being responsible for the current amnesty by the FARC, the democratisation of North African countries and the exposure of the vote fixing in the Iranian elections.
2.6 Staying Informed

Social Networks grouping so much online social activity into one place that theoretically it should be easier to stay informed about changes in one’s network. Social Networks had been designed with this in mind, providing email notification for the majority of activities within the network. Users were emailed every time they are added as a contact, every time they receive a message and unfortunately for almost every other activity in the social network.

The reality of this was that users were ‘drinking from an email firehose’, Niehaus (2011), and it soon becoming overwhelming. User’s behaviour and Social networking technology soon adapted, with users looking now exclusively to their social network inboxes to communicate directly with their contacts and Social Networks providing something commonly called the feed.

Feeds of web enabled content have been around since the early days of the internet, providing users with a stream of constantly updated content from the services they subscribe to. Social Networks have taken this concept further and often provide a single ‘River of News’ style feed automatically based on the relevant changes made to their network. The intention is that this constantly streaming feed is scanned every time the user goes to their home page, or the page for a group or event, and if they spot anything interesting they can dig deeper into the story in the feed by navigating further to the relevant page.

The content of the feed comes mostly from user or system generated status updates, a concept originating from the ‘away messages’ used in Instant messaging services, and now commonly referred to as microblogging.

These feeds can now be updated via mobile applications for smartphones and can act as aggregators for other microblogging services such as Twitter, which means that users can update their status wherever they are. The effect of which is real time updates which can be particularly interesting when it concerns major events.
3 Principles of Social Computing

We know the definition of Social Computing, and have seen its application and power to change in the above examples. But what are the underlying principles?

3.1 People are the focus

All Social Computing applications are focused around people; users must have the freedom to generate an online or in-system presence and act in human ways, each user must also be able to customise their experience in a Social Computing environment tailoring how they share and consume information to suit them.

3.2 Sharing

Sharing is a natural human behaviour, serving many purposes. In Social Computing applications the sharing of information is key to generating interesting content for other users. Social Computing applications actively encourage this by suggesting or automatically sharing activity with others who may be interested. Privacy is also a natural human behaviour the social computing application must be able to exhibit enough controls to make less gregarious users feel comfortable in the social network with confidence.

3.3 Connectivity

Social Computing relies on the easy ability to connect everything, people connecting to other people, events and other focal points connecting communities. One way it does this is by making everything an object, with its own profile page. Just as individuals have a network so will events, groups and other community objects.

3.4 The Technology does the Work

The major factor to Social Computing success in the consumer market is that Social Computing applications offer an easier way to execute certain social behaviour. Feed provide us with an easier way to stay in contact, friend suggestions do the work of looking for old school friends or colleagues, and intelligent algorithms suggest events and even products for us based on our preferences.

4. The Effects of Social Computing

Over the last 10 years there is no doubt that Social Computing has had permanent effect on the world we live in and many of the things we do there. Some of these effects have been related to the media, many to the way we interact, here is my summary of those effects.

4.1 Ambient Information

Microblogging, feeds, river of news, the ‘fire hose of email’ are presenting us with information overload, more often that not this means we are only able to scan the information coming to us which can mean missing important details. However we are adapting to it and we are getting very good at scanning feeds. The term Ambient Intimacy has been coined by Blogger Leisa Reichelt, who admits that her friend’s twitter updates and microblogs may just appear as pointless noise to some, but argues that they help her stay in touch, and gain a certain intimacy with those who may have otherwise remained as acquaintances. In the world of Social Computing communications has become so prevalent and accessible that it is actually ambient around us. ‘It surrounds us wherever we want it, not necessarily when it wants us. We dip into it whenever we like LAURIE (2010).

Ambient Intimacy, or in the enterprise Ambient Information is still a very new effect, and channelling and filtering this will be key to reducing information overload. However it is here that Social
Computing differentiates itself from traditional collaboration software, applying the right Social Computing tools in a collaboration environment can result in information transfer by osmosis, and a better all round knowledge of what is happening, within a community, network or organisation.

4.2 Transparency

The sharing principal of Social Computing has resulted in an increase in transparency, early file sharing sites succeeded because of the lack of control and governance, while more formalised and legal media sharing sites like YouTube and flickr provided a platform to show off and sharing was definitely a positive capability. Over time users have realised that sharing and being transparent within ones own network is a positive trait, building trust in the user and in return more sharing from other users. It is clear that, as with physical interactions, introverts and extroverts exhibit very different level of communication with those around them in Social Computing applications, however the overriding need to be approved by those around us and the need for a social context has generally shifted us toward more transparency.

4.3 Social responsibility

A useful side effect of this transparency has been what I call social responsibility, as we all share and open up our lives in Social Computing applications, it is very clear that those in our Social Networks can observe us in more detail than ever before. Social Networks are not the place for a private life, and as the number of divorces originating from Facebook testifies. As our Social Networks grow it becomes increasingly difficult to keep a private life. We must either accept that we don’t want to participate, or take responsibility for who we are and how we behave.

4.4 Community Building

Historically communities were defined by their common location, even as this evolved in to the groupings of people with similar interests, beliefs or needs we were limited by location. Social Computing and the internet break those boundaries allowing us to form cohesive communities regardless of location. More importantly Social Computing enhances the impact of these communities by taking advantage of the networking effect to grow communities faster than would otherwise be possible. Larger communities tend to have more power to achieve their common needs.

5. Social Computing and Shipbuilding

Social Computing is changing the way we work, the way we interact at an amazing pace. A few years ago Social Computing was almost exclusively the domain of the younger generations but now in developed countries we see that all ages and generations have adopted Social Computing and expect this kind of technology in every aspect of their lives, so why not their work?

One problem has been the historical back ground of Social Computing in the consumer market, this focus on applications for the individuals private benefit has given corporations the negative impression of Social Computing application as being time wasting and distracting, although this has been the case we also see that individuals are choosing to use Social Computing applications as business tools. The most obvious example is Social media being used as a new marketing channel, but users are also turning to Facebook, LinkedIn and twitter to stay in contact with their business contacts and create forums for discussions. There is a clear need to apply this technology to the workplace creating focused applications for specific business purposes. The following sections highlight some of the reason for this.

5.1 Globalization

We are passing through the age of globalization, an economic trend summed up by this quote from IBMs chairman Samuel J. Palmisano
‘When everything is connected, work moves... the work of business and the work of technology. Work flows to the places where it will be done best - that is, most efficiently and with the highest quality. It’s like water finding its own level. The forces driving it are irresistible.’

As a result we an interesting redistribution of work, where German design offices, work for Dutch ship owners on new super efficient container ships which will be built in Chinese yards specializing in lower cost production. Or in the case of cruise ships we see 80% of the total value of such vessels consisting of materials and outside services, in other words shipyards are shifting from multicraft companies to assembly factories., Kanerva (2005),

This has the ultimate effect that, physical communities that are essential for the communication of information no longer exist. At the very best telephone calls or costly meetings can be arranged, all of which fail to give the same richness as the old way of sitting in the same office space, speaking the same language, with an openness that comes with people you know at a deep personal level.

5.2 The Social Network already Exists

Social networking and social behaviour already play an important role in any organisation. In fact in any organisation there are two very different organisations. The formal organisation, defined by the relative positions in a hierarchical organisation chart, and the informal organisation shaped by day to day interactions between employees.

The informal organisation can be easily observed; at the coffee machine, during the lunch break and at team events. The informal organisation is cross functional, the connections one has in this network can be highly productive, for example knowing that developer just that little bit better may get you that crucial fix earlier. Or having a high visibility in the company network may get you noticed when it’s time for promotions or redundancies.

As well as there being two distinct types of organisations within an enterprise, there are distinct communities. For example in the AVEVA development world there will be a distinct group of people interested in databases, not just developers but also customer representatives, sales persons, implementers, trainers and so on, more often than not this ‘community’ will only have the chance to interact in one to one email communications or if they are lucky at an event where people can meet face to face.

Social Computing can be used to channel and enhance this natural behaviour of community forming and sharing, especially in the globally distributed enterprise.

Fig. 3: Old School Social Networking
5.3 The Need for Information

Individuals and communities will naturally seek out information to do their job, since the advent of Social Computing, the way we seek out this information has changed. Information transfer has traditionally been achieved via a corporate push mechanism, via emails, newsletters briefings and so on, however due to Social Computing we have become accustomed to pull the information we need to us, either in the form of configurable feeds, or by proactively seeking out what we want in the network. Increases in this behaviour can be seen in globally distributed companies where the intranets, shared calendars and gossip are highly regarded sources of information.

5.4 The Need for Recognition

Engineers are not particularly well known for their desire to share and distribute their precious expertise and knowledge; however they do have a need to be recognised for their productivity and ingenious work. Many engineers choose not to be in managerial or supervisory positions and therefore have limited channels of influence. Social Computing can provide these people with the tools to form their own communities where their expertise and value will be recognised. ‘One of the keys to a successful social media site is to let the community police and manage themselves’, Derry Vaughan (2010)

5.4 Knowledge Management

As the shipbuilding work force ages, the ability to attract the best young engineers diminishes, and outsourcing increases, retaining the knowledge of shipbuilding processes and techniques becomes an ever more important challenge for design offices and yards.

Information, knowledge and the application of it are very subjective, and many argue that the notion of capturing it and managing will be impossible to achieve. However most of us know that the major restriction to learning that new skill is the availability of that same information, the knowledge how to apply it and the time to practice. While shipyards in developing economies focus on gaining the knowledge to improve efficiency and processes, the yards in knowledge rich economies must fight hard to retain their know-how and expertise, and thus their ability to offer higher value products and services despite being least competitive on cost.

Knowledge management in its most basic form involves documenting procedures and processes and making it readily available. Documenting procedures was the most efficient way to capture knowledge when data storage cost was significant, however it did bring with it a huge overhead in the documentation itself and the maintenance of that data. The Web 2.0 world and Social Computing has taught us that capturing and storing huge amounts of unprocessed data has more value, reducing the overhead to capture in the first place and proving just as useful when combined with instant searching and intelligent algorithms which learn what kind of data we are looking for.

A new term called Personal Knowledge Networking has emerged from the Social Computing phenomenon, where the shift is away from formalised, top down Knowledge Management, towards informal or grass roots Knowledge Management, instead of accessing a centralised server, users use a combination of Social Computing tools to find what they want, where the natural community forming abilities, and support bring clear benefits.

6. PLM, the Integrated Shipbuilding Solution and Social Computing

Product Lifecycle Management (PLM) is a product development strategy focused on managing the information generated and consumed during the lifecycle of that product. PLM systems typically need to bring together data from the various Enterprise IT systems such as CAD, PDM, ERP, and CRM before they are highly customized to provide functionality to support the multiple business processes that rely on data from all of the above systems.
If we look at the needs of the shipbuilding industry, studies show that the business focus is on effective control and execution of design and production activities. Many of the world’s most highly developed yards have already optimized their planning, logistics, design and production processes with a combination of home grown applications and best of breed software components. However these solutions are costly to maintain and difficult to integrate with other Enterprise software components. At the other end of the scale new green field shipyards are looking for rapidly deployable enterprise solutions to meet their rapidly growing business needs.

AVEVA have recognized the importance of planning and bill of material management in the support of these activities and recently made acquisitions to address this. Now with the best of breed design, engineering and manufacturing system, a shipbuilding specific ERP functionality for planning and materials, and a proven PLM backbone. AVEVA are able to offer the Integrated Shipbuilding Solution. This solution will intelligently integrate data from the planning, design, engineering, procurement and manufacturing processes to provide rapidly deployable solutions to better direct, control and execute the major shipyard processes.

AVEVA helps to integrate these business systems on a common “Digital Information Hub”, powered by AVEVA NET, to allow you to easily capture all of the activities across all departments working together to design and build the vessel. In fact AVEVA’s propriety database technology has always been able to do this. As designers work, the delta of their actions is saved to the database, this advanced capability provides a history of the design activities and is used to analyze changes in the model or drawings, or can allow a roll back of the whole project. In the same way transactions in AVEVA planning and materials applications can be published to the AVEVA NET backbone providing a similar historical record.
If we apply Social Computing principals to the integrated shipbuilding solution we can transcend the integration of data offered by traditional PLM systems and focus on what people do, how they do it and how they interact with others to do it. Social Computing will not focus on a specific business process, but the integration of activities in an enterprise to achieve the enterprise aims and objectives.

6.1 Social Computing and AVEVA NET

AVEVA NET has at its core the concept of one page per object, which is configurable according to the customer needs. These object pages display information relevant to the object type, for example major equipment may show a summary of its technical attributes, its location on a drawing or diagram the connected pipes, or cables and even a 3D preview of it.

Fig. 6: AVEVA’s Object web concept delivered by AVEVA NET Portal

Fig. 7: A document overview in an AVEVA NET prototype
Whereas for a document it would be more appropriate to show the approval workflow and revision history on its page.

This concept is easily extended to cover Social Computing aspects, where each organization, be it a formally organized team or self-organized community, has their own page displaying data related to their activities and topics of focus.

If we also apply the Social Computing principal that the people provide the content and the technology does the work, we can imagine that a suitable filtered feed could be provided from the useful information captured by the design system and Integrated Shipbuilding System.

Fig. 8: An engineering feed that could be automatically generated from the Integrated Shipbuilding System

Fig. 9: Mock-up of a home page for a Hull Panel with an automatically generated feed
The feed could be applied to a person, a team, and event or an engineering object, where configuration would allow control of what data is automatically fed to each home page. For example, the feed to the home page for a model object such as a Hull panel would contain information relevant to that object and closely related objects such as production drawings, or logically connected items such as stiffeners or plates.

While a user may configure their personal feed to provide information about data or documents they have created, modified or otherwise subscribed to.

Integrating with planning systems and even office calendars we would bring in planned activities or events as social objects, and in the same way each event could have a feed of information related to it. In this case we could see if resources have been allocated, changes to the logistics been made and even changes or updates to the sub tasks of the activity. If we extend this concept to include Enterprise data from the planning and material procurement processes we can quickly build up a very rich and useful repository of information, where the use of a mash-up style interface can be used to provide a dashboard for management.

In summary the concept of applying Social Computing to AVEVA NET is, to capture automatically the human interaction with all activity in the Integrated Shipbuilding System, i.e. activity related to plans, production and design activities, catalogs, components, lists, design assemblies, parts and documents. And then to enable the natural social behavior required to effectively propagate, consume and analyze this data.

7. Findings

Clearly we are at a very early stage in investigating the application of Social Computing in our industry, and we do not have much concrete data. However the application of Social Computing in the engineering work place has proven to be an extremely interesting and controversial topic. On the one hand business leaders are aware of the changes Social Computing has brought to many areas of our lives and can’t ignore the growth in use of sites like facebook. On the other hand Facebook is responsible for most of the negative connotations we have around Social Computing, especially regarding useful use of time, and privacy.

However without doubt leaders recognize the importance of networking, both inside and outside an enterprise, and are willing to fund this in the form of team meetings, conferences and kickoffs. As shipbuilding becomes ever more distributed and teams have to collaborate remotely, the need for collaboration supporting tools is clear. Social Computing is so pervasive and successful as a basis for collaborative activities that major software companies have began to implement Social Computing principles in their internal collaboration and enterprise software. Sometimes this is referred to as Enterprise 2.0 and in most cases is the addition of Blogging, Wikis and instant messaging capability in the tools set.

AVEVA’s database technology and AVEVA NET based Integrated Shipbuilding Solution provides a unique opportunity to take Social Computing in Enterprise software even further by introducing the concepts of ambient information and natural community forming, how these concepts will be used and benefited from is not yet proven, however from our studies we can easily imagine the following effects:

7.1 More Access to Information

The data that I have shown in the simulated feeds is already captured in the AVEVA design products and AVEVA NET, however it can only be consumed if special reports are configured and then studied. The concept of an automatic feed provides contextual information where it is expected to be found.
This may result in some information overload to begin with but with user controllable filtering there is no doubt that user will be able to access more relevant information in the system without having to resort to telephone call or even meetings. This access to information could result in many things, less time searching for data, more time reading it, the generation of ideas and innovation, new patterns of work and ways of collaborating.

7.2 Improved Communication

Being exposed to the human element of our work stimulates communication, Social Computing should enhance the social interactions needed to do collaborative work by doing some of the groundwork for us. For example when we can see in a feed who has been working on a particular tasks or object we know who to contact to resolve any problem we have with that item, and if instant messaging or microblogging are enabled, we have even more convenient options to open the communications channels with those persons. The feeds in a way are doing the communicating for us.

7.3 Increased Productivity

Aside from the obvious productivity benefits coming from access to more relevant information, it can be argued that exposing a certain amount about the work individuals are doing can have a positive effect on productivity. Certainly there will be those engineers who will actively seek out ways get updates that enhance others perception of their good work/professionalism/ninja skills. It is also more than likely that the concept of social responsibility will play a role in how aware users are of others perception of their work.

7.4 Knowledge Management

As mentioned before huge amounts of activity and data are captured by enterprise IT systems even today, however extracting value form it requires significant effort. Applying the Web 2.0 principle in a Social Computing context means that the information is exposed in a relatively unstructured way but in a way which the social groups using it will self organize for their own benefit. I believe that consumer Social Computing will lead the way in how we can get the best from this mass of data, for example Google products like Google analytics, Google books and many applications available in Google labs.

7.5 Object Centric or People Centric

During this study, the initial focus was to make the system people centric, as Social Computing principals suggest. However engineers clearly work with objects or tags which are the focus of their activity. Applying Social Computing in the engineering field could result in confusion unless the paradigm is consistent.

For example People Centric:  

Joe Blogs has approved the Purchase Order.

Or Object Centric:  

The Purchase Order has been approved by Joe Blogs.

Our initial study shows that people react in a different way to these paradigms, notably PLM systems have historically been object centric and engineers may have come accustomed to that. More study is required to fully understand the effect of this.

7.6 The Social Network as an Enterprise Platform

The amazing growth of Facebook has allowed it to grow form a social network to a social platform, the Facebook platform is the basis for networking, entertainment in the form of movies, games and apps, and is quickly becoming the basis for all communication between certain groups of people, in fact project ‘Titan’ aims to integrate webmail so that we could in theory capture a more significant
proportion of any individuals total electronic communications. We can easily see how this could evolve into a form of operating system.

If the Social Computing is applied to the work place it is very likely to grow in use as Facebook has, and in the same way becomes a platform for much more than networking and communications. Considering it as a platform, perhaps as a part of the PLM strategy, before applying to Shipbuilding would then make sense, including integration with all major business systems in enterprise. Applying Social Computing must be an enterprise level decision, effecting company culture and behaviour significantly.

7.7 Social Computing as a Business System

Most of the criticism of Social Computing I have heard is related to its misuse and a lack of understanding how best to use the technology. Complaints of pointless updates and too much information would not be the case in a business environment where some form of guidance would be provided in the implementation of Social Computing. Moreover much of the misuse of Social Computing in the consumer market comes from 3rd party commercial applications which abuse the Social Networks to gain more revenues, applied to a business environment, any apps would be or a productive nature, for example an add on to organise meetings, or create dashboards.

Another common criticism of Social Computing in private use is the ability to present yourself in a favourable light by filtering the information appearing in your profile, this not revealing the full picture. I would argue that we all do this to an extend in our public lives, especially in the work place Social Computing only being limited by our normal human behaviour.

7.7 Conclusion

Clearly there are sceptics, many believe that Social Computing is reducing our attention spans and reducing our ability to form normal relationships by using our social skills. Others quite rightly complain that Social Computing has eroded our privacy. However, like any change we are still getting used to this new application of technology. During this study it has become clear that the Web 2.0 phenomenon and Social Computing have resulted in a new approach to technology and many inventive features that are highly applicable to Enterprise IT systems and information management.

As Forrester research says:

'Social Computing is not a fad. Nor is it something that will pass individuals or corporations by. Gradually, Social Computing will impact almost every role, at every kind of company, in all parts of the world. Firms should approach Social Computing as an ongoing learning process, using some of the best practices of firms that have successfully taken the first steps'.

Those first steps were taken mostly by consumer product companies, who have embraced the Social Media as a new marketing channel, however it now seems that Social Computing is inevitable as an Enterprise concept with the major IT players such as Cisco and IBM having mature full spectrum offerings and CAD specific providers offering socially enabled collaboration software. Startup Vuuch have gone further still offering what they call an Enterprise Social System for Manufacturing, which takes the best of the features from Social Computing sites and applies them in an innovative way to the discrete manufacturing industry.

Shipbuilding is a conservative business market and it could be that it is too early to begin a full on Social Computing approach, however our study of Social Computing has highlighted that automatically generated feeds and the collaborative aspects of Social Computing would be hugely beneficial in today’s globalised Shipbuilding environment. PLM systems tend to have 3D, or business processes at their core, Social Computing systems have people at their core, unlike data, people are self organizing and self optimizing.
References


KANERVA, M. (2005), Advanced PLM solutions in ship engineering, IBM PLM Shipbuilding Workshop, Hamburg

KIRKPATRICK, D. (2010), The Facebook Effect, Simon & Schuster


Vision-Based Underwater SLAM for the SPARUS AUV

Josep Aulinas, Institute of Informatics and Applications, Girona/Spain, jaulinas@eia.udg.edu
Yvan R. Petillot, Oceans Systems Laboratory, Edinburgh/UK, Y.R.Petillot@hw.ac.uk
Xavier Lladó, Institute of Informatics and Applications, Girona/Spain, llado@eia.udg.edu
Joaquim Salvi, Institute of Informatics and Applications, Girona/Spain, qsalvi@eia.udg.edu
Rafael Garcia, Institute of Informatics and Applications, Girona/Spain, rafa@eia.udg.edu

Abstract

An overview of underwater SLAM implementations as well as submapping SLAM approaches is given in this paper. Besides, the implementation of the so-called selective submap joining SLAM on the SPARUS AUV is presented. SPARUS carries a down-looking optical camera. The information gathered by this camera is run through SLAM, together with on-board navigation sensors, producing a precise localization of the vehicle and a consistent final map. Experimental validation on a real dataset is described, showing a promising performance of our implementation.

1. Introduction

In the last decade, different underwater vehicles have been developed in order to explore underwater regions, especially those of difficult access for humans. The use of Remotely Operated underwater Vehicles (ROVs) is very common, however, ROVs require a link, i.e., a tether, to the ship in order to be operated by a person aboard of the ship. The tether is a group of cables that carry electrical power, video and data signals back and forth between the operator and the vehicle. In order to avoid the need for a tether, several research groups and developers focused on developing Autonomous Underwater Vehicles (AUVs). AUVs are equipped with on-board sensors, which provide valuable information about the vehicle state and the environment. Combining this information with control algorithms makes the vehicle fully autonomous.

Some examples of these AUVs are the ICTINEU (see Fig. 1) and the SPARUS (see Fig. 2) developed by VICOROB research group at the University of Girona. Both vehicles were developed with the purpose to participate in the Student Autonomous Underwater Challenge – Europe (SAUC-E) competition. ICTINEU won the 2006 SAUC-E edition, while SPARUS is a more recent development, torpedo shaped AUV, which won the 2010 SAUC-E edition. SPARUS is the vehicle we used to test the method presented in this work.

Fig.1: Picture of ICTINEU AUV operating inside the test water tank at the Underwater Robotics Research Center (CIRS – Girona, Spain).
Some widely used sensors for land and aerial robots do not work or are not precise enough underwater. For instance, the use of cameras is difficult due to the lack of visibility and scattering; the laser range finders are imprecise working in these scenarios because of light attenuation; and GPS signal does not work underwater. The sensors used on SPARUS AUV are the Inertial Measurement Unit (IMU) and the Doppler Velocity Log (DVL) to measure navigation data, while a down looking camera is used to gather data from the environment. The IMU and the DVL do not give absolute localization, therefore if the vehicle is wrongly localized, nor the IMU neither the DVL will provide useful information to recover the right position. In addition, as the positioning is relative to past information, the localization problem is biased and the measurement noise produces drift. On the other hand, the detection of salient features in this environment is a complex task, since camera images are noisy. Noise together with lack of other navigation aids makes the task of mapping and localization a difficult challenge.

A solution to the lack of GPS signal and the presence of noise are the Simultaneous Localization and Mapping (SLAM) algorithms. SLAM algorithms aim to build an approximate map of the area and calculate the approximate position of the vehicle within this map. In order to do so, SLAM algorithms combine the information coming from all sensors. Our SLAM approach, called the selective submapping SLAM, uses navigation readings to improve vehicle localization, and the map through its correlation with the vehicle position. To have a SLAM algorithm working properly, we need to select robust landmarks, i.e. objects, rocks and other salient elements. These robust landmarks must be easy to observe when seen for a second time, and easy to associate with previous observations. This procedure is important to close a loop, i.e. revisiting an area, because closing a loop means a reduction on the uncertainty and a more consistent final map.

In what follows, a background on underwater SLAM implementations is first given in Section 2. Afterwards, a summary on existing SLAM algorithms working with submaps is presented. Section 3 describes the implementation of our approach on SPARUS. Section 4 presents the experimental validation, while Section 5 gives the conclusions.

2. Background

This section surveys main existing SLAM implementations for underwater applications, focusing on the filtering technique used to handle noise and drift uncertainty, the main sensor used to gather data from the scenario, and the type of feature used to build the map. Afterwards, submapping SLAM approaches are summarized.
2.1. Underwater SLAM

Several approaches tackle the localization problem on known scenarios. Some approaches use GPS-aided localization, Caiti et al. (2005), Erol et al. (2007), but the attenuation of electromagnetic waves through the medium of water limits the application of GPS to near surface activities, or otherwise forces the vehicle to visit often the surface to recover its position. A standard for bounded xyz navigational position measurements for underwater vehicles is the Long-BaseLine (LBL) acoustic transponder system, Hunt et al. (1974), Olson et al. (2006).

The equivalents to GPS underwater are the acoustic transponders, such as LBL or Short-BaseLine (SBL). These positioning systems have limited range, accuracy and an associated cost of deployment. LBL operates on the principle of time-of-flight and it is been proven to operate up to a range of 10 km, Whitcomb et al. (1999). The main drawback of LBL is that it requires two or more acoustic transponder beacons to be tethered to the sea floor. SBL systems provide more accurate positioning information, but suffer from the same drawbacks than the LBL. Recently, several AUVs use Ultra Short-Baseline (USBL) technology, which consists of a transceiver, usually placed on the surface, on a pole under the vessel, and a transponder mounted on the AUV. This technology is more accurate than LBL and SBL. Another set of approaches avoid the use of external devices by using computer algorithms. For instance, the use of particle filters for AUV localization presented in Maurelli et al. (2008). This approach is shown to work with high performance. However, it only works when the map is known a-priori.

When the map is unknown, SLAM is conducted. Underwater scenarios are still one of the most challenging scenarios for SLAM because of reduced sensory possibilities. Underwater SLAM approaches have many problems due to the unstructured nature of the seabed and the difficulty to identify reliable features. Many underwater features are scale dependant, sensitive to viewing angle and scale. A SLAM proposal tackles the problem using point features, Williams et al. (2004). This approach proposed to fuse information from the vehicle’s on-board sonar and vision systems. They use EKF based SLAM combined with sonar and vision to obtain 3D structure and texture (see Fig. 3).

Leonard et al. (2001) and Newman et al. (2003) also used point features. The former implemented the decoupled stochastic mapping and performed tests on a water tank, while the later proposed the constant time SLAM and used LBL information to help on the localization. On the other hand, non-feature based approaches to SLAM using bathymetric information were presented by Barkby et al. (2009) and Roman et al. (2007).

![Fig.3: Terrain models built by projecting the texture of the visual images onto a surface model generated by sonar data, Williams (2004)](image-url)
A particle filter is used to handle the uncertainty in the navigation solution provided by the vehicle, *Fairfield et al. (2008)*. This approach was successful in minimizing the navigation error during a deep sea mapping mission. The method was capable of providing real-time localization, with comparable results to the ones given by SBL and USBL. A vision-based localization approach for an underwater robot in a structured environment was presented in *Carreras et al. (2003)*. The system was based on a coded pattern placed on the bottom of a water tank and an on-board down-looking camera. The system provided three-dimensional position and orientation of the vehicle along with its velocity. Another vision-based algorithm, *Eustice et al. (2008)*, used inertial sensors together with the typical low overlap imagery constraints of underwater imagery. Their strategy consisted on solving a sparse system of linear equations in order to maintain consistent covariance bound within a SLAM information filter. The main limitation on vision-based techniques is that they are limited to near field vision (1-5m), and also deep water mission will require higher amounts of energy for lighting purposes. In a previous works, *Eustice et al. (2005)* and *Eustice et al. (2006)*, they presented the reconstruction of the RMS Titanic from a set of images and using IF. Using Sparse Extended Information filter (SEIF) and forward-looking sonar, *Walter et al. (2008)* presented a SLAM approach to inspect ship hull.

Instead of vision, *Ribas et al. (2008)* used mechanically scanned imaging sonar to obtained information about the location of vertical planar structures present in partially structured environments. In this approach, the authors extracted line features from sonar data, by means of a robust voting algorithm (see Fig. 4). These line features were used in the EKF base SLAM.

![Fig.4: Abandoned marina SLAM example, using imaging sonar, image extracted from Ribas et al. (2008). Top left plot shows the superposition of imaging sonar readings, based on dead reckoning trajectory. Top right plot is the same superposition but in this case after filtering the trajectory through EKF base SLAM. Bottom picture shows a satellite image of the abandoned marina, with the lines representing the boundary between water and land. The trajectory of the vehicle is plotted, using dead reckoning estimates (dashed line) and using SLAM algorithms (solid line).](image-url)
In Tena-Ruiz et al. (2004) side-scan sonar was used to sense the environment. The returns from the sonar were used to detect landmarks in the vehicle's vicinity. Observing these landmarks allows correcting the map and vehicle location; however, after long distances the drift is too large to allow associating landmarks with current observations. For this reason, they proposed a method that combines a forward stochastic map in conjunction with a backward Rauch-Tung-Striebel (RTS) filter to smooth the trajectory.

Underwater SLAM implementations have some points in common, for instance, imaging sonar is widely used, the most common filtering technique is the Extended Kalman Filter (EKF) and point features are commonly used to represent the map. Some approaches use side-scan sonar or optical cameras, which seems to become more important as technology advances. The use of EKF based SLAM has been shown to handle uncertainties properly; however, the computational cost associated with EKF grows with the size of the map. In addition, linearization errors accumulate in long missions, increasing the chance of producing inconsistent mapping solutions.

2.2. Submapping SLAM

The use of submaps has been shown to address both issues, linearization errors and computational costs, at the same time, thereby improving the consistency of EKF based SLAM, Castellanos et al. (2007). An early example of this strategy is the decoupled stochastic mapping, Leonard et al. (2001), which uses non-statistically independent submaps. As a result, correlations are broken and inconsistency is introduced into the map. The constant time SLAM, Newman et al. (2003), uses multi-overlapping local submaps with the frame referenced to one of the features in the submap. This technique maintains a single active map and computes a partial solution, independently. However in non-linear cases the consistency is not proven.

Different techniques, such as the constrained local submap filter, Williams et al. (2004) or the local map joining, Tardós et al. (2002), produce efficient global maps by consistently combining completely independent local maps. The main idea behind this approach is to build maps of limited size and then, once completed, merge these small maps to a global one. The so called atlas SLAM, Bosse et al. (2004), consists of a hierarchical strategy that achieves efficient mapping of large-scale environments. They used a graph of coordinate frames, with each vertex in the graph representing a local frame, and each edge representing the transformation between adjacent frames. In each frame, they build a map that captures the local environment and the current robot pose along with the associated uncertainties. The divide and conquer SLAM, Paz et al. (2008), uses the divide and conquer strategy from fundamental graph theory. The hierarchical SLAM, Estrada et al. (2005), consists on a lower (or local) map level, which is composed of a set of local maps that are guaranteed to be statistically independent, and the upper (or global) level, which is an adjacency graph whose arcs are labeled with the relative location between local maps. An estimate of these relative locations is maintained at this level in a relative stochastic map. Every time the vehicle closes a loop a global level optimization is performed, producing a better estimate of the whole map. Conditionally independent SLAM, Piniés et al. (2008), is based on sharing information between consecutive submaps so that, a new local map is initialized with a-priori knowledge.

3. Implementation on SPARUS AUV

SPARUS is equipped with several sensing devices: Doppler velocity log (DVL), inertial measurement unit (IMU), down-looking camera, forward-looking camera, imaging sonar and GPS (see Fig. 5). In this work, only DVL, IMU and down-looking camera are used, producing information about velocities, orientations and about the sea floor. The SLAM approach used here is the so called selective submap joining algorithm, Aulinas et al. (2010). The main idea of this approach is to use EKF based SLAM to build local maps \((x_i, P_i)\), where \(x_i\) is the state vector describing vehicle’s pose, vehicle’s velocities and the map, while \(P_i\) is its associated uncertainty. The size of these local maps is bounded by the total number of features and by the level of uncertainty. The relative topological relationship between consecutive local maps is stored in a global level map \((x_G, P_G)\). The global level
is used to search for loop closure (H_Loop), i.e. the vehicle is revisiting a region. The loop closing strategy involves a decision on whether to fuse local maps depending on the amount of found correspondences between submaps. The whole process is presented in Algorithm I, and detailed in Aulinas et al. (2010).

The main novelty in this implementation as compared to the one presented in Aulinas et al. (2010) is the use of an optical system as the main environment sensor unit. Therefore, the observation model is redefined in order to match with a camera model. In this case, the inverse depth parametrization is used, Civera et al. (2008).

**Algorithm I: Selective Submap Joining SLAM**

begin mission

while navigating do

\( \tilde{x}_i, \tilde{P}_i = \text{EKF SLAM}() \rightarrow (\text{Build submap } M_i) \)

\( \tilde{x}_G, \tilde{P}_G = \text{build global map}(\tilde{x}_i, \tilde{P}_i) \)

\( H_{Loop} = \text{check possible loops}(\tilde{x}_G, \tilde{P}_G) \)

for \( j = H_{Loop} \) do

refer \( M_i \) and \( M_j \) to a common base reference

\( H_{ij} = \text{data association}(\tilde{x}_i, \tilde{P}_i, \tilde{x}_j, \tilde{P}_j) \)

if \( H_{ij} > \text{threshold} \) then

\( \hat{x}_{ij}, \hat{P}_{ij} = \text{map fusion}(\tilde{x}_i, \tilde{P}_i, \tilde{x}_j, \tilde{P}_j, H_{ij}) \)

\( \hat{x}_G, \hat{P}_G = \text{update global map}(\hat{x}_{ij}, \hat{P}_{ij}) \)
endif

endfor

endwhile

4. Experimental validation

Experimental validation was done through the data acquired by SPARUS during a survey mission. The mission consisted of navigating an area of about 20m×20m, in a grid of 5m×5m. Vehicle's depth was almost constant around 17 meters. The total navigation time was about 17 minutes. The vehicle carried a down-looking camera that acquired a total of 3199 images, Fig. 6. Experimental results obtained with SLAM show that there is a significant improvement on trajectory estimate as compared to dead reckoning, Fig. 7.
In order to test the performance of SLAM using submaps, a subset of random 2D points were extracted from a mosaic of the scene, Garcia et al. (2006). These 2D points were then back referred to the image they belonged. This subset of points was used, instead of automatically detecting features. The performance of our SLAM implementation using this set of points is shown in Fig. 8. This figure shows a sequence of 5 frames containing first one landmark, and later on two landmarks. In addition, the uncertainty projected on the image plane is drawn, decreasing consistently after being observed for the second time. Fig. 9 presents a top and a frontal view of the resulting map and trajectory. In these views, one can see vehicle's trajectory corrected with SLAM and the landmark location, as well as its associated uncertainties. Finally, Fig. 10 shows a 3D plot of these results.
Fig. 9: Different views of the results produced by SLAM. On the top, presents a top view of the execution, while on the bottom a frontal view is presented. On the left, landmark uncertainties are drawn, while on the right, only landmarks are shown. In all plots, vehicle trajectory is drawn.

5. Conclusions

The main contribution of this paper is a SLAM implementation for an underwater vehicle, SPARUS AUV. First, the most representative underwater SLAM implementations were surveyed, reaching the conclusion that Extended Kalman filter is widely used for this sort of applications. However, Extended Kalman filter suffers several limitations that can be addressed by using submaps. For this reason, a summary of the state-of-the-art on submapping approaches was presented. A SLAM algorithm is then briefly introduced, and adapted for its use on the SPARUS AUV. Experiments done with real data show a bounded effect of the linearization error, a precise trajectory estimates, and a three-dimensional map reconstruction. Besides, the observation model for a down-looking optical camera was introduced. This model was based on inverse depth parameterization. Experiments conducted in a real unstructured environment demonstrated that SLAM improves vehicle trajectory in comparison to dead reckoning. Moreover, SLAM combined with inverse depth parameterization was capable of producing a consistent map.

Acknowledgments

This work was partially funded through the Spanish Ministry of Education and Science (MCINN) under grant CTM2010-15216 and the EU under grant FP7-ICT-2009-248497.
References


ESTRADA, C.; NEIRA, J.; TARDÓS, J.D. (2005), Hierarchical SLAM: real-time accurate mapping
of large environments. IEEE Transactions on Robotics 21/4, pp.588-596.


Control-Oriented Modeling of an Underwater Wave Glider

Andrea Caiti, Vincenzo Calabrò, Sergio Grammatico, Department of Energy and Systems Engineering, Pisa/Italy, {caiti, v.calabro, s.grammatico}@dsea.unipi.it
Andrea Munafò, Integrated Systems for the Marine Environment, Genova/Italy, a.munafo@gmail.com
Mirko Stifani, Italian Navy (CSSN) Department, La Spezia/Italy, mirko.stifani@marina.difesa.it

Abstract

This paper introduces a control-oriented modeling approach for a hybrid autonomous underwater vehicle: the Underwater Wave Glider. Thanks to a non-conventional design, the vehicle can accomplish both surface and underwater tasks, by changing its shape. The vehicle can operate as a wave glider at the sea surface, where the potential energy of the waves and the solar radiation can be exploited to recharge the onboard batteries, while moving at low speeds. Moreover, the Underwater Wave Glider can switch to the typical torpedo-shaped configuration in order to operate both as a Glider and as a self-propelled Autonomous Underwater Vehicle. The vehicle dynamics are modeled via a non-standard Lagrangian approach and some simulations are provided.

1. Motivation

During the last few years, examples of clean-energy powered systems have been developed around the oceans of the world, for instance, by efficiently using the wave potential energy. First, the Pelamis, Jones (2008), a power plant system, converts the wave energy into electric power by using the relative motion between its parts. This idea has also been successfully employed in ships and autonomous vehicles. The vessel Suntory Mermaid II, Geoghegan (2008), reached the Hawaiian coast starting from Japan, after 108 days of navigation, travelling for about 6400 Km. This vessel was powered by solar panels only for electronic and communication devices. It uses a smart spring-fin system which converts the wave potential directly into propulsion.

The Wave Glider (WG) autonomous vehicle, Manley et al. (2010), completed in only 82 days a travel of 4000 Km from Hawaii to San Diego, California. This vehicle is also equipped with solar panels to power the on-board electronics, while the propulsion is achieved with a submerged multiple-wings system connected with the surface float part through a cable. The idea developed in this paper is to combine the interesting and innovative solutions included in these three systems, into a single hybrid Autonomous Underwater Vehicle (AUV) characterized by the WG endurance, together with the typical underwater exploration and navigation capabilities of AUVs and GLIDERs. Commercial hybrid AUV/GLIDER vehicles, Alvarez et al. (2009), exist but, they suffer by the usual energy density problem and they cannot achieve complete autonomy and long endurance requirements.

2. Underwater Wave Glider concept

Given the above experience and the parallel advances in the field of AUVs, including oceanographic gliders, it becomes natural to investigate the feasibility of an Underwater Wave Glider (UWG) vehicle capable to combine the properties of both AUV/GLIDER and WG vehicles into a single compact and clean-energy powered system. In this paper we consider the control-oriented modeling of possible configurations of such a vehicle. In particular, we consider an UWG equipped with solar panels, a variable buoyancy main body and arms, floating wings, which combined in different configurations, allow the vehicle to change its shape and consequently its operating mode, Fig. 1.

The dynamical model of torpedo-shaped underwater systems has been established in the literature, Fossen (2002), and recently complex models have been proposed for a hybrid class of AUV/GLIDER with variable mass and Center of Gravity (CoG), Caiti et al. (2010). However, during the WG mode, since there are no variations of the overall mass and of the CoG position, the resulting model can be
derived from the standard form of Fossen (2002), with the exception of body-fluid interactions. In order to derive a dynamical model for the longitudinal plane motion (surge, heave and pitch), a Lagrangian approach is chosen.

3. Direct kinematics

The reference frames of the classic Denavit-Hartenberg (DH) convention, Denavit et al. (1995), are used, Fig. 2. The base reference frame is assumed to be North-East-Down (NED), as usual for the standard navigation frame in marine robotics. The corresponding DH parameters for the UWG are summarized in Table I. Each row of the DH table represents a two-dimensional roto-translation transformation of the kind:

\[ T_i^{i-1}(d, \theta, \alpha, \alpha) = \begin{bmatrix} \cos(\theta) & -\cos(\alpha)\sin(\theta) & \sin(\alpha)\sin(\theta) & a\cos(\theta) \\ \sin(\theta) & \cos(\alpha)\cos(\theta) & -\sin(\alpha)\cos(\theta) & a\sin(\theta) \\ 0 & \sin(\alpha) & \cos(\alpha) & d \\ 0 & 0 & \cos(\alpha) & 1 \end{bmatrix} \]  

(1)
There is an additional transformation to recover the end-effector reference frame, which has origin coincident with the wing aerodynamic center, Fig. 3, which is the point where all hydrodynamic forces, generated by the relative motion between the wing and fluid, can be assumed to be applied.

The end-effector position $p_e^b$ and orientation $R_e^b$ in the base frame $\{b\}$ are described by the transformation

$$T_e^b(q) = T_1^b T_2^b (p_e^a) T_3^b (p_d^a) T_4^b (q_1) T_5^b (q_2) T_6^b (q_3) = [R_e^b(q) \ p_e^b(q)]^T,$$

$q = [p_e^a, p_d^a, q_1, q_2, q_3]^T \in \mathbb{R}^5$ is the vector of the generalized positions and the final roto-translation $T_e^b$ is obtained by the extra DH row with $d_6 = 0$, $\theta_6 = 0$, $a_6 = 0$, $\alpha_6 = \pi/2$.

4. Differential kinematics

To relate the velocities $\dot{q}$ of the (virtual) joints to the velocity of a particular point $p_e^b$, the approach of the geometric Jacobian can be used:

$$J^b(q, p_e^b) = \begin{bmatrix} \cdot & \cdot & \cdot \\ J_{p_1} & \cdots & J_{p_n} \\ J_{q_1} & \cdots & J_{q_n} \end{bmatrix},$$

where

$$\begin{cases} J_{p_i} (q, p_e^b) = \begin{bmatrix} 0 \\ 0 \\ z_{i-1} \end{bmatrix} & \text{if joint } i \text{ is (P)} \\ J_{q_i} (q, p_e^b) = \begin{bmatrix} z_{i-1} \\ 0 \\ z_{i-1} \wedge (p_e^b - p_{e,p_i}) \end{bmatrix} & \text{if joint } i \text{ is (R)} \end{cases}$$

Table I: DH parameters for the UWG

<table>
<thead>
<tr>
<th>d</th>
<th>$\theta$</th>
<th>a</th>
<th>$\alpha$</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>$\pi/2$</td>
<td>0</td>
<td>$\pi/2$</td>
</tr>
<tr>
<td>$p_{e,n}$</td>
<td>0</td>
<td>0</td>
<td>$-\pi/2$</td>
</tr>
<tr>
<td>$p_{e,d}$</td>
<td>$-\pi/2$</td>
<td>0</td>
<td>$-\pi/2$</td>
</tr>
<tr>
<td>0</td>
<td>$q_1$</td>
<td>$a_1$</td>
<td>0</td>
</tr>
<tr>
<td>0</td>
<td>$q_2$</td>
<td>$-a_2$</td>
<td>0</td>
</tr>
<tr>
<td>0</td>
<td>$q_3$</td>
<td>$-a_3$</td>
<td>0</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
<td>$\pi/2$</td>
</tr>
</tbody>
</table>
shows the different contributions in case of prismatic (P) joints or rotational (R) ones. Based on the definition of the Jacobian matrix (3), we have

\[ \dot{v}_b^p(q, \dot{q}) = \dot{p}_b^p = J^b(q, p_b^p) \dot{q}, \]

(5)

\( \dot{p}_i^b \) and \( \omega_i^b \) are, respectively, the linear and the angular velocities of the point \( p_i^b \).

Since the hydrodynamic effects act on a single part of the multi-body system, we have to consider the local velocities, that, in a general frame \( \{u\} \), are given by the equation

\[ v_u^e(q, \dot{q}) = \begin{bmatrix} 0 \\ R_u^w \end{bmatrix} J_u^b(q, p_u^b) \dot{q} = J_u^b(q, p_u^b) \dot{q}. \]

(6)

Finally, we remark the duality property valid for a generalized force \( f_e^b \) applied at the point \( p_e^b \), that corresponds to the joint generalized force

\[ \tau(q, p_e^b) = J^b(q, p_e^b)^T f_e^b. \]

(7)

5. Vehicle dynamics via Lagrangian modeling

The dynamics equations are here derived via a standard Lagrangian approach, based on the differential equation

\[ \frac{d}{dt} \left( \frac{\partial L}{\partial \dot{q}} \right) - \frac{\partial L}{\partial q} + \frac{\partial F_d}{\partial q} = \tau^T \]

(8)

\( L = T - U \) is the Lagrangian function, \( T \) is the kinetic energy, \( U \) is the potential energy, \( q \) is the configuration vector, \( \tau \) is the vector of the generalized forces and \( F_d \) is a Rayleigh-like dissipation function used to model the viscous effects acting on the serial-chain joints.

To model the dynamics of a vehicle moving in a fluid, also added mass, hydrodynamic damping and restoring forces have to be considered as shown in the next subsections.

The hydrodynamic forces, the wave potential and the generalized forces generated by the wings and by the flexible joints are included in the term \( \tau \). This is an interesting novelty because the hydrodynamic effects of each body part, in terms of added masses and centripetal contributions, Fossen (2002), are considered as applied in each local reference frame and then projected into the whole body. The hydrodynamic forces on each body part are then modeled following the classic analytical approach of Chwang (1975). The same approach is used for the generalized forces generated by the underwater wing.

5.1 Kinetic energy

Let \( p_{mi}^b \) be the centre of mass of the link \( i \), having mass \( m_i \) and inertia \( I_{mi} \) with respect to the reference frame \( \{i\} \). The kinetic energy of the link \( i \) is

\[ T_i(q, \dot{q}) = \frac{1}{2} m_i \dot{q}^T J_{pi}^T(q, p_{mi}^b) J_{pi}(q, p_{mi}^b) \dot{q} + \frac{1}{2} \dot{q}^T J_{\Omega_i}^T(q, p_{mi}^b) R_i^b(q) I_{mi}^b R_i^{bT}(q) J_{\Omega_i}^T(q, p_{mi}^b) \dot{q}. \]

(9)

The notation \( J_{\cdot}^\cdot \) is used to indicate the Jacobian that takes into account the contribution of the velocities up to the link \( i \) of the serial chain.
Remark 1. The first two joints are modeled as virtual, in fact they have null masses and inertias.

Finally, the total kinetic energy is obtained by summing all the contributions of the single links:

\[ T(q, \dot{q}) = \sum_{i=1}^{5} T_i(q, \dot{q}). \]  

(10)

5.2 Potential energy

The potential energy describes the effects of restoring forces, such as buoyancy and gravity, and additional terms due to the wave-body contact. In particular, the potential energy of the link \( i \) is

\[ U_i(q) = (\rho V_i - m_i) g_0^{\text{bT}} s^b_{mi}, \]  

\[ g_0^{\text{bT}} = \begin{bmatrix} 0 & 0 & g \end{bmatrix}^T, \]  

where \( g \) is the gravity acceleration, \( V_i \) the volume of the link \( i \), \( \rho \) the fluid density.

The wave potential energy is modeled as a two-spring system, one acting on the (vertical) heave axis and one along the pitch axis, because it is quite natural to assume that the vehicle is not rigidly constrained on the sea surface. Therefore the potential energy of the waves can be modeled as follows.

\[ U_w(f) = \frac{1}{2} k_d (p_d^b - f_d^b)^2 + \frac{1}{2} k_i (q_i - f_i)^2, \]  

(12)

where \( k_d, k_i \) are positive constants to be identified and \( f = [f_d^b, f_i]^T \).

A realistic way of modeling the wave profile is given by considering a sinusoidal wave of amplitude \( A_w \) and frequency \( \omega_w/2\pi \):

\[ f_d^b(t) = A_w \sin(\omega_w t), \]  

\[ f_i(t) = \arctan(A_w \omega_w \cos(\omega_w t)). \]  

(13)

Obviously, the total potential energy is given by the sum

\[ U(q, f_w) = U_w(f_w) + \sum_{i=1}^{5} U_i(q). \]  

(14)

5.3 Drag modeling

The viscous friction acting on the joints is modeled as a linear function of the joints velocities:

\[ D \dot{q} = \left( \frac{\partial F_d}{\partial \dot{q}} \right)^T. \]  

(15)

Instead, the sum of linear terms and quadratic ones of the kind \( \tau_h = -D_h(q) \dot{q}, \tau_{h^2} = -D_h(q) \dot{q} \dot{q} \) is considered for the hydrodynamic effects.

Remark 2. Since it is expected that the UWG navigates at the sea surface with low speeds, it is possible to assume that the body-fluid relative motion is dominated by a laminar flow. This assumption simplifies the drag characterization of each link. In fact, the small Reynolds numbers that characterize the laminar condition allow to model the drag effects with high accuracy.

5.4 Dynamic model

Considering the contributions shown in the previous subsections, the following dynamic multi-body model can be derived.
\[ B(q)\ddot{q} + C(q, \dot{q})\dot{q} + D(q, \dot{q}) \dot{q} + g(q, f_w) = \tau + J^{BT}(q, p^b_e) f^b_e, \]  

(16)

where the drag matrix is \( D(q, \dot{q}) = D_q + D_h(q) + D_{h_d}(q, \dot{q}). \)

In the dynamics equation (17), the end-effector wrench generated by the wing \( f^b_e(R^b_e, R^b_w, v^b_e, v_w) \) is actually due to the end-effector orientation \( R^b_e \) and velocity \( v^b_e \), besides the wave front orientation \( R^b_w \) and velocity \( v_w \).

6. Simulation results

The adopted control strategy reflects the specification of having a clean-energy powered vehicle. In fact, by using flexible joints, the vehicle is passively controlled during WG mode and the exceeding energy can be collected as in [W.D. Jones, 2008] to recharge the batteries. Therefore, the use of solar panels and the conversion of the wave potential energy provide an ideally unlimited endurance for the UWG.

The passive control strategy for the wave glider configuration is simulated over the typical wave profiles of the Mediterranean Sea, Cavalieri (2004). In the case shown in Fig. 4, after the initial transient, the UWG navigates over a wave profile of amplitude of 0.5 m and frequency of 0.1 Hz.

The vehicle floating part is chosen with an ellipsoidal shape with major axis width of 1 m. A two-wing configuration, based on the NACA 0009 profile, Raymer (1992), having a dimension of 0.3 m x 0.5 m for each wing, is used.

The numerical simulation shows a surge speed of about 1 knot, which is comparable with the declared speeds of the Wave Glider, Manley et al. (2010), in similar sea wave conditions.

![Mean Speed = 0.665610 (m/s)]

![Vertical Position (m)]

Fig. 4: Surge speed and heave position.
7. Conclusion and future work

A Lagrangian modeling approach is presented for a novel class of hybrid Underwater Wave Gliders is proposed. The benefit of the proposed vehicle design is that the vehicle can autonomously accomplish both surface and underwater missions, with an ideally unlimited endurance.

This work may be improved in many directions. An ongoing line of research is focused on the optimization of the shape parameters of the vehicle, for instance to maximize the nominal speed in the wave glider operating mode.

Future work will be spent for the physical realization of a prototype of Underwater Wave Glider and for experimental testing.

References


Noise Limits in Harbour – A Method to Optimise the Noise Emission of Vessel's Ventilation Systems

Clemens Koechert, Nordic Yards Holding GmbH, Wismar/Germany, Clemens.koechert@nordicyards.com

Abstract

Environmental pollution of ships will become more and more important in early design stage of a vessel. Noise requirements of port authorities were tightened and result in stricter requirements of vessels design, especially for cargo ventilation systems of RoRo and RoPax vessels. A method has been developed to compare the noise prognosis of different supplier’s. Under use of optimisation algorithms, this evaluation method has been extended to an efficient tool for the use in early design stage.

1. Introduction

Most noise requirements on ships aim at the comfort of the passengers and crew. For those requirements, noise predictions and measurements procedures were developed and established. The growth of harbours and the increasing population of harbours cities result in a conflict of increasing number of ships in the ports and the noise limits in urban living quarters. Because of this, the focus of the ships outside noise level in context of environmental noise calculation in port areas will become more and more important.

1.1 External noise of ships

For shipyards, three different problems have to be solved:

- Noise Prediction
- Noise Evaluation
- Noise Measurements

Noise predictions have to be done in the early design stage to give noise limits to the supplier of ventilation systems and insulation. With the data received by suppliers, first evaluation of the expected noise levels will give an impression if the noise level can be reached. In an early design stage, necessary changes in geometry of the ventilation ducts can be implemented and the risk of not achieving the noise requirements can be minimised. The noise measurement itself is not easy to perform because of high influence of environmental conditions around the shipyard. Normally shipyards do not have a suitable area without buildings to perform noise measurements on quayside of the ship so that measurement corrections have to be taken into account or the measurement has to be taken on sea side. Requirements with low noise limits can have the problem that the background noise level generated by wind and harbour working is quite in the same range of noise limits. This results in short time slot to perform those measurements.

2. Noise calculation

For noise calculations in open fields the used formulas can be reduced to the addition and distance calculation of noise levels. Because of the wide range of noise pressure, between $2 \times 10^{-5}$ Pa as minimum hearing possibility and 630Pa for starting jet, a logarithmic representation is used for noise calculation. Noise pressure level is calculated with a reference noise pressure $p_0 = 2 \times 10^{-5}$ Pa to represent the noise level in dimensionless decibel (dB):

$$L_p = 20 \log \left( \frac{p}{p_0} \right) = 10 \log \left( \frac{p}{p_0} \right)^2$$
Typically more than one noise source has to be taken into account for calculation. This gives the need for logarithmic addition of noise levels:

\[ L_{\text{tot}} = 10 \log \left( \sum_{i=1}^{N} 10^{L_i/10} \right) \]

Two noise sources of the same strength increase the total noise about 3dB. With a higher difference between the noise levels, a smaller increase of the higher noise level will be generated, Fig. 1. This gives the possibility that a single noise level will dominate the total noise level.

The hearing capability of humans gives a range between 16 Hz and 20 kHz. For calculation and measurements, the hearing capability can be subdivided into thirds of octaves to give a better representation of the results. The total noise level of a noise source will be calculated with a logarithmic addition of every frequencies noise level. To give a better impression in the hearing capability of human ear, noise levels are typically weighted with an A-filter, Fig. 2. Human ears do not record the noise linear over the whole frequency band, so that typically an A-Band filter will be used for correction.

With the correction value of every frequency, a weighted noise level can be calculated:

\[ L(A) = 10 \log \left( \sum_{i=1}^{N} 10^{(L_i+\Delta_i)/10} \right) \]
To calculate noise outside of a vessel, distance calculations of noise levels in free field have to be done. The noise behaviour according to distance is presented with the noise distance law $p_{dB} = 10 \log \left( \frac{r}{r_0} \right)$, which results in a noise pressure calculation in distance $r$:

$$L_p = L_{p0} - 20 \log \left( \frac{r}{r_0} \right),$$

with $r_0 = 1$ m. Every doubling of the distance to the noise sources results in a noise reduction of about 6 dB, Fig. 3.

![Fig. 3: Noise reduction caused by distance to noise source](image)

### 3. Implementation of calculation method

The prediction of noise outside a vessel is a challenge that can not be easily solved. To simplify the noise calculations, the complex problem can be divided into two separate problems:

- Noise calculation inside the vessel to determine the noise level at the outlets of the vessel
- Calculation of the noise level in measurement distance under use of the noise levels at the outlets

The determination of noise levels at the outlets of the ventilation systems requires detailed calculations of ventilator fans and damper systems noise levels and also calculations of the flow induced noise levels in ventilation ducts. These calculations will not be focus in the presented method.

For the calculation system outside the ship, a calculation method was first developed to evaluate the noise prognosis by different suppliers. This calculation method was verified by measurement of the vessel and extended with an optimisation routine to define noise limits at the ventilation openings by given noise level in measurement distance.

### 3.1 Method to evaluate different supplier's noise predictions

Noise prognosis reports for shipyards normally were prepared by the suppliers of ventilation systems, engineering offices or others. The Shipyard has to compare different reports with heir own experiences. In the reports, typically basic information can be extracted as the noise level of the emitter, e.g. openings of ventilation systems and exhausts, and a noise level in some points at measurement distance from the ship. The calculations by the suppliers have always some differences in the used emitter, the position of the selected receiver points in measurement distance and there calculation implementation. Because of this, a direct comparison of different predictions is not easily possible, Fig. 4.
The shipyard is in need to present the ship owner a validation of the different results. To give a direct impression of the results of different predictions, it is necessary to perform calculations of the receiver noise level under use of the emitter noise level taken from the supplier’s predictions. Receiver points will be place in measurement distance besides the vessel and the noise level $L_{R}$, generated by every emitter noise level $L_{E}$, can be calculated:

$$L_{R} = L_{E} - 20 \log \left( \frac{r}{r_{0}} \right)$$

With a reference distance $r_{0}$ of 1m and an individual calculated distance $r$ between receiver and emitter point. In every receiver point a total noise level has to be calculated under use of all relevant noise emitters:

$$L_{R_{tot}} = 10 \log \left( \sum_{i=1}^{N} 10^{L_{R_{i}}/10} \right)$$

Under use of receiver points in measurement distance, a calculated noise level curve can be displayed and compared with the supplier’s prediction, Fig. 5.

![Fig. 5: Comparison of two supplier predictions](image)
3.2 Verification of the calculation method

Noise evaluations with supplier’s data sets have the disadvantage, that the results in measurement distance can only be as good as the prediction of the emitter’s noise level at the vessels outlets. Because of this it is not sufficient to measure only the noise level in measurement distance, additional measurements of the emitters are necessary to verify the prediction.

During a shipyard project it was possible to take a whole data set of noise measurements. Measurements of the noise emitters at 1m distance were taken as well as noise levels in the specified distance of the vessel. In the results, a good correlation between the noise measurements in measurement distance and the noise calculation with the results of the 1 m distance measurements can be found, Fig. 6.

In the area of the HVAC, the calculation results are a higher than the measurements because the influence of diffraction at the ships geometry was not taken into account. The direct track between noise emitter and receiver point is blocked by the ship and the effect of diffraction will give an additional noise damping.

A second problem that influences the measurement results is the background noise level of the harbour area. The background noise can increase the noise level up to 3 dB, if the background noise and the noise coming from the emitter are on the same level.

3.3 Prediction of noise limits in 1m distance under use of optimisation algorithm

An early prediction of the noise levels in 1m distance of the vessels emitter outlets is of importance to give the supplier the possibility to optimise their systems. A maximum allowed noise level in a specified distance is given by the port authorities and has to be fulfilled. An optimisation method was included to generate results in short time. As objective function, the sum of all noise emitters \( L_{Ei} \) was chosen and in every receiver point constraints were defined that the receiver noise level \( L_{R} \) should not be higher than the maximum noise level:

\[
\max \sum_{i=1}^{n} L_{Ei} \quad \text{subject to} \quad L_{R}(L_{Ei}) \leq L_{R\text{max}}
\]

In every receiver point, the noise level has to be calculated under use of all relevant emitters and their
specific distance:

\[ L_R(L_{Ei}) = 10 \log \left[ \sum_{i=1}^{n} \left( \frac{L_{Ei}}{10} \right)^{20 \log \left( \frac{r}{r_0} \right)/10} \right] \]

To give an example of the noise prediction method, a noise dimensioning calculation of a RoPax ferry will be presented. The calculation model uses 104 different noise emitters and on every side of the ship are 20 receiver points located. In this case two different noise cases, bow-loading and stern-loading of the vessel, have to be taken into account so that there are 80 receiver points in total. This is quite too much to handle the problem manually, but the numerical optimisation calculation takes less than one minute.

All information of importance will be represented in plots of the results. A plot of the vessels contour and the position of the noise emitters give an impression of the geometric situation. The noise emitters are collected in different groups like engine room or cargo hold ventilation. Results of the predicted noise emitters in 1m and the resulting noise level in measurement distance of every emitter group are plotted as well as the resulting total noise level and the allowed noise level, Fig. 7.

![Fig. 7: Prediction of maximum emitter noise level at 1 m distance](image)

### 4. Conclusions

A fast method for noise prediction and evaluation was developed and verified. First results of measurements show that the method in general delivers results that can be used in daily work on a shipyard. The approach to keep the method as simple as possible tends to a good usability. The prediction of the emitter noise level show a result that is conservative in comparison to the yard measurements. For first predictions the method gives an excellent result that can be used to estimate the possibility of reaching the given limits and specify suppliers noise levels.

In the noise evaluation calculation, especially when first measurements of the installed systems were taken, an extension of the method will be useful. In actual version, the method uses the total noise
level of every noise emitter and no information about the frequency band is used. Because of this, no influence of diffraction and interference is calculated. Taking this into account, a more precise result can be achieved.

References


Evaluating the Feasibility of Open Source CFD Software for Small Ship Design Offices

Colin Barth, University of Applied Sciences Bremen, Bremen/Germany, colin.barth@conoship.com
Jan Jaap Nieuwenhuis, Conoship Int. B.V. Groningen/The Netherlands, janjaap.nieuwenhuis@conoship.com
Andreas Kraus, University of Applied Sciences Bremen, Bremen/Germany, andreas.kraus@hs-bremen.de

Abstract

In this paper we present the results of an investigation of the feasibility of Open Source CFD software for small ship design offices. CFD calculations are more and more often used to analyze and evaluate ship designs, amongst others to compare hull form variants, or to optimize part of the hull, like the design of the bulbous bow. Most of the smaller ship design offices sub-contract these calculations to specialized institutes or companies, especially when the number of hull shapes for which an analysis is required annually is relatively low (below 5-10). However Open Source CFD programs might provide a good opportunity for small offices to do some CFD work in-house, as one of the drawbacks of commercial CFD tools, their high licensing costs, can be avoided. This paper shows the potential of one Open Source CFD solver (Open Foam) and a number of pre/post processing tools. The paper also describes some of the problems and the effort required to develop a working system which provides useful results.

1. Introduction

Computational Fluid Dynamics (CFD) is a widely used tool in modern ship design. In earlier days CFD was mainly done by specialized consulting offices and “knowledge institutes” like model test basins. The increased capacities of “normal” computers however provides the opportunity for more and more companies, like shipyards or design offices to perform CFD calculations themselves Bertram (2000). But, most of the smaller design offices still sub-contract the CFD calculations to specialized consulting offices. There are a number of reasons why subcontracting CFD calculations remains common for smaller companies:

- The number of times a CFD calculation is required annually is rather low, which does not justify the high licensing and/or maintenance costs;
- In a small design office there is no room for a “full time” CFD engineer. One, or a number of designers will have to do CFD calculations besides other work. It is questionable whether the relatively low number of calculations is sufficient to maintain their CFD skills and provide ample opportunities to keep up with all new developments;
- Many people expect that the computing capabilities of normal office PC’s are insufficient for CFD calculations.

 Nonetheless, Conoship International, a small design company from the North of Holland specialized in the design of innovative short sea vessels, Fig. 1, was interested to see whether it is possible to do in-house CFD calculation, despite the limitations of a small design office with respect to workforce and financial means. The main reason to consider doing CFD work ourselves, is the fact that hull form design is a large part of our work and is getting more and more important, as the design problems Conoship works on become more critical and “extreme” with respect to fuel consumption and ship’s speed. Therefore more and more CFD calculations are required, and in an earlier phase of the design process in order to define the feasibility of a project. Sub-contracting CFD work substantially affect the speed and progress of the design process, as many consulting offices can’t start the work at the moment Conoship would like them to start. Furthermore subcontracting the calculations also requires a lot of effort from Conoship’s side, as a great deal of communication is required before and after the calculations.
At the start of this project there was no experience at Conoship with running CFD calculations, we were however confident that our experience with respect to evaluating CFD results and improving hull forms accordingly, would help us to develop the required CFD modeling skills and to evaluate the various available programs. To investigate the feasibility of in-house CFD calculations first an extensive investigation of commercially available programs was made (both potential theory and RANSE). A number of these programs provided the capabilities Conoship requires, but the annual licensing costs or the acquisition costs were, after a long and thoughtful contemplation, considered to be too high. The investigations did however strengthened our belief that running in-house CFD calculations should be feasible. This resulted in a decision to shift our focus to the potential of Open Source CFD tools, avoiding the high licensing costs, but realizing that a large effort would be required to get a running system providing useful results. However by joining forces with the Hochschule Bremen the necessary “workforce” was arranged to carry out this project.

1.1. Aim of the project /desired results

The aim of the project is not to completely replace the work of the specialized consulting offices, but to reduce the number of iterations required whenever CFD calculations are sub-contracted, by doing the first evaluations in-house. In general, when we sub-contract calculations, 3 to 6 hull form variants are analyzed, starting with an overall hull form optimization, an evaluation of the bow form and ending with RANSE calculations for the entire hull with a focus on the aft ship and the appendages. In total this takes several weeks, up to more than a month. The intention is to sub-contract at least one calculation, as a validation of our in-house work, but also giving us the possibility to discuss the hull lines and potential improvements with a number of experts.

At this moment it is clearly not our intention to do complex calculations, such as evaluating the performance of a rotating propeller in nozzle while considering the actual aft ship form of the vessel. As a start, the results of the selected open source CFD tools should thus be comparable to the results we currently use to evaluate and optimize the overall hull design and the bow form in particular. I.e. the output should be comparable to the output of programs like RAPID (MARIN), or v-SHALLOW (HSV A) and at least provide us with:

- A wave profile around the hull;
- A wave pattern around and aft of the vessel;
- A pressure distribution around the hull.
In the (near) future we would like to be able to check for flow separation in the aft ship and evaluate the wake field of a design.

1.2. Requirements for the Conoship CFD Tools

Besides being able to produce the desired results, the selected CFD tools should meet a number of other requirements:

- All tools should work on “normal” office PC’s, like a dual-core computer with 2GB RAM;
- All program should run under Windows/Linux;
- Calculations should be reasonably fast, as the calculations are to be used to compare various hull form variants in an early design stage. I.e. calculations times of a week are not acceptable, calculation times of 24-48 hours might be acceptable, especially when the calculations can be run in a batch mode, for example over a weekend.
- Preferably the selected tools should be able to run on a computer network, providing the possibility to run calculations during the night on multiple computers in the office.

As a start, only open source, or other freely available programs were considered, but when commercial meshing or post-processing tools might provide large advantages, acquiring these relatively cheap programs is considered to be an option.

1.3. Followed Approach

In order to make a selection of the various tools and to develop a system that provides useful results the following approach was followed (roughly corresponding to the approach as described by Bertram (2010)):

- Perform a literature search identifying the advantages and disadvantages of the available tools and check whether they have been used for any ship design cases;
- Get familiar with a number of tools and options;
- Run example cases given in the manuals or on the internet to “experience” the possibilities, requirements and drawbacks;
- Select a set of most suitable tools and settings;
- Try to reproduce a documented ship design case, like the “Wigley case”, from the CFD-Online Forum and optimize the settings, meshing, etc.;
- Try to reproduce the calculations made by a specialist for one of Conoship’s hull designs. This involves a large number of iterations, optimization of settings, etc.;
- Try to reproduce the calculations made for another Conoship hull form, with the defined set of “optimal” settings (still to be done);
- Evaluate the trend predicting capabilities of the tools by deliberately creating a worse hull/bow form and compare the results with the original, “optimal” hull form (still to be done).

In our opinion this approach gives us a reasonable indication whether the selected open-source tools could produce useful results for our aim. More validating will however be required before we will actually use the results. Further more work is expected to be required to improve the workability and the usability of the system.

2. Available Open-source (or freely available) CFD tools

The working “process” with open source CFD is in general not different than with commercial CFD – packages, as you need pre-processing (meshing) – solving – post processing. Differences might be that the tools are less “integrated”. For each of the phases a number of open source tool exits, all with their own advantages and drawbacks, as will be shown in the following sections.
As there was no previous experience with creating CFD models and running the calculations, we started modeling and calculating based on the advices given in the "Best practice Guidelines for marine applications in Computational Fluid Dynamics" WS Atkins (2002). Though comparatively old, considering the wide range of developments during the last couple of years, most of the recommendations are still valid.

2.1. Pre-processing

We tried a number of pre-processing tools, but seem to get the best results with the combination of the following tools:

- Salome: to convert the ship geometry from an IGES-file to an STL-file.
- BlockMesh: to generate the “background” mesh/domain;
- SnappyHexMesh to combine the background mesh and the STL-file of the geometry

SnappyHexMesh and BlockMesh are part of OpenFoam, an open source CFD package. Salome is a generic platform, which besides pre-processing can also be used as a post-processing tool.
The first problem regarding meshing we encountered while trying to import a hull form geometry was that the Conoship’s CAD system can only export to 3D DXF and unfortunately we were not able to find any free meshing tools that can import 3D DXF. Further we could not find any free programs that could easily convert 3D DXF to IGES. However as an update of our CAD system is expected in the near future, which will be able to export to IGES we didn’t put a lot of effort in finding appropriate conversion tools and continued with an IGES of one of our hull designs created by a third party.

With some experience the meshing process can be done with Salome and the OF meshing tools within one morning. This includes converting the hull from an iges file to a stl file, prepare it by closing all ‘holes’ and create a solid body instead of an half body model, that can be introduced to the background mesh. Since the processes that are taking place in blockMesh and snappyHexMesh, are kind of standardized they can be executed by a batch-script which only needs ship parameters as an input to calculate the recommended domain size.

![Image: Mesh with bad quality, “hole” in the bow area](image)

The usual mesh size was limited to 2M cells due to the hardware capacities. Using more cells the RAM will be exceeded and the calculation crashes. With a ship with a Lpp of 120 m, a width of 15.8 m and a draft of 6 m, we used a domain of 440 x 121.6 x 57 m (L x B x H).

A detailed mesh study has not been done yet, but effects have been observed when using a too coarse mesh, the accuracy and uncertainties increase. The mesh size of 2M cells is truly not sufficient to get exact results that display detailed viscous effects however, based on discussions with experts and a comparison of our results with the results of the sub-contracted calculations for the same hull, we expect that it is sufficient to compare the relative performance of two hull forms.

As most of the open source CFD tools do not have a lot of fixed parameter settings, like some of the specialized programs for dedicated ship design calculations have (Shipflow, RAPID, etc.), a large number of parameter settings need to be defined during pre-processing. For example: boundary conditions, physical settings, discretization schemes solution algorithms and so on. Finding the right settings took much effort and we are still not convinced that we have reached our goal. We expect that when we figure out the correct settings they can be reused for many of our cases, as most of the designs are rather similar with respect to main dimensions, sailing area (water depth), etc.

### 2.2. Solving

Many different codes are available for “solving”, see for example the list published on the CFD-online forum, *CFD-Online.com* (2011). The trouble with most of the free codes was that they are usually
purpose made for one specific, in most of the cases academic problem. Among the codes on this list
only the Gerris-solver and OpenFOAM (OF) were further investigated, as they looked the most
promising considering our current (and near future) aims. Further the potential code Flotilla was
considered (Boatdesign.net, 2010) but as this code was basically developed for slender hull forms,
like a Wigley shape, and we would mainly like to evaluate general cargo vessels (which are rather full
compared to a Wigley hull), we didn’t further take this solver into account.

We choose Open Foam (OF) because from our point of view this was the most sophisticated program,
even though it quickly turned out that no free surface potential flow solver was available in OF (which
would be the most suited solver for our initial aim). The main reasons to nevertheless continue with
OF were:

- OF has a large and active user community providing the possibility for “free” support;
- A number of ship design cases published in literature and on the internet looked promising;
- OF is capable to do the RANSE calculations we would like to do in the near future;
- The project orientated set up of OF gives us the opportunity to add other features to the solver
  which would provide the possibility to add/change an existing solver to do free surface
  potential theory calculations (this development hasn’t been further pursued in this project);
- The Hochschule Bremen students had some experience with OF from university courses;

OF is published under the GNU General Public License. I.e. you are free to use the code without
license fees and are allowed to manipulate the source code which is also distributed. The only
obligation that comes along with this license is that once a part of code has been used for another
project, the resulting code has to be distributed for free under the GNU General Public license as well.
OF has a large user base across most areas of engineering and science, from both commercial and
academic organizations. OF has an extensive range of features to solve anything from complex fluid
flows involving chemical reactions, turbulence and heat transfer, to solid dynamics and electro-
magnetics.” Openfoam (2011). We have used version 1.7.1 and 1.7.x, as supplied by OpenCFD Ltd.

Since OF is a tool-box for any kind of CFD-calculations an appropriate solver needs to be selected
before any calculations are made. The range of available solvers in OF reaches from simple potential
solvers to single phase RANS solvers and from combustion solvers to multiphase RANS solvers for
both compressible and incompressible problems. In our case we choose the interFoam solver for two
incompressible fluids with a VOF approach. With this solver we were able to capture the free surface
elevation and consequently the wave pattern which we require for the relative performance
evaluation. As mentioned before, the number of parameters which can be adjusted in OF is large. The
main problem was (or is) to find the correct settings for all the parameters required by the interFoam
solver in order to produce useful results. Note that not for all free solvers the number of parameters to
set is as large as in OF. For example with Flotilla solver only a small number of options can be
changed. With OF there are 3 folder, each containing a number of files that together define all settings
for a case, for an overview see Table I.

<table>
<thead>
<tr>
<th>Folder</th>
<th>Folder “0”</th>
<th>Folder “system”</th>
<th>Folder “constant”</th>
</tr>
</thead>
<tbody>
<tr>
<td>Used for:</td>
<td>Definition of the</td>
<td>Controlling the</td>
<td>defining all constants for calculations; Sub-folders: poly-mesh and trisurface for geometry</td>
</tr>
<tr>
<td></td>
<td>initial settings</td>
<td>calculations</td>
<td></td>
</tr>
<tr>
<td>Files, a.o.:</td>
<td>alphal</td>
<td>controlDict</td>
<td>g</td>
</tr>
<tr>
<td></td>
<td>k</td>
<td>snappyHexMeshDict</td>
<td>transportProperties</td>
</tr>
<tr>
<td></td>
<td>nut</td>
<td>calcMassDict</td>
<td>RASPProperties</td>
</tr>
<tr>
<td></td>
<td>U</td>
<td>setFieldsDict</td>
<td>turbulenceProperties</td>
</tr>
<tr>
<td></td>
<td>p_rgh</td>
<td>fvSolution</td>
<td>pointLevel</td>
</tr>
<tr>
<td></td>
<td>nutTilde</td>
<td>fvSchemes</td>
<td>cellLevel</td>
</tr>
<tr>
<td></td>
<td>epsilon</td>
<td>decomposeParDict</td>
<td></td>
</tr>
</tbody>
</table>
Parameters defined for each file:

<table>
<thead>
<tr>
<th>Inlet</th>
<th>Outlet</th>
<th>Symmetry</th>
<th>Atmosphere</th>
<th>Solid/Hull</th>
</tr>
</thead>
</table>

We carried out the initial testing with the “Wigley case” Paterson (2008) and later on used one of Conoship’s hull designs for which CFD calculations were made by a specialist. Trying to find the right settings required much effort and is an ongoing and iterative process. Doing so we encountered many problems, such as:

- Reflecting waves from (supposedly) transmissive boundaries;
- Inexplicable pressure distributions;
- High waves at the front end of the domain;
- Water seeming to leave the domain.

Also in calculations with an “empty” domain, i.e. a domain without a ship geometry these problems occurred, pointing out that most of the problems were caused by (a combination) of faulty boundary condition settings. See for example Fig. 5, showing an “empty” domain with waves which are not supposed to be there. To find the right settings, we mainly changed the settings in the “k”, “epsilon”, “nutTilda”, “U”, “p_rgh” and “SetFieldsDict” file, focusing on the settings for the boundary conditions and wall/patch types. See Fig. 6 and Table II for an overview.

![Fig.5: Example of an empty domain wherein waves occurred](image1)

![Fig.6: Definitions of wall / patch type](image2)
Table II: Overview of boundary conditions (with k-\(\varepsilon\)-turbulence model)

<table>
<thead>
<tr>
<th>Inlet</th>
<th>Airinlet</th>
<th>Outlet</th>
<th>Symmetry-plane</th>
<th>Atmosphere</th>
<th>Hull</th>
</tr>
</thead>
<tbody>
<tr>
<td>U</td>
<td>fixedValue</td>
<td>fixedValue</td>
<td>zeroGradient</td>
<td>SymmetryPlane</td>
<td>zeroGradient</td>
</tr>
<tr>
<td>p_rgh</td>
<td>zeroGradient</td>
<td>zeroGradient</td>
<td>zeroGradient</td>
<td>SymmetryPlane</td>
<td>totalPressure</td>
</tr>
<tr>
<td>alpha</td>
<td>fixedValue</td>
<td>fixedValue</td>
<td>zeroGradient</td>
<td>SymmetryPlane</td>
<td>inletOutlet</td>
</tr>
<tr>
<td>k</td>
<td>fixedValue</td>
<td>fixedValue</td>
<td>inletOutlet</td>
<td>SymmetryPlane</td>
<td>inletOutlet</td>
</tr>
<tr>
<td>Epsilon</td>
<td>fixedValue</td>
<td>fixedValue</td>
<td>zeroGradient</td>
<td>SymmetryPlane</td>
<td>inletOutlet</td>
</tr>
<tr>
<td>nut</td>
<td>calculated</td>
<td>calculated</td>
<td>calculated</td>
<td>SymmetryPlane</td>
<td>calculated</td>
</tr>
<tr>
<td>nuTilda</td>
<td>fixedValue</td>
<td>fixedValue</td>
<td>zeroGradient</td>
<td>SymmetryPlane</td>
<td>inletOutlet</td>
</tr>
</tbody>
</table>

For the turbulence model, we tried both the two equation approach with the k-\(\varepsilon\)-turbulence model (good for far field problems) as the k-\(\omega\)-SST model (good modeling for both far field and boundary effects Bertram (2000)).

2.3 Post processing

In this project we only considered one post-processing tool, Paraview, as it suited our needs very well right from the start. It comes with OF, but is a general platform for data analysis and visualization. Through paraFoam, also included in OF, the results are converted into a format that can be read by Paraview.

Problems concerning the post processing basically came from bad documentation. For example it took us a while to figure out the differences between “elevation” and “wave height”, how to change the scale of the “elevation”, etc. Further be aware that the speed of the computer is significantly slowed down when the post-processing tool and the solver run in parallel.

3. Results

Currently we are able to produce a wave profile along the hull that shows a reasonably good match with the wave profile from the sub-contracted calculations (based on a potential theory code), as shown in Fig. 7. I.e. the height of the wave shows as a good match, as does the location of the peaks. Further the wave pattern around the vessel does no longer show any anomalies like reflecting waves, neither do waves occur in an empty domain.

Up till now, about 9 man-months were required to find the right software and set-up the system, to find the best settings and the evaluate the results. All the calculations were done at “full scale”, with about 1.0 M to 1.8 M cells.

Meshing goes fast, with calculation times to create the mesh in snappyHexMesh of about 0.5 to 1.0 h. Solving calculation times were much longer. With the current domain size, for 30 s time domain 32 – 36 h calculation are required, with some working on the machine in between (e.g. for post-processing, reporting, etc.). Calculation of 120 s (over a weekend) requires about 72 h. Both meshing and calculation are done on a PC with 2 GB of RAM and a dual core processor with 2 GHZ per processor.
3.1. Advantages / Disadvantages of OF

As the previous sections clearly showed, the problems we encountered were of different nature. However one of the common problems was the lack of sophisticated support. Even though OF has a large user community which provides a lot of help, accessing this support is not always easy. I.e. there is no single phone number for all of your software problems, like there is with commercial packages. Furthermore the people on the internet fora are no “full time held desk employees”, so it might take some time before someone answers your question. However when you do have time, the community can provide real good help, as many people are active on the CFD-Online forum (http://www.cfd-online.com/Forums/openfoam/) and as people from a range of industry participate a wide range of experience is available. Further all kind of national and international “user groups” are founded (like the Dutch Open Foam Group), of which a number have special interest groups that meet in “real-life” as well, providing the opportunity for a vivid knowledge exchange. Even though that, through the collected user experience, the learning curve can be steep, it is in no way comparable to buying a commercial package, with some instructional course and dedicated support. Thus when you are in a hurry to get a working system and produce results, Open Source CFD is not the best way to go.

Another disadvantage OF is the fact that “you have to do it”. I.e. there are no, or at least not much, pre-defined settings which are tested, evaluated and validated for ship design problems, like there is when you buy a commercial program like ShipFlow. This requires at least some understanding and knowledge regarding programming, hydrodynamics and mathematics. Once you have installed OF, a very complex process of adjusting and validating starts, before you can actually produce some useful
results and it can be a very steep way to go before you reach your goal. The number of parameters that can be adjusted in OF is very large, and unfortunately the manual gives a rather limited description of the working of most of the parameters. In most cases the community can provide the required answers, however for a number of the parameters also the community isn’t sure about the working or the required setting. But, having no pre-defined settings is also an advantage of OF. It is not a “black-box” and the freedom to make changes gives many opportunities, which is shown by the broad application of OF throughout the industry and the academic world. Further the object orientated way of building up of OF allows for an “easy” extension of the created system. In the ship design case, for example a seakeeping program can be added, as done by Paterson Paterson (2009), or a free-surface potential flow solver can be added to speed up the calculation required to define the overall performance of the ship’s hull.

Clearly the “out-of-pocket” costs for a completely open-source system are very low compared to purchasing a dedicated commercial CFD package, with annual license fees of over €25.000,-. However the required effort to get a reasonably working system is not to be underestimated. For a small company like Conoship, this would not have been feasible without the assistance of internship and graduation students. The amount of effort required is simply too large to be done besides the normal day-to-day design activities and freeing up a designer for such amount of time would have been just as costly as acquiring a commercial package. A great advantage of the open source CFD tools is, that when there is a working system, you are free to run on as many nodes as you like, which could considerable speed up the calculations, or provides the possibility to evaluate various alternatives concurrently. With many of the commercial packages the license is limited to running on a single node.

4. Conclusion & Outlook

An answer to the question whether open-source CFD tools are a feasible solution for a small ship design office, is “yes and no”. Although we are not there yet, we are convinced that it is possible with Open Foam to get useful results for the application we are after. To get there however requires such an effort that most of the smaller design offices won’t be able to pull that off. A good compromise could be to hire external support of Open Foam experts to (once) set-up the system and to define suitable settings for the required application, while the actual calculations are performed by the design company’s own designers thus avoiding the high annual licensing costs while acquiring a powerful CFD tool with a relatively small investment.

Conoship meanwhile continues with the investigations of the suitability of OF. Further work will amongst others be focused at:

- Evaluating the advantages and disadvantages of model scale calculations and compare the results to the full scale calculations;
- Further evaluate and define the required calculation time to come to a stable solution;
- Further evaluate the effect of mesh refinements to define optimal meshing for our application;
- Improve user-friendliness;
- Develop in in-house manual and guideline.

All in all we expect to have work for another 3 – 5 internships or graduation projects.

Acknowledgment

We wish to thank everybody on the CFD-online forum for their continuous support and assistance.
References

WS ATKINS (2002), *Best practice Guidelines for marine applications in Computational Fluid Dynamics*
https://pronet.wsatkins.co.uk/marnet/guidelines/guide.html (visited 11.03.2011)

BERTRAM, V.; COUSER, P. (2010), *Aspects of selecting the appropriate CAD & CFD software, 9th Int. Conf. on Computer and IT Applications in the Maritime Industries, Gubbio*


PATERSO, E. (2009), *Simulation of wakes, wave impact loads, and seakeeping using OpenFOAM*, 4th OpenFOAM Workshop, Montreal

http://www.openfoam.com (visited 11.03.2011)

http://www.cfd-online.com/Wiki/Codes (visited: 11.03.2011)

http://www.blender.org/ (visited: 11.03.2011)

http://brlcad.org/ (visited: 11.03.2011)

http://www.paraview.org/ (visited 11.03.2011)

http://www.boatdesign.net/forums/design-software/flotilla-demo-2-07-a-32276.html post 1 by Leo Lazauskas (visited 11.03.2011)

Numerical Hull Series for Calm Water and Sea-Keeping

Patrick Couser, Formation Design Systems Pty Ltd, Fremantle/Australia, patc@formsys.com
Stefan Harries, Friendship Systems GmbH, Potsdam/Germany, harries@friendship-systems.com
Fabian Tillig, SSPA Sweden AB, Göteborg/Sweden, Fabian.Tillig@sspa.se

Abstract

Naval architects draw inspiration from previous designs, literature reviews, statistical regression models and systematic series. In this paper, a complementary approach, using simulation-driven design, is presented: exploration of the multi-dimensional design-space using first-principles methods. The vessel is modelled parametrically with the free-variables that define the design-space. The design-space is then populated by systematic variation of these variables. The key benefit of the proposed method is that it allows the design team to quickly explore the design-space and build up a knowledgebase ahead of an anticipated project. This then allows quick interrogation of the numerical model series to substantiate design decisions during the bidding and tendering process.

Nomenclature

<table>
<thead>
<tr>
<th>Abbreviation</th>
<th>Definition</th>
</tr>
</thead>
<tbody>
<tr>
<td>$B_{WL}$</td>
<td>Beam on waterline</td>
</tr>
<tr>
<td>$CB$</td>
<td>Centre of Buoyancy</td>
</tr>
<tr>
<td>$CG$</td>
<td>Centre of Gravity</td>
</tr>
<tr>
<td>$DWL$</td>
<td>Design Waterline</td>
</tr>
<tr>
<td>$GMt$</td>
<td>Transverse metacentre above $CG$</td>
</tr>
<tr>
<td>$GZ$</td>
<td>Hydrostatic righting lever</td>
</tr>
<tr>
<td>$LCB$</td>
<td>Longitudinal Centre of Buoyancy</td>
</tr>
<tr>
<td>$LCG$</td>
<td>Longitudinal Centre of Gravity</td>
</tr>
<tr>
<td>$L_{PP}$</td>
<td>Length between perpendiculars</td>
</tr>
<tr>
<td>$VCB$</td>
<td>Vertical Centre of Buoyancy</td>
</tr>
<tr>
<td>$VCG$</td>
<td>Vertical Centre of Gravity</td>
</tr>
<tr>
<td>$\mathbb{R}^n$</td>
<td>n-dimensional (design) space</td>
</tr>
</tbody>
</table>

Abbreviations

<table>
<thead>
<tr>
<th>Abbreviation</th>
<th>Definition</th>
</tr>
</thead>
<tbody>
<tr>
<td>CFD</td>
<td>Computational Fluid Dynamics</td>
</tr>
<tr>
<td>COM</td>
<td>Component Object Model</td>
</tr>
<tr>
<td>CPU</td>
<td>Central Processing Unit</td>
</tr>
<tr>
<td>FLOPS</td>
<td>Floating-point Operations per Second</td>
</tr>
<tr>
<td>FFW</td>
<td>FRIENDSHIP-Framework (Software)</td>
</tr>
<tr>
<td>GPU</td>
<td>Graphics Processing Unit</td>
</tr>
<tr>
<td>HM</td>
<td>Hydromax (Software)</td>
</tr>
<tr>
<td>MSI</td>
<td>Motion Sickness Incidence</td>
</tr>
<tr>
<td>RAO</td>
<td>Response Amplitude Operator</td>
</tr>
<tr>
<td>SK</td>
<td>Seakeeper (Software)</td>
</tr>
</tbody>
</table>

1. Introduction

The aim of this paper is to demonstrate, by means of an example application to a mega-yacht, how numerical simulation can be used to explore the design-space early in the concept design stage of a project and how this information may be used to gain deeper insight into the design compromises which will have to be made.

Table I shows the principal particulars of the proposed vessel; these would typically be given by the client: “Design me a mega-yacht that’s a bit faster, a bit bigger and a bit more luxurious than the one I bought last year!” As can be seen, the design requirements are quite vague, so it is of utmost importance to gain an understanding of the design-space in which the solution will lie (or even to ascertain the feasibility of the proposal).

1.1. Why?

Information is power! Prior knowledge of the relevant design-space for a ship-design project enables the design team to achieve a sensible compromise that meets the customer’s requirements. This knowledge can be gained in several ways. For example, an existing vessel may serve as a basis design from which a new, improved vessel that better fulfils the customer’s requirements can be derived. Or, if there is little prior knowledge or the project requires a completely novel vessel design, then it is important for the designer to gain an understanding of the design-space by some other means. To summarise, the proposed approach might be used to:
• Gain an insight of the design-space early in the project;
• Enable rapid prototyping of ideas for novel design solutions;
• Provide data for decision support for possible design changes required to achieve desired performance; and
• Anticipate consequences of requested design changes.

<table>
<thead>
<tr>
<th>Table I: Principal particulars of the proposed mega-yacht</th>
<th>minimum</th>
<th>maximum</th>
</tr>
</thead>
<tbody>
<tr>
<td>Length between perpendiculars, $L_{pp}$ [m]</td>
<td>68.00</td>
<td>72.00</td>
</tr>
<tr>
<td>Beam on DWL [m]</td>
<td>14.00</td>
<td>14.25</td>
</tr>
<tr>
<td>Design Waterline, DWL [m]</td>
<td>3.9</td>
<td></td>
</tr>
<tr>
<td>Displacement in seawater at DWL [tonnes]</td>
<td>approximately 2200</td>
<td></td>
</tr>
<tr>
<td>Cruise speed [kts]</td>
<td>16.0</td>
<td></td>
</tr>
<tr>
<td>Maximum speed [kts]</td>
<td>20.0</td>
<td></td>
</tr>
</tbody>
</table>

1.2. How?

The example presented serves to illustrate the concepts and methodology. However, there is no reason why different design aspects could not be examined or different numerical tools used. The key is being able to automatically vary the proposed vessel in a manner so as to produce viable variants and then be able to predict the variants’ performance characteristics pertinent to the design requirements.

The design-space investigation thus comprises three main tasks:
1. Definition of a suitable parametric model which can be used to generate feasible design variants from a small number of key parameters.
2. Numerical analysis of the vessel using simulation tools which can provide an assessment of the vessel performance characteristics of interest (in this case, hydrostatics, resistance and sea-keeping). These tools need to be selected so that they can provide sufficiently reliable data within available time and computational resource constraints.
3. Automation of vessel design variation, analysis, results gathering and post-processing tasks.

The FRIENDSHIP-Framework (FFW – Friendship Systems, 2009) is used to firstly define the hull geometry in a parametric manner which can then be systematically varied and secondly to systematically vary the design, control the analyses and collate the results for all the design variants.

1.3. What is Important?

What is of interest and importance to the designer will depend on the individual project being undertaken. In this example, static stability as well as resistance and also passenger comfort when the vessel is under the influence of waves are considered.

The vessel’s calm water resistance was estimated using SHIPFLOW (Flowtech 2004, 2009), whilst sea-keeping characteristics and hydrostatic stability were predicted using Seakeeper (SK) and Hydromax (HM – Formation Design Systems, 2011).

1.4. Computer Hardware

It is interesting to look at the increase in computer performance over time; this is shown in Fig. 1 for the last 30 years (SUPERCOMPUTER 2011; Thibault et al. 2009; Koomey et al. 2009). There continues to be exponential growth in not only the performance of supercomputers but also that of personal micro-computers. What is also interesting is the application of GPUs rather than CPUs to solving CFD flows (Thibault et al. 2009). GPUs can be optimised for floating-point calculations and
matrix inversion much more than CPUs (which are required to perform a much broader range of operations). The rate of increase in performance of GPUs is greater than that of both CPUs and Supercomputers.

The rapid development of computer hardware and the advent of computer clusters and clouds (e.g. Amazon Elastic Compute Cloud –EC2) and other distributed systems now mean that the hardware resources necessary for the type of numerical investigations described in this paper are now accessible to even the smallest design teams.

2. Methodology of the Investigation

In this section, we shall look in some detail at the numerical method used for the design-space investigation. The key concept to take from this paper is the methodology; different analysis software can be substituted and different performance measures will be appropriate for different projects.

2.1. Parametric Modelling

The general hull-form chosen for the example mega-yacht was a classical twin-screw design with bulbous bow and skeg. Appendages were not included at this initial phase of the design. The bulb was modelled in some detail, since it had a significant impact on the hull resistance. The bulb was blended into the main hull over a region of transition aft of the forward perpendicular. The main hull itself was split into fore- and aft-body regions joined at the section with maximum cross-sectional area. A full 3D model of this geometry was realised in the FFW.

The model was parameterised so the geometry could be manipulated by a small number of key features which the designer would wish to vary. These parameters are the free-variables of the n-dimensional design-space to be investigated and were used to generate design variants within that space. The parameters (or free-variables), with their range of variation are given in Table II and the primary curves describing the model are shown in Fig. 2. The body plan, plan, profile and perspective views of a representative instance of the parametric model are shown in Fig. 3. Full details of the parametric model are described in Harries (2010).
Table II: The free variables that are used to define the parametric model

<table>
<thead>
<tr>
<th>Parameter</th>
<th>minimum</th>
<th>maximum</th>
</tr>
</thead>
<tbody>
<tr>
<td>Length between perpendiculars, ( L_{pp} ) [m]</td>
<td>68.00</td>
<td>72.00</td>
</tr>
<tr>
<td>Beam on ( DWL ) [m]</td>
<td>14.00</td>
<td>14.25</td>
</tr>
<tr>
<td>Midship area coefficient</td>
<td>0.82</td>
<td>0.89</td>
</tr>
<tr>
<td>Prismatic coefficient of fore part of hull</td>
<td>0.60</td>
<td>0.63</td>
</tr>
<tr>
<td>( DWL ) half angle of entrance [deg]</td>
<td>14.0</td>
<td>18.0</td>
</tr>
<tr>
<td>( DWL ) fullness coefficient</td>
<td>0.58</td>
<td>0.62</td>
</tr>
<tr>
<td>Bulb area to midship area ratio</td>
<td>0.092</td>
<td>0.098</td>
</tr>
<tr>
<td>Bulb fullness coefficient</td>
<td>0.75</td>
<td>0.85</td>
</tr>
<tr>
<td>Longitudinal position of section with max. cross-sectional area [% ( L_{pp} )]</td>
<td>44.0</td>
<td>48.0</td>
</tr>
</tbody>
</table>

Fig. 2: Parametric model of round bilge mega-yacht

Fig. 3: Example of typical bare hull with bulbous bow generated from the fully parametric model
2.2. Performance Prediction

The numerical tools used to calculate the vessel performance are described below. The tools presented cover three of the main areas of interest during initial design: resistance, sea-keeping and static stability. However it would be entirely feasible to include other tools to compute other performance parameters, for example production cost, manoeuvring, etc. The scope of the performance parameters to be considered depends on the time available to complete the study as well as the tools available and the level of detail of the ship model required to produce meaningful results.

2.2.1. Flow Simulation and Resistance Prediction

When predicting calm-water resistance, there is generally a trade-off between accuracy and the computational effort required. Since only the bare hull was modelled, it was considered appropriate to employ potential flow theory to solve the non-linear wave resistance problem with free sinkage and trim combined with a thin boundary layer theory calculation for the frictional resistance, further details are given in Harries (2010). When fine-tuning appendages, such as brackets, later in the design, a RANSE calculation should be undertaken to accurately capture the viscous phenomena, for example Brenner (2008).

The flow simulations were computed on a standard dual core notebook and took about four to five minutes per variant and speed. With a CFD license for both cores, around 200 designs could be computed in one overnight job. A typical panel arrangement and results are shown in Fig. 4.

![Fig. 4: Typical panel arrangement of free surface and hull with wave-wake height contours and hull streamlines at \( F_N = 0.393 \)](image)

2.2.2. Motions in Waves and Comfort Measures

The vessel motions due to waves were predicted using Seakeeper – a linear strip theory method in the vein of Salvesen et al. (1970). Two scenarios were considered (details are given in Table III):

1. Vessel at anchor or in a marina in a very slight sea-state – the so-called “Party” condition. (Note that mooring forces were not considered.)
2. Vessel underway at a cruising speed of 16kts in a higher sea-state, as might be encountered when traveling between two such “Party” locations – the “Cruise” condition.

The motion sickness incidence (MSI) after two hours exposure was computed at different longitudinal positions along the length of the vessel (Fig. 5). That is the percentage of people who can be expected to vomit after having been subjected to the motions for a period of two hours, as calculated by the method proposed by O’Hanlon and McCauley (1974) and McCauley et al. (1976). The performance
measure extracted from the analysis was simply the minimum MSI along the length of the vessel for each of the two scenarios considered; assuming that the vessel layout could be adjusted so that MSI-critical systems (e.g. the bar) could be sited accordingly.

Table III: Two scenarios considered for the sea-keeping calculations

<table>
<thead>
<tr>
<th></th>
<th>“Party”</th>
<th>“Cruise”</th>
</tr>
</thead>
<tbody>
<tr>
<td>Vessel speed [kts]</td>
<td>0.0</td>
<td>16.0</td>
</tr>
<tr>
<td>Characteristic wave height [m]</td>
<td>0.5</td>
<td>2.0</td>
</tr>
<tr>
<td>Modal period [s]</td>
<td>2.0</td>
<td>7.1</td>
</tr>
<tr>
<td>Wave heading</td>
<td>Head seas</td>
<td></td>
</tr>
<tr>
<td>Wave spectrum type</td>
<td>1-Parameter Bretschneider</td>
<td></td>
</tr>
</tbody>
</table>

Fig. 5: Typical MSI distribution over the length of the vessel

The sea-keeping model used 41 equally spaced sections through the hull. Conformal mapping was used to model the sections and compute the sectional added mass and damping in heave (five mapping terms were used to give a good fit to the hull sections). The vessel heave and pitch response amplitude operators (RAOs) was then calculated at 200 frequencies and these were used to calculate the MSI. The calculations, for 200 variants, were computed on an average desktop computer using SK, again in an overnight job controlled by the FFW.

2.2.3. Hydrostatic Stability Criteria

Virtually all vessels must comply with hydrostatic stability criteria specified by class. A small subset of intact-vessel stability criteria, which are typically applied to this class of vessel, were selected from the *Large Commercial Yacht Code* (*Maritime and Coastguard Agency 2007*) intact stability standards for monohull vessels, section 11.2.1.1. These criteria are summarized in Table IV.

Table IV: Stability criteria considered

<table>
<thead>
<tr>
<th>Section</th>
<th>Description</th>
<th>Required value</th>
</tr>
</thead>
<tbody>
<tr>
<td>11.2.1.1.1a</td>
<td>Area under $GZ$ curve from 0 to 30 deg. heel shall not be less than</td>
<td>0.055 m.rad</td>
</tr>
<tr>
<td>11.2.1.1.1b</td>
<td>Area under $GZ$ curve from 0 to 40 deg. heel shall not be less than</td>
<td>0.090 m.rad</td>
</tr>
<tr>
<td>11.2.1.1.2</td>
<td>Area under $GZ$ curve 30 to 40 deg. heel shall not be less than</td>
<td>0.030 m.rad</td>
</tr>
<tr>
<td>11.2.1.1.3</td>
<td>Maximum $GZ$ at 30 deg. or greater heel shall not be less than</td>
<td>0.2 m</td>
</tr>
<tr>
<td>11.2.1.1.4</td>
<td>Angle at which maximum $GZ$ occurs shall not be less than</td>
<td>25 deg.</td>
</tr>
<tr>
<td>11.2.1.1.5</td>
<td>Initial metacentric height ($GM_t$) shall not be less than</td>
<td>0.15 m</td>
</tr>
</tbody>
</table>

In order to obtain a meaningful performance measure of stability, the maximum vertical centre of gravity ($VCG$) at which all criteria were just passed was calculated for a range of displacements. A
typical curve of maximum allowable $VCG$ against displacement, for three representative design variants, is shown in Fig. 6. The measure of performance used was the area under the maximum allowable $VCG$ curve integrated over the displacement range of 1800t to 2600t. This measure was chosen because early in the design process, neither the $VCG$ nor the displacement would be known with certainty; the measure gives some indication of the scope of $VCG$ change that can be accommodated whilst still passing the criteria.

Damage stability has not been considered at this stage because this would depend on the compartmentation layout which would not be available early in the initial design. Once a design variant has been selected for detailed design development, the internal layout and compartmentation would have to be chosen so that damage stability requirements were met.

The analysis was performed in $HM$ using a range of heel angles at each displacement to calculate the $GZ$ curve for a given $VCG$. The vessel was free-to-trim ensuring a longitudinal balance of $CG$ and $CB$ (the $LCG$ being derived from the $LCB$ of the upright vessel). The $VCG$ was then systematically varied to determine the maximum value of $VCG$ at which all the stability criteria were still passed. Managed by the $FFW$, the calculations, for 200 variants, were computed in a matter of several hours.

![Fig. 6: Typical MSI distribution along the length of the vessel](image)

2.3. Software Integration

The $FFW$ and the simulation software are developed by different software vendors. However, in order to automate the task of generating design variants and analyzing their performance, it is essential that the software systems are able to communicate. Under Microsoft Windows there exists a paradigm for inter-process communication. This is known as the Component Object Model (COM). For full details of COM, the interested reader is referred to Box (1998). COM allows access to suitably COM-enabled applications via a common interface from a variety of programming languages: C#, VBA, etc. and also the $FFW$'s own macro language.

Suitable macros were developed in the $FFW$ to export the hull geometry and then import this geometry and run the analyses in $HM$ and $SK$. The results of the analyses were then read back into the $FFW$ for post-processing to calculate the final performance measures for each variant. Fig. 7 shows a screenshot of the $FFW$, $HM$ and $SK$ in action.
2.4. Design of Experiments

The design-space was investigated using a “Design of Experiments” approach to populate the domain with variants. The principal particulars of the vessel are given in Table I and the nine free-variable which were used to define the variants are given in Table II. These nine free-variable thus establish a nine-dimensional space $\mathbb{R}^9$. A Sobol algorithm, Press et al (2007) was used to give a quasi-random, yet uniform sampling of these variables over the desired range (see Table II). The performance was calculated for 200 variants. A typical distribution of variants (for one free-variable) is shown in Fig. 8; as expected, the Sobol algorithm provides a uniform, quasi-random sampling over the design-space. The design of experiments approach covers the design-space more economically than a regular grid approach – a regular grid of just two parameter variations in 9 dimensions would require $512$ ($2^9$) variants.

2.5. Response surfaces

The design-space exploration generates a large quantity of data and represents a not insignificant amount of computational effort (especially if sophisticated numerical simulation tools have been
It is useful then, to reuse this data, potentially for automated optimisation or other similar applications. There are several ways in which this data can be captured so as to be able to determine the vessel performance measures for a set of specified values of the free-variables. These include: statistical regression; artificial neural networks (e.g. Couser 2004) and response surfaces. All of these methods effectively allow interpolation of the performance measures of the design given a set of values of the design parameters (free-variables) without having to redo the numerical simulation, thus saving a lot of computational effort.

Following the work of Harries (2010) a response surface, meta-model method has been used. The n-dimensional (where n is the number of free-variables) response surfaces for the performance measures are fitted using a Kriging approach, see Tillig (2010). Once the response surface has been generated, interpolation is more or less instantaneous (compared with a CFD or sea-keeping calculation which might take a few minutes to several hours to perform). Continuous iso-parametric curves and surfaces can then be generated from the response surface making it easier for the designer to visualise the design space: the designer is able to see the effect of continuously varying one or two free-variables rather than seeing discrete results for variants where all the free-variables have been modified (which is the raw output from the design of experiments investigation of the design-space).

3. Results

This section presents some results for the mega-yacht example. One should not forget that these findings are only meaningful in the context of the chosen parametric model (the established design-space) and that they rely on the validity of the simulations. Even though these simulations are built on first principles, there are notable simplifications, for instance the wave resistance and sea-keeping analyses, as used in this example, ignore viscosity.

3.2. Correlations

Some samples of the raw results from the Sobol investigation of the design-space are presented by means of correlation plots (as shown in Figs. 9 to 14). These correlation plots can highlight general trends in the data but it should be noted that the points represent discrete variants where all of the free-variables have changed; thus these diagrams do not accurately represent the continuous variation of a single variable. The band-width of the scatter of points about the mean line gives an appreciation of the difference that can be achieved due to variation of all the other free-variables. It should be noted that even when there is reasonably strong correlation between performance and a free-variable, there is often a significant range of performance (which thus depends on the other free-variables). For example, in Fig. 11, at a length of 70m the Cruise MSI can vary between 4% and 5%. This also implies that there is always room for improvement even though one (or several) free-variables need to be fixed at a certain point in the design process. The range of performance can be taken as an initial indication of how much potential for optimisation is available.

3.2.1. Principal Hull Geometry

Fig. 9 presents the vessel displacement against the length between perpendiculars. A general trend towards higher displacement for longer vessels can be seen. Nevertheless, as discussed above, there are instances of vessels with higher and lower displacements (for a fixed $L_{PP}$) that depend on the values of the remaining free-variables.

3.2.2. Calm Water Resistance

Fig. 10 shows the correlation between vessel length and predicted wave resistance coefficient. As might be expected the longer the vessel, the lower the resistance. The interested reader is referred to Harries (2010) for further details and results of the resistance calculations performed.
Fig. 9: Typical correlation of a performance measure with a free variable (Displacement and Length)

Fig. 10: Strong correlation between Wave resistance coefficient and $L_{PP}$

Fig. 11: Strong correlation between MSI and $L_{PP}$
Fig. 12: Very weak correlation between MSI and Beam

Fig. 13: Strong correlation between Stability performance measure and $L_{pp}$

Fig. 14: Un-correlated relationship between Stability performance measure and Bulb Fullness coef.
3.2.3. Sea-Keeping

As might be expected, the MSI shows a reasonably strong inverse correlation to the vessel length: as the vessel length increases, the motion sickness incidence decreases, Fig. 11. Sometimes it may be found that there are surprising correlations (or lack thereof); for example Fig. 12 shows that the correlation between MSI and beam is not very strong, contrary to what might be expected.

3.2.4. Hydrostatic Stability Criteria

A strong inverse correlation between vessel length and stability was found – Fig. 13. This is probably due to the fact that the displacement range for the stability calculations was fixed irrespective of vessel length. Shorter vessels would be broader and/or deeper in the water generally resulting in greater intact stability (up to the angles of heel investigated). Other parameters showed little or no influence on stability (Fig. 14) indicating that they can be varied to improve other performance measures without penalising the stability performance.

3.3. Response Surfaces

Once the n-dimensional response surfaces have been fitted to the discrete data obtained from the design-space exploration, continuous iso-parametric curves and surfaces can be generated for continuous variation of only one or two free-variables (the others remaining constant). In Figs. 15 to 17, all but two free-variables are kept constant resulting in iso-surfaces through the design space. In each diagram the range of each free-variable has been normalised to 1.0.

In most cases the response surfaces follow what might be expected: Fig. 15 shows that the delivered power requirement is reduced for longer and generally narrower vessels; and Fig. 16 shows that MSI is reduced for longer vessels, with the optimum beam being about the middle of the range. However in the case of the stability performance measure response surface, Fig. 17, the effects of length and beam are more complex. It should be noted that since the entire design-space exploration is not covered by the variants tested, care should be taken to ensure that the response surface is used for interpolation, and not extrapolation. The sharply raised corners in Fig. 17 are due to extrapolation with insufficient variants to adequately describe the response surface in these regions.

Fig. 15: Response surface for Power vs. Length and Beam
Fig. 16: Response surface for Sea-keeping vs. Length and Beam

Fig. 17: Response surface for Stability vs. Length and Beam

4. Taking Things Further / Practical Application

In this paper we have presented an example using a mega-yacht initial design project. Relatively simple numerical simulation tools have been used to investigate three aspects of the design process: calm-water resistance (using potential flow and boundary layer theory), sea-keeping (using strip-theory) and static stability. However, there is no reason why the same methodology cannot be applied to different problems using different simulation tools.
The FRIENDSHIP-Framework is very useful in that it facilitates a parametric model of the hull geometry and allows this geometry to be systematically varied. It then manages sending the geometry to and retrieving the results from the external simulation tools. The COM interface provides a relatively simple mechanism for inter-process communication on the Microsoft Windows platform. (The coupling between the FFW and the analysis software, SK and HM, was achieved using COM.)

Although not presented in the current work, using the resulting response surfaces to drive an optimisation search is entirely possible and would be the logical next step of the design process (see Harries (2010) for an example).

5. Conclusions

One of the most challenging tasks for the ship designer is to gain an insight into the non-linear relationships between competing objectives, constraints, free- and dependent-variables so as to be able to obtain a suitable final design that meets the customer’s requirements. The methodology described in this paper shows how first-principles simulations, coupled with a parametric model of the vessel can facilitate rapid exploration of the design-space. The methodology can be summarised as follows:

1. Creation of a suitable parametric model of the vessel. The parameters chosen to be free-variables entirely define the vessel and span the design-space of interest. They can be regarded as the free-variables of an optimisation problem.
2. Performance measures and constraints such as hydrostatics and hydrodynamic performance are identified and determined by means of numerical simulations based on the vessel obtained from the parametric model.
3. The design-space is then systematically and automatically explored using formal methods.
4. The results of the design-space exploration are captured by response surfaces that allow for very rapid interpolation of the performance measures for any set of values of the free-variables.
5. Once the design-space is known and understood, the data can be used to answer “what if?” type questions as well enabling optimisation searches to be performed quickly.

The key things to take from this paper is the methodology. The details of the specific parametric model and analysis tools used are of less importance because they can (and should) be adapted and tailored to the specific needs of the individual project. However, what this paper aims to show is how an in-depth knowledge of the design-space in which one finds oneself can be gained by more formal and extended use of numerical simulation. Of course, as computational power continues to increase and the accuracy of numerical simulation techniques continues to improve, it will be appropriate to change the hardware and software used to perform the design-space exploration. It is believed by the authors that the approach described in this paper will aid naval architects during their design tasks by providing familiarity with novel design ideas more quickly and allowing them to make appropriate design modifications to match evolving client requirements more easily.

References


BRENNER, M. (2008), Integration of CAD and CFD for the hydrodynamic design of appendages in viscous flow, Diploma thesis, TU Berlin


HARRIES, S. (2010), Investigating multi-dimensional design spaces using first principle methods, 7th Int. Conf. on High-Performance Marine Vehicles (HIPER), Melbourne.

FLOWTECH Int. AB (2004), *XPAN / XBOUND Theory*, Gothenburg


The ability to share data during ship design, production and management is very limited, with undesirable consequences. Data that could be shared with benefit can be loosely categorized as CAD and non-CAD, as a function of source, nature, but also practical use and possible exploitation parameters during the various stages of the ship's life span. Existing off-the-shelf tools and technology can be used today to share, manage and exploit such data. Examples of the above, including software recently developed for the purpose will be illustrated, and a practical example presented.

1. Introduction

The effective use of orally transmitted data has been the foundation of mankind's evolution, Danese (2010), but for practical and effective use in modern industry data needs to be re-usable exactly. At least three corollaries follow:

- data must be stored persistently.
- data must be retrievable.

Therefore

- data must be stored electronically.

Therefore, the creation of an electronic version of data is the first step. Electronic storage being a rather non-standard environment by definition, data must be formatted intelligibly, and formatting must be documented. While being obvious, this is the case far less often than desirable. The vast number of electronic data organizer tools developed in recent times is a loud symptom of the importance of data management, and the absence of one, or even a few, standard-setting product(s), an even more evident indication data-sharing proficiency has not been achieved.

The reason for data not being shared effectively is not fundamentally technical, plenty of tools exist today to help find, retrieve and process information. On the other hand, assuming that data has been made available in the first place, which is more and more the case, how it is made available will determine whether it can be found or not, and then exploited, or not.

With regards to the CAD environment, and more specifically the ship related industry, a majority of data does exist in electronic form, is readily available in industry standard file formats, but is not exploitable for a number of reasons, two immediately identifiable culprits being proprietary data structure, and absence of contextualization.

2. Common place data management tools

Among the remarkable efforts undertaken so far, some are representative of the main avenues of development: MS-SharePoint, the web search engine (Google, Yahoo, etc.) and social networking systems (FaceBook, etc.). However:

- MS-SharePoint manages documents of various nature, but not necessarily the data contained in them.
- Search engines like Google, Yahoo, etc. use keywords, which can be loosely compared to a relational database structure, and successive searches, but are somewhat limited to formatted text and, e.g., do not read inside databases, files, etc. and cannot interpret pictures and graphs.
- Social networking systems allow multiple sources to contribute to a given data set ("friends" are allowed to post to an individual's page), but again data is exposed through text.
Perhaps cynically, it could be argued that the difference between the above and the eternal (and ubiquitous) card filing systems is "limited" to the global reach of the formers, and the filing cabinet bound nature of the latter.

3. CAD and non-CAD data

In a first, perhaps somewhat simplified approach, two generic data groups will be identified, CAD and non-CAD. CAD data is loosely intended to include all data that can and is explicitly generated and represented by "graphical" means, such as a line, a solid, etc. This data is generally created by CAD programs, or via graphical libraries (ex. Hoops, etc.). non-CAD data includes all data that is difficult to or cannot be represented by graphical means, such as relations between objects, number of objects, results of calculations, dates, etc. Nonetheless, this data can be and often is, an integral part of the message to be delivered in conjunction with or even by CAD data.

![Fig.1: CAD data](image1)
![Fig.2: non-CAD data](image2)

The word "graphical" undeniably lends itself to argument, so let us consider a debatable example: the concept of thickness. Thickness can be represented graphically, generally by hatching a section or of two parallel segments, but its value is generally expressed with a number, written as such.

4. Data structure, contextualization and inconsistency

Further to previous work on the nature of data and more specifically ship-related data, Danese (2010), a high-level study of data structure and contextualization was undertaken, taking into account the implications of the intrinsic inconsistency of data.

4.1. Data structure

"Data structure" refers to how the data is stored and, hence, made available. Data structure will depend on the nature of the data (somewhat universal), and on how the data will be interpreted in order to process and present it.

4.1.1. Data structure of CAD data

Let us consider different ways to describe a line segment and a cylinder, while in each case using the same data set (one data set for the line and one for the cylinder).

These examples apply equally to lines and cylinders created via the graphical interface of a CAD program, via the programming of a graphical library, or via the use of neutral file formats, such as Autodesk's DXF. So, while the information needed to identify the line and the cylinder is always present in an explicit format, unless the structure of the format is known the data cannot be interpreted and used.
### 4.1.2. Data structure of non-CAD data

Let us now consider a table of numbers, from which we wish to retrieve totals. Interestingly, such a table could be created using a CAD program or a graphical library, but in this case the numbers would not be treated as such unless the software is capable of managing and processing logical information, which is not always the case. The contents of the tables are the same (the individual numbers), but the totals will depend on the direction of summation (by row, by column, diagonally, etc.). Therefore, while containing the same data, the two tables shown hereafter will have to be interpreted differently. In the first case the summation will be conducted by column, in the second case the summation will be conducted by row:

<table>
<thead>
<tr>
<th>1</th>
<th>2</th>
<th>3</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>2</td>
<td>3</td>
</tr>
<tr>
<td>1</td>
<td>2</td>
<td>3</td>
</tr>
<tr>
<td>3</td>
<td>6</td>
<td>9</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>1</th>
<th>1</th>
<th>1</th>
<th>3</th>
</tr>
</thead>
<tbody>
<tr>
<td>2</td>
<td>2</td>
<td>2</td>
<td>6</td>
</tr>
<tr>
<td>3</td>
<td>3</td>
<td>3</td>
<td>9</td>
</tr>
</tbody>
</table>

For a slightly more abstract example, let us consider a common planning situation, such as the need to identify which parts of the ship are scheduled for assembly on a certain date, and to cross check the availability of a hired crane of sufficient lifting capacity to move the finished assembly to another location in the yard. While the ship parts can be represented graphically, and even the crane for that matter, cross-checking assembly completion date and crane availability probably cannot. To go one step further, let us consider the case of the crane becoming unavailable at the last minute, e.g. due to a mechanical failure, and the ensuing consequences due to the fact that the assembly hall remains unnecessarily occupied by the completed assembly.

Once again, CAD data (which parts of the ship will be affected) and non-CAD data (the planning corrective measures to be taken) are related and inter-dependent, and it will be beneficial for both data types to be documented in an exploitable fashion and be available together.

### 4.2. Data contextualization

For the purpose of this work, data contextualization is intended as the intrinsic meaning of the data itself. For example, a line could represent another object altogether, such as a stiffener, and a number could represent a weight or a frequency, or a person, etc.

The intrinsic meaning of data creates and controls the need for data to be available, found and interpreted in a given context. For example, weight data is relevant in the context of weight reporting but, more to the point, the context itself will drive the use of that data. Consider the need to either know the weight of an object or the total weight of a group of objects. In the first case we are dealing with direct data reporting, while in the latter we are dealing with data processing at first, and data reporting next. Going further, the total weight of one type of object might have to be multiplied by a unit cost, while the total weight of another group of object will be multiplied by a different unit cost. So, contextualization extends beyond the initial data itself, to the realm of data processing.

Data contextualization is generally achieved by the use of words, such as titles, descriptions, etc., which works well in the case of a printed document, but less so in the case of electronic data storage and transferring, as the method is limited and additional interpretation is required: e.g., how many
different meanings one same word can take depending on the context. Other, but simpler and more limited ways to contextualize data is to use colours, layers, file names, etc.

Data structuring and formatting is a pre-requisite to the contextualization of electronic data, and effective use of contextualization more often than not requires handling of the data itself, as opposed to simply accessing data storage. For example, the title of a drawing or of a file may indicate the presence of certain data, but will directly supply neither data itself nor its structure, which must be searched for, retrieved, interpreted and possibly processed.

4.2.1. Contextualization of CAD data

Going back to the example of the line and the cylinder, here are a few different possible meanings for these graphical entities:

<table>
<thead>
<tr>
<th>possible meaning of a line</th>
<th>possible meaning of a cylinder</th>
</tr>
</thead>
<tbody>
<tr>
<td>distance</td>
<td>hollow pipe</td>
</tr>
<tr>
<td>trace of a stiffener</td>
<td>solid column</td>
</tr>
<tr>
<td>centreline of a pipe</td>
<td>tank</td>
</tr>
<tr>
<td>edge of a plate)</td>
<td>hole in a solid</td>
</tr>
<tr>
<td>etc.</td>
<td>etc.</td>
</tr>
</tbody>
</table>

It can be argued that contextualization of CAD data is necessary for the data to be useful.

4.2.2. Contextualization of non-CAD data

non-CAD data is perhaps even more in need of contextualization. To use the tables of numbers example again, although the two tables contain the same numbers, each table may refer to different quantities, such as weight and thickness. Moreover, each number itself may represent something different. Hence, data structuring must be developed as required to correctly contextualize each piece of data.

4.3. Data inconsistency

While other words may be more fitting than "inconsistency", its general meaning accurately represents the practical problem presented by the need to associate data of the same type, but in different groups and to different extents. The theory of ensembles is a simple way to illustrate the above:

The practical task is to structure data in such a way that CAD and non-CAD data can be associated in any desired manner. E.g., a line representing a stiffener may have a colour, the stiffener may have a different colour, and the planned date of assembly of that stiffener will not have a colour assigned to it at all. Yet, line, stiffener and date are associated. What is more, the colours themselves may be de-
scribed differently, such as by name, by number, by pantone codes, by RGB percentages, etc. So, managing data inconsistency requires the documentation of relations and associations. One effective way to stipulate associations is the use of relational data sets. Relational data sets are thereby contextualized to the deepest level, as required to associate the individual data elements. This can be achieved by using potentially large numbers of files and an overall management software to link the data in the files, or by using relational databases.

5. Databases

Using files and corresponding management software does not offer unique, outstanding advantages and, another diminishing factor, carries the potentially disastrous disadvantage of the data not being collected in a single repository. This makes it very vulnerable to damage and loss, and difficult, if not impossible to share and to associate with data from other sources. Relational databases, such as SQL, can be as immediately human-readable as plain text files, and additionally offer several significant advantages:

- the data is collected in one container (the database file)
- scripts, queries and procedures provide a rational way to store, retrieve, pre- and post- process data, and insulate the data from the user thereof
- data integrity is protected
- the database engine automatically manages concurrent access to same data by multiple users
- association of data belonging to different databases is a feature by design
- the amount of data that can be contained in a single database and the number of associations are virtually unlimited

Moreover, reviewing ship data models, databases and associated CAD files, if any, tend to occupy 2.5 times less data storage space than file-only systems, on average. The author having direct experience mostly with the ShipConstructor database, this has been used in the research and development work presented hereafter.

6. The information model

A detailed review of the ship data model, Danese (2010), can be summarized, in terms of data content, by the flowchart in Fig. 4.
Differently than most if not all other ship data systems, the flowchart shows data flowing back into ShipConstructor's database. In other words, any non-CAD data directly related to any "object" in the ShipConstructor model can, and should be stored in the ShipConstructor database. Two crucial comments are in order here:

- "objects" in the ShipConstructor database are not limited to graphical and physical objects, such as lines or solids, but include logical entities, such as a branch in any of the user-defined hierarchical model structures, or a stock.
- the ability to collect and most importantly associate CAD and non-CAD data is the sine-qua-non criteria which must be satisfied in order to qualify a true data and information model as such. In fact, it becomes appropriate at this point to use the "information model" label in its full right, as the "data" connotation becomes, by definition, an underlying pre-requisite.

Moreover, such an information model is no longer limited to the ship per-se, but involves and affects other, farther-ranging domains, such as company management, extra-ship PLM, etc. One final note to the above considerations is that data can also flow from the ShipConstructor database to any other data repository. Therefore, the conditions to define an information model can be resumed to:

- it must be possible to associate CAD and non-CAD data, from different sources
- it must be possible to import data from and export data to other concerned software systems

7. The ShipConstructor information model: macroscopic structure and data flow

Macroscopically, the ShipConstructor information model is based on a single SQL database and on AutoCAD files. Data is exchanged and synchronised between the SQL database and the AutoCAD files via a direct, real-time, transparent interface. Moreover, data can be input from both inside and outside the ShipConstructor environment, via AutoCAD and via SQL.

7.1. Macroscopic structure

In short, the AutoCAD file contains a copy of every ShipConstructor object created in it (lines, solids, relations, associations, etc.), while the SQL database contains all ShipConstructor objects created either in AutoCAD files, and directly in or via SQL, e.g. using the graphical SQL database manager, as well as all the libraries of objects, relations, associations, etc. and the project's environment settings. This combination, and in some instances the managed duplication, of CAD and non-CAD data is fundamental to the achievement and exploitation of the multi-source input/output data flow described hereafter.

7.2. Data flow

While the words "import" and "export" describe the flow of data, these terms generally refer to the use of intermediate data exchange files. In the case of ShipConstructor, while classic import and export are supported, the tools available to exploit the data contained in ShipConstructor information model actually allow for the direct exchange of data, provided of course that other software involved in the process does, too.

Data flow is described in some detail in the following paragraphs, but one aspect thereof deserves foremost mentioning: the purposely built-in by-design ability to interact with the ShipConstructor information model using commercial off-the-shelf, shrink-wrapped tools, be they software in their own right (ex. Crystal Reports, Oracle/SQL interfacing software, MS-Office, etc.), or programming tools (ex. C#, VB, Lisp, etc.). This is possible thanks to several features offered by the ShipConstructor environment, such as:

- an increasing number of ShipConstructor commands are exposed in the AutoCAD environment and can be scripted.
- database interaction is made virtually transparent thanks to the ShipConstructor API, stored procedures, and exposed queries.
- access to raw data is permitted (but not encouraged).
- user-defined tables may be added to the ShipConstructor SQL database.
- the ShipConstructor SQL database may be linked to other databases.
- last and perhaps most powerful, the ability to interact with the ShipConstructor information model in a remote, IT neutral fashion, using standard communications technology.

Much work has already been carried out in this respect, and a few examples are presented hereafter.

7.2.1. Data input

In terms of data creation, several mechanisms are available in the ShipConstructor environment:
- CAD input, via the AutoCAD environment, using AutoCAD commands, ShipConstructor functions and non-ShipConstructor functions and programs.
- non-CAD input, via the dedicated ShipConstructor graphical SQL database interface, and via non-ShipConstructor functions and programs.

It is very important to note that reference is made here to "input", as opposed to "data". The distinction is fundamental, and it is made because the ShipConstructor user will create and input CAD and non-CAD data alike via both the AutoCAD and the SQL environments. So, graphical entities, modelling schemas, parameters, associations, etc. will be defined in some instances via the CAD AutoCAD environment, and in other instances via the non-CAD SQL database environment. A special note is in order at this point to underline the importance of CAD and non-CAD data associating mechanisms, specifically using input from multiple sources. One remarkable feature offered by ShipConstructor is the User Defined Attribute (UDA), a particularly powerful and versatile tool in several respects:
- UDA data can be input from both inside and outside the ShipConstructor environment
- UDA data can be virtually anything: numbers, strings, hyperlinks, soon also formulas, etc.

7.2.2. Data output

Many output options are available, but a concise summary of output "types" will suffice here:
- graphical (ex. paper drawings, pictures, etc.)
- tabular (ex. reports)
- to neutral file formats (ex. dxf, pdf, dwf, rtf, etc.)
- to proprietary file formats (ex. MS-Excel, NavisWorks, etc.)
- via off-the-shelf software (ex. Crystal Reports, MS-Office, etc.)
- via custom applications (ex. targeted harvesting from the AutoCAD and SQL environments)

It is very important to note that data and information can be extracted from the ShipConstructor environment without need for AutoCAD and/or ShipConstructor, directly from project database. Therefore, the contents of the ShipConstructor information model become available to virtually anyone at any time, and more particularly they inherently become a dynamic component of company management, tactical and strategic decision making.

8. Development examples

Most of the few examples illustrated here have been contributed by several fellow researchers, who therefore deserve the credit, and have been chosen specifically to cover a range of different contexts:
- CAD environment: custom labelling of ShipConstructor parts.
- SQL environment: custom reports of ShipConstructor data.
- combination CAD / database environment: combining CAD and non-CAD ShipConstructor data into User Defined Attributes, in order to embed custom Product Hierarchy information into a standard ShipConstructor Bill of Materials table (BOM).
- SQL environment: external input from an IT neutral, non-ShipConstructor source.
- SQL environment: combining data from two ShipConstructor databases, then processing it.
- SQL environment: automating the acquisition of planning and scheduling information from third party suppliers using wireless technology such as WAN (2G : GSM / CDMAone, 3G : UMTS / CDMA2000, 4G : LTE / WiMax), LAN (Wi-Fi) or PAN (ZigBee).
- Company scale example: exploitation of the open ShipConstructor Information environment in the early phases of estimation.

8.1. CAD environment application: Custom labelling of ShipConstructor parts

Custom labelling was developed to accommodate specific drawing detailing and presentation requirements. Custom labelling is effectively an AutoCAD application, and uses data contained in the AutoCAD file. It directly harvests data from the ShipConstructor objects present in the current AutoCAD (as opposed to a X-ref file), and uses the harvested data to create formatted, persistent text strings (the parts’ labels). The labels are stored in the file, and can be printed. The following example shows one possible code set for interrogating the ShipConstructor parts and for harvesting the data:

```csharp
public static DetailInfo GetDetailInfo(Guid DetGuid, out Point3d p3d)
{
    DetailInfo DI = new DetailInfo();
    DI.GUID = DetGuid;
    double x = 0, x1 = 0, y = 0, y1 = 0, z = 0, z1 = 0;
    Guid guid = Guid.Empty;
p3d = new Point3d(0,0,0);
    SCon.DataLayer.StructPart.STRUCT_PlateParts platePart = new
    SCon.DataLayer.StructPart.STRUCT_PlateParts();
    platePart.Connection = SConApp.ProjectSettings.Connection;
    platePart.LengthConversionUnit = GEN_LengthUnit.MM;
    platePart.FillPKGUID(DetGuid);
    GEOM_3DPoints points3d = new GEOM_3DPoints();
    points3d.Connection = SConApp.ProjectSettings.Connection;
    points3d.DataLevel = DataLevelEnum.Basic;
    points3d.LengthConversionUnit = GEN_LengthUnit.MM;
    if (platePart.MoveNext() == SCon.DataLayer.Gen.DLBOOL.DLTRUE)
    {
        DI.Name = platePart.PlatePartName;
        DI.Stock = platePart.eStockDescription;
        DI.Material = platePart.eMaterialName;
        guid = platePart.ExtMaxPointGUID;
        points3d.FillPKGUID(guid);
        if (points3d.MoveNext() == SCon.DataLayer.Gen.DLBOOL.DLTRUE)
        {
            x = points3d.X;
            y = points3d.Y;
            z = points3d.Z;
        }
        guid = platePart.ExtMinPointGUID;
        points3d.FillPKGUID(guid);
        if (points3d.MoveNext() == SCon.DataLayer.Gen.DLBOOL.DLTRUE)
        {
            x1 = points3d.X;
            y1 = points3d.Y;
            z1 = points3d.Z;
        }
p3d = new Point3d((x + x1) / 2, (y + y1) / 2, (z + z1) / 2);
    return DI;
}
```

Fig.5: Custom label example
Fig.6: Custom labels management dialog
8.2. SQL environment: custom reports of ShipConstructor data

Custom reporting was developed to accommodate specific report formatting and content requirements in a single report, and more specifically:

- custom-defined part count (quantities) and custom-defined part grouping.
- reporting user-selected part properties, including those embedded in the part object as text.

This custom reporting example is written in Crystal Reports, and only requires access to the SQL database. It will be helpful here to describe how a ShipConstructor part is “made”. In short, it is composed dynamically from a collection of associated information including CAD data (geometry, text, et.) and non-CAD data (attributes, properties, associations, parameters, User Defined Attributes, etc.). So, custom reports are generated by harvesting data from the ShipConstructor database, and arranging into a custom-formatted report sheet, as follows:

- in the ShipConstructor database, find the required parts employing user-defined criteria.
- harvest the parts' information and collect it in a matrix.
- create a table using the desired parts and the desired, specific part information.

The following example shows one possible code set for interrogating the ShipConstructor database and harvest the required data:

```csharp
void GetData()
{
    int num = 0;
    GEN_Units unitDataset = new GEN_Units();
    STRUCT_PlateParts plateParts = new STRUCT_PlateParts();
    STRUCT_PlatePartStringAttributes plateAtt = new STRUCT_PlatePartStringAttributes();
    STRUCT_PlatePartStringAttributeValues plateAttVal = new STRUCT_PlatePartStringAttributeValues();
    STRUCT_PlatePartBevels plateBevels = new STRUCT_PlatePartBevels();
    STRUCT_PlatePartFlangeStandardJoins plateFS = new STRUCT_PlatePartFlangeStandardJoins();
    STRUCT_PlatePartTextMarkings plateTM = new STRUCT_PlatePartTextMarkings();
    STRUCT_ExtrusionParts extrusionParts = new STRUCT_ExtrusionParts();
    STRUCT_ExtrusionPartStringAttributes extPartsSA = new STRUCT_ExtrusionPartStringAttributes();
    STRUCT_ExtrusionPartStringAttributeValue extPartsSAV = new STRUCT_ExtrusionPartStringAttributeValue();
    STRUCT_ExtrusionPlots extPlots = new STRUCT_ExtrusionPlots();
    STRUCT_ExtrusionPlotSheets extPS = new STRUCT_ExtrusionPlotSheets();
    STRUCT_CorrugationParts corrParts = new STRUCT_CorrugationParts();
    STRUCT_CorrugationPartStringAttributes cpSA = new STRUCT_CorrugationPartStringAttributes();
    STRUCT_CorrugationPartStringAttributeValue cpSAV = new STRUCT_CorrugationPartStringAttributeValue();
    STRUCT_CurvedPlateParts currPlateParts = new STRUCT_CurvedPlateParts();
    STRUCT_CurvedPlatePartStringAttributes cppSA = new STRUCT_CurvedPlatePartStringAttributes();
    STRUCT_CurvedPlatePartStringAttributeValue cppSAV = new STRUCT_CurvedPlatePartStringAttributeValue();
    STRUCT_TwistedExtrusionParts tep = new STRUCT_TwistedExtrusionParts();
}
```

Fig. 7: Custom reports, management dialog
8.3. Combination CAD / database environment: combining CAD and non-CAD ShipConstructor data into User Defined Attributes, in order to embed custom Product Hierarchy information into a standard ShipConstructor Bill of Materials table (BOM)

This application exploits the flexible nature of User Defined Attributes (UDA) to create and document a custom-defined, post-processing hierarchical sequence. The post-processing sequence is described in a verbose fashion and, in this case, uses the names of Product Hierarchy branches according to the company's specific production strategy. The post-processing sequence is documented in the Nest Drawing, by embedding the desired UDAs into the user-defined ShipConstructor Bill of Materials table (BOM) - the BOM is laid out using keywords, which refer to data in the database. The custom information is used by the NC-machine operator to sort the cut parts according to each part's post-processing sequence. This a mixed AutoCAD / ShipConstructor application, using data from both the AutoCAD file and from the project database. The application works as follows:

- interrogate the Nest file to identify the parts therein (AutoCAD environment)
- then, from the SQL database, harvest the Product Hierarchy structure corresponding to each part, starting at the individual part's level, then upstream (SQL environment)
- for each part, populate the corresponding UDAs in the database with the appropriate branch name (write to the database)

The following example shows a possible code set for populating the UDAs by writing to the database, after having interrogated the ShipConstructor Nest file to identify the parts present, and having harvested their relevant data from the database.

```csharp
while (plateParts.MoveNext() == DLBOOL.DLTRUE)
{
    bool find = false;
    try
    {
        if (plateParts.ePrimaryAssemblyGUID == null)
        {
            continue;
        }
    }
    catch { continue; }
    string attVal = ChangeAttributeValue(plateParts.ePrimaryAssemblyGUID, ref find);
    if (!checkCondition(plateParts.ePrimaryAssemblyGUID))
    {
        continue;
    }
    if (find)
    {
        plateAttVal.Clear();
        plateAttVal.FillFromPlatePart(plateParts.PlatePartGUID);
        try
        {
            while (plateAttVal.MoveNext() == DLBOOL.DLTRUE)
            {
                plateAtt.FindGUID(plateAttVal.PlatePartStringAttributeGUID);
                if (attributes[plateAtt.StringAttributeGUID] == AttName)
                {
                    numAtt++;
                    plateAttVal.AttributeValue = attVal;
                    plateAttVal.IsDeferred = DLBOOL.DLFALSE;
                }
            }
        }
    }
}
```
plateAttVal.Update(UpdateMethod.AbortOnError);
}
}
catch
{
    
}

Fig.8: Selection dialog: choosing the UDAs to be populated with Product Hierarchy branch names

Fig.9: Part of a BOM with UDAs

8.4. SQL environment: external input from an IT neutral, non-ShipConstructor source

The example described here is a small part of a much larger ShipConstructor-assisted company management environment. Its peculiarity is that its functionality is available through any html compatible environment, irrespective of operating system or device brand. For lack of a proper name, it is referred to as RDM (remote data management).

The goal is to "de-materialize" access to and management of selected ShipConstructor data contained in the SQL database, under the MS-Windows OS. Of course not all data should be modifiable from the outside, and not everyone should have access to all data. The ShipConstructor database is accessed via IIS services, which are a standard part of the Windows OS distribution and installation. This is achieved very simply by a password controlled log-on, which also allows one to dynamically create a user-specific graphical interfaces, tailored to cater to the user's requirements and assignment, in accordance with his permission clearance. While programmatically rather complex, the application is very simple in concept, and works as follows:
- from any html compatible device connected to the internet or to a LAN/WAN network, the user connects to the IP address of the PC hosting the RDM application.
- the RDM sends the log-on html GUI to the device.
- the user logs-on with name and password.
- the RDM sends the user-specific GUI to the device.
- the user operates using keyboard, mouse, touch-screens, etc.
- the RDM establishes and manages the connection to the database.

While much of the source code is proprietary and is not presented in this document, interface read and write examples are illustrated in Fig. 11. The application has been tested successfully using the following devices: PC (various versions of Windows and web browsers), BlackBerry, Android mobile phones, IPhone, etc.

Fig.11: The RDM log-on/off dialog in Google Chrome (note the IP address in the url field), and the dialog for editing UDAs via the remote html interface.

8.5. SQL environment: Combining & processing data from two ShipConstructor databases

Custom reports can be as simple as specially formatted tables, or as complex as full data processing engines involving data from different sources. With reference to other examples in this document, the results of data processing can be written back to the desired database(s), e.g. by populating a UDA. This allows data processing outside the ShipConstructor environment, while collecting and storing raw data and results in it. Custom reports are based on data harvesting, and subsequent data processing. With reference to the examples presented in this paper, the Crystal Reports document, Figs. 12 and 13, harvests data from two ShipConstructor databases and processes it into a combined result.

Fig.12: Crystal Reports - connecting to 2 different databases
Fig.13: Crystal Reports - drag & drop the desired fields from the database tables to the report layout

Fig.14: The data from two different SQL database sources, and its processing into overall totals
8.6. Exploitation of the open ShipConstructor Information environment in early estimates

The following information flow schema was developed to exploit the open ShipConstructor information environment as of the early phases of estimation. The schema is presented succinctly, and reference is made to the information presented so far.

8.6.1. The ShipConstructor information environment comprises (summary)

- AutoCAD drawings, used for 3D model input and output. It is a familiar, configurable environment, and, contains both ShipConstructor and non-ShipConstructor data.
- 3D realistic, solid model: attribute and property rich. Properties include CAD and non-CAD data. Model is fully integrated and multi-disciplinary, gives overall geometrical and non-geometrical data.
- SQL database: offers export facilities, supports external harvesting of data, reports include user-selected data in user-defined formats, etc.

8.6.2. Estimation using the single model approach, extensible to design and production

This approach consists in modelling a scalable, representative Unit in detail, and the rest of the ship macroscopically (scalable primary structure such as main frames, bulkheads, decks, etc., representative secondary structure, primary and large schedule distributed systems, etc.). This data and ensuing information is then used as-is or further processed, e.g.:

- as qualitative input to create or validate scheduling and planning, e.g. in MS Project.
- to feed preliminary purchasing and required resources estimations (ERP/MRP) and other administrative and management tools (ex. SAP), which in turn will influence the preliminary planning and scheduling.

The CAD portion of the data is computed automatically and stored as object properties, such as weights, number of plates to cut, weld lengths, etc. The non-CAD portion of the data will include user defined processing entities, such as blasting grit grade, paint and finish specifications, assembly sequence, etc., and user specific parameters such as manufacturing hours as a function of yard resources and task complexity, etc. non-CAD data is yard specific, user supplied and added to the model via object properties and User Defined Attributes.

Information model dependent reports include the relevant data from the SQL database, sorted and grouped according to the user’s requirements, either using the ShipConstructor reporting engine, or other software, like Crystal Reports. Data is further processed as required to obtain the relevant quantities for use in management, tactical and strategic decision making.

Raw and processed data is thus available for immediate use, as direct input for building an MS-Project Gantt chart, or for further processing by ERP/MRP and management software.

Other non-CAD information also influences the planning (availability of materials and resources, other objective constraints) and the structure of the ShipConstructor Information model. For example, the availability of steel may influence the planned assembly sequence.

8.6.3. Integration of planning, cost analysis and model

One or more iterations may be needed to synchronise the model (choice of yard processing and build sequence, CAD modelling schedule to produce required data, etc.) and the overall planning, also with respect to cost and other objective constraints. Estimated and projected data are input back into the model (ex. model “by” dates, nest “by” dates, names of assigned operators, revised processing information such as finish, assigned resources, etc.), principally in UDAs associated with model parts and non-CAD objects (ex. the projected elapsed time needed to complete the assembly of a branch of a
Product Hierarchy based on available resources at the planned moment in time and on objective difficulty and complexity factors, as well as the corresponding cost). As the information is now collected and managed by several connected tools, it becomes appropriate to use the Information Model definition, the ShipConstructor Information Model being part thereof. As cross-feedback balances out, the Estimation model can be considered achieved.

From this point on, data flow continues to be managed at least multi-directionally between ShipConstructor, planning / scheduling software (ex. MS Project), ERP/MRP software (ex. MARS) and management software (ex. SAP), but perhaps also to/from other data sources (ex. a subcontractor suffers a set-back, or a strike forces re-tasking and re-scheduling, or bad weather delays painting, or the price of a sourced item drops thereby allowing its purchase and its fitting sooner, etc.). Throughout the process, the interface between information and management (which may not be proficient in CAD and other discipline specific software) is achieved without the direct use of any of the programs generating the raw data (ShipConstructor, AutoCAD, SAP, etc.), but rather via graphical collaborative reviewing software (ex. Autodesk's NavisWorks) and dynamically reporting software (ex. Crystal Reports).

8.7. SQL environment: automating the acquisition of planning and scheduling information from third party suppliers using wireless technology such as WAN (2G : GSM / CDMAone, 3G : UMTS / CDMA2000, 4G : LTE / WiMax), LAN (Wi-Fi) or PAN (ZigBee).

The ability to write information to the ShipConstructor database opens innumerable doors towards catering to many management requirements. The example presented here is based on a rather common circumstance, the geographical distance between a cutting company and the shipyard(s) where those parts will be assembled. In fact, the geographical distance often entails crossing of borders varying transport regulations, perhaps vastly different meteorological conditions, traffic jams, etc. Therefore, real-time knowledge of the location of a truck transporting cut parts is of great interest. While geolocalization has become common place technology, it was sought to make available relevant information about the cargo being carried in an operator-less fashion. Therefore, programmable active/passive RFID technology was added to existing GPS/GSM geo-localization. In short, the planned system, based on current off-the-shelf technology and hardware, will works as follows:

- upon loading of cut parts on a pallet or flat rack, an RFID tag attached to the pallet or to the flat rack is programmed with a tracking number and the list of parts (obtained from the NC-cutting file or other ShipConstructor supplied information).
- upon leaving the premises, the tag connects with the active transponder of the RFID system, and the following information is sent to the Information Model via GSM: pallet / rack tracking number, list of parts.
- at pre-set time intervals, the geo-localisation portion of the tag signals its GPS position, which is fed to the Information model in wireless fashion.
- finally, upon entry of the shipyard's grounds, another active transponder reads the tag's contents and feeds the arrival data to the Information model.
Incidentally, when produced in small quantities, the cost of the tag and of the corresponding active RFID transponder described above have been estimated to be a few tens of Euros, negligible at the scale of the ship industry.

9. Conclusion

The use of off-the-shelf technology and tools to compose and exploit a true Information Model has been documented and advocated before, Danese (2010). The few examples of completed and ongoing development work discussed here represent only a small cross section of what has already been achieved, and are aimed at documenting the ease and simplicity of building the basic bricks of an Information Model, a goal within just about everyone’s reach, given that the required data lends itself to the exercise. Of course, requirements change as new tools become available, and vice-versa. However, experience would suggest that the use of off-the-shelf software, hardware and technology greatly reduces the amount of resources to be expended in catering to change. This intrinsically scalable approach is then deemed viable.

Acknowledgements

A special thanks for supporting the author’s research work is owed to Pedro Antunes (Vera Navis, Portugal), Luis Batista (Vera Navis, Portugal), Stefano Cipriani (Nice RFID, France), Darren Larkins (ShipConstructor Software Inc., Canada), Deniz Morais (ShipConstructor Software Inc., Canada), Lambertus Oosterveen (Royal Huisman, The Netherlands), Justin Paquin (ShipConstructor Software Inc., Canada), Stefan Raev (Royal Huisman, The Netherlands), Peykan Sahelnia (PS Development, France), Arkadiy Zagorskiy (Marine Technologies, Russia)

References

DANES, N. (2010), Ship CAD systems - past, present and possible future, COMPIT, Gubbio, pp. 396-408
Generating Contract Scenarios for the Conceptual Design Optimization of Non-Transport Vessels

Stein Ove Erikstad, NTNU, Trondheim/Norway, stein.ove.erikstad@ntnu.no
Siri Solem, NTNU, Trondheim/Norway, siri.solem@ntnu.no

Abstract

The ship design and deployment optimization problem (SDDP) is aimed at developing the outline specification for non-transport, service type of ships, such as offshore support vessels, FPSOs and floating LNG production vessels. These vessels typically face a set of contracts or market opportunity scenarios with different start-up periods, durations and vessel capability requirements, and the design problem can be modeled as the assignment of vessels from a pool of possible designs to available contracts in order to maximize total revenue. In this paper we particularly look into how to generate contracts and market opportunities in alternative scenarios.

1. Introduction

Determining the appropriate design specification of a new ship represents an important strategic decision for a shipowner as part of a fleet renewal or expansion programme. This decision involves a high financial risk and a long time horizon, typically in the range of 30-40 years. For transport ships, this problem is typically modelled as part of a routing problem for a given expected demand of transportation services, either for the design of single ships, Jansson and Shneerson (1985), Garrod and Miklius (1985), or with a fleet size and mix perspective, Dantzig and Fulkerson (1954), Bellmore (1968), Fagerholt (1999).

For non-transport ships, such as offshore support vessels (OSVs) and FPSOs, this approach is less relevant. Here, we may instead model the future operating scenario of the vessel as a set of contracts to which the vessel might be assigned during its economic life. These contracts may vary in their duration, from long-term contracts lasting the lifetime of the vessel, to short-term contracts lasting for a year, a season or less, and in terms of their requirements for vessel capabilities and capacities. The general tendency is that these requirements become more demanding over time.

The decision problem associated with this situation can be modeled as a Ship Design and Deployment Problem (SDDP), Erikstad and Fagerholt (2011). The SDDP formulation supports the concurrent identification of the optimal ship design and the corresponding optimal deployment of the vessel. This is a key strategic decision problem primarily for shipowners, but also for stakeholders such as offshore contractors, often entering into long-term leasing contracts of 5-10 years, and ship design consultants that increasingly take a more active role in the development of the outline specification in close collaboration with the customer, Ulstein and Brett (2009). This contributes to a more well-informed “requirements elucidation” process that is beneficial for both parties, supporting a better balance between opportunity revenue and vessel capability cost, Andrews (2003).

Thus, when determining the specification for a new vessel the shipowner needs to strike a balance between optimizing the vessel for the (likely) first contract entered into, while investing in additional performance capabilities that provide a sufficient degree of capability and flexibility to meet future contract requirements. To further add complexity, these additional capabilities might either be made part of the vessel at the design time, or they may be provided as design options, to be called dependent on information made available in the future.

The approach taken here is based on the assumption that we are able to create one or several scenarios that capture the expectations about the future market situation for the vessel(s). These scenarios can be based on actual contracts that will become available on the market, or they can be based on more general expectations about future market opportunities. Each of these contracts or market situations
will specify a set of requirements for vessel capabilities, such as cargo capacity, deck area, bollard pull, operating depth, etc.

Naturally, there is a substantial degree of uncertainty related to the definitions of these scenarios. For a short time planning horizon, the scenario(s) may be based on actual contracts available in the market, or specific markets or offshore areas expected to be available. For the longer time horizon, a larger set of contracts may be generated randomly from a chosen distribution, thus at best representing a realistic, but not a real, scenario.

We believe that the best approach is to combine these two approaches, by using a limited set of "signal market events" to guide the generation of a larger set of concrete contracts to which the vessel may be assigned. How these contract scenarios may be generated will be the main focus of the remaining part of this paper, which is organized as follows: In Section 2 we summarize the SDDP model, and provide a small illustrative example. In Section 3 we discuss alternative strategies for the contract development process. In Section 4 we provide a computational study, followed by a final discussion and conclusion.

2. The Ship Design and Deployment Problem for Non-Transport Vessels

Thus, we have a set of contracts $N$, each one described by the following attributes

$$\{T^*_t, D_t, R_t, [\psi^*_1, \psi^*_2, \ldots, \psi^*_n]\},$$

where $T^*_t$ is the starting time, $D_t$ is the duration, and $R_t$ is the revenue of contract $t$. $[\psi^*_1, \psi^*_2, \ldots, \psi^*_n]$ is contract $t$'s set of capability requirements for the vessel to serve the contract, where $n$ is the total number of requirements. Examples of such capability requirements might be the operating depth, the vessel range, crane capacity, bollard pull, service speed, etc.

Thus, we have a set of vessels types $V$, each described by $[C_v, \psi^*_1, \psi^*_2, \ldots, \psi^*_n]$, where $C_v$ is the cost of acquiring vessel $v$, and $[\psi^*_1, \psi^*_2, \ldots, \psi^*_n]$ is a set of vessel capabilities. A vessel type $v$ is said to be compatible with contract $t$ if its capability $\psi^*_v$ is sufficient to match the corresponding contract capability requirement $\psi^*_t$ for all requirements, i.e. for $k = 1, 2, \ldots, n$.

$$\{C_v, [\psi^*_1, \psi^*_2, \ldots, \psi^*_n]\}$$

$$\{T^*_t, D_t, R_t, [\psi^*_1, \psi^*_2, \ldots, \psi^*_n]\}$$

![Matching contract requirements and vessel capabilities](image)

Fig.1: Vessel capabilities are matched with contract requirements to form vessel specific contract scenarios for which the optimal deployment path can be found

In the pre-processing of the model, the contracts capability requirements will be matched with the capabilities of the vessels in the vessel pool, so that for each vessel we derive a subset $N_v \subseteq N$ that contain the specific set of contracts for which vessel $v$ is compatible. In Table I a small example is described. Further, the future operating scenario is modeled as nine available contracts, Table II.
Table I: Vessel design alternatives

<table>
<thead>
<tr>
<th>Vessel type number</th>
<th>Cost</th>
<th>Contracts available</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>30</td>
<td>1,2,4,5,9</td>
</tr>
<tr>
<td>2</td>
<td>35</td>
<td>2,3,5,6,7</td>
</tr>
<tr>
<td>3</td>
<td>34</td>
<td>1,2,3,5,8,9</td>
</tr>
<tr>
<td>4</td>
<td>28</td>
<td>3,4,6,7,8</td>
</tr>
</tbody>
</table>

Table II: Contracts scenario

<table>
<thead>
<tr>
<th>Contract number</th>
<th>Contract period</th>
<th>Revenue</th>
<th>Vessels capable</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>2011-2013</td>
<td>12</td>
<td>1,4</td>
</tr>
<tr>
<td>2</td>
<td>2012-2016</td>
<td>25</td>
<td>1,2,3</td>
</tr>
<tr>
<td>3</td>
<td>2011-2020</td>
<td>30</td>
<td>2,3,4</td>
</tr>
<tr>
<td>4</td>
<td>2016-2020</td>
<td>20</td>
<td>1,4</td>
</tr>
<tr>
<td>5</td>
<td>2013-2015</td>
<td>21</td>
<td>1,2,3</td>
</tr>
<tr>
<td>6</td>
<td>2016-2018</td>
<td>12</td>
<td>2,4</td>
</tr>
<tr>
<td>7</td>
<td>2017-2019</td>
<td>18</td>
<td>1,4</td>
</tr>
<tr>
<td>8</td>
<td>2012-2014</td>
<td>15</td>
<td>3,4</td>
</tr>
<tr>
<td>9</td>
<td>2014-2016</td>
<td>15</td>
<td>1,3</td>
</tr>
</tbody>
</table>

In Fig. 1 we show the result of matching the requirements of the contracts with the capabilities of each vessel type 1 from Table I. This corresponds to a longest path network as given in Fig.2. Nodes S and T are artificial source and sink nodes, respectively.

![Fig.1: The contracts available (grey bars) for vessel 1](image1)

![Fig.2: Longest path network for vessel type 1](image2)
We can generalize this into a mathematical model that considers both the vessel selection and the deployment simultaneously. We start by defining the notation before the model is presented.

Sets:
- \( V \) The set of vessel types that are available, indexed by \( v \)
- \( N \) The set of contracts available
- \( N_v \) The set of contracts that are compatible with vessels of type \( v \), \( N_v \subseteq N \)
- \( A_v \) The arcs in the network that can be traversed by a vessel of type \( v \)

Parameters:
- \( R_{\ell v} \) Net revenue for servicing contract \( \ell \) with a vessel of type \( v \) (the contract’s revenue deducted the variable cost of using the vessel for that contract)
- \( C_{v} \) Cost of choosing a vessel of type \( v \) (net present value of capital costs and running costs that do not depend on the deployment of the vessel)

Variables:
- \( x_{ijv} \) Binary variable equal to 1 if a vessel of type \( v \) traverses arc \((i,j)\), and 0 otherwise

It should be noted that if \( x_{ijv} = 1 \), it means that both contracts \( i \) and \( j \) are serviced by a vessel of type \( v \). Now, the SDDP can be formulated as follows:

\[
\max \sum_{\ell \in N} \sum_{v \in V} R_{\ell v} x_{\ell v} - \sum_{v \in V} \sum_{\ell \in N_v} C_{v} x_{\ell v},
\]

subject to

\[
\sum_{v \in V} x_{\ell v} = 1, \quad \ell \in N
\]

\[
\sum_{v \in V} x_{ifv} - \sum_{v \in V} x_{fiv} = 0, \quad \forall f \in N_v
\]

\[
\sum_{v \in V} x_{ijv} = 1, \quad \forall i \in V, (i,j) \in A_v
\]

Objective function (1) calculates the maximum profit of the chosen path in the network. Constraints (2) ensure that only one vessel is selected by stating that exactly one arc from the source node should be used. Continuity is ensured by constraints (3), while constraints (4) correspond to (2) and state that exactly one arc should be used to the sink node. Constraints (4) are redundant because of constraints (2) and (3), but are included for consistency. Binary requirements for the flow variables are imposed by constraints (5). By solving (1) – (5), both the optimal vessel to acquire, as well as the deployment of the vessel during the given planning horizon are determined. In *Erikstad and Fagerholt (2011)* this problem is solved even for large instances of both vessels (300) and contracts (150) with solution times of approximately 3 minutes. In the same paper, the basic SDDP presented above is extended to capture problem variants such as an existing fleet of vessels and multi-vessel contracts.

3. Alternative strategies for contracts scenario development

The insight gained into important design considerations through this approach will to a large extent depend on the quality and realism of the future contracts scenarios. There will naturally be a substantial degree of uncertainty related to the development of these scenarios. For a short time planning horizon, the scenario(s) may be based on actual contracts available in the market, or specific markets or offshore areas expected to be available. For the longer time horizon, a larger set of contracts may be generated randomly from a chosen distribution, thus at best representing a realistic,
but not a real, scenario. Thus, the deployment plan of the chosen vessel should not be interpreted literally, but rather as a testing scenario for which valuable insight into important design tradeoffs can be obtained.

Scenarios containing real contracts for the planning horizon of the vessel on the one hand, and randomly generated scenarios based on generic distributions on the other hand, represent two opposite extreme positions in scenario generation. By combining the two, we derive a strategy where a set of high level field development event serve as seeds for a cloud of actual contracts generated in the neighborhood of the guiding event. Here, “neighborhood” relates to both the time and requirements domain, guiding the contract startup period and the minimum required capabilities of the vessel serving the contract. This approach reflects that opportunities typically arise as a consequence of large scale developments of complete oil and gas fields, where a single country or oil major is the primary responsible, and where a staged and coordinated development plan is made publicly available for the market. One example is the Brazilian sector. Here, Petrobras has made public a concrete, long-term plan for a staged development, with corresponding estimated needs of both exploration and production facilities, as well as support services. Another example, but with less explicitly formulated requirements and time schedule, is Shtokman natural gas field in the Barents Sea, which in the near future is likely to be followed by other developments in the High North.

Fig. 3: A contract scenario developed by combining high level guiding events with generic contracts from a uniform distribution. Here, the time periods are shown on the horizontal axis, and the required operating depth of the contract on the vertical axis.
In Fig.3, an illustrative example showing the development of a contract scenario based on this combined strategy. Here we see four guiding events generating a number of derived contracts, in addition to more generally available contracts in the market. For each of the derived contracts, the requirements of the guiding event is set as a minimum, and with diminishing probabilities of having requirement values far above this minimum value.

For a given contract scenario, we then have to match the contract’s requirements with that of the vessel design capabilities. For each design we can thus derive a set of available contracts, for which we can generate a network model as was illustrated in Fig.2. The complexity of this network will naturally depend on the capability level of the vessel, since highly specified vessels are able to serve more contracts than lower specified ones. An example of the derived network for one vessel for a scenario of 50 contracts is given in Fig.4. This will be solved as binary integer programming problem, simultaneously determining the vessel design with the highest revenue over the relevant time horizon, as well as the corresponding optimal (longest) path of contracts to take.

![Network of contracts](image)

**Fig.4:** The network of possible contract paths for a given vessel for a scenario of 50 contracts

### 4. Handling uncertainty in future contracts scenarios

Even if we so far have generated the contract scenarios partly based on random parameters, the actual selection of the optimal vessel design and corresponding deployment has been based on solving a deterministic longest path problem. An improvement of this model would be to explicitly take into consideration that the future is uncertain. One possible approach is to develop strategies for identifying design solutions that are robust towards changes in the contractual scenarios, in terms of performing well across a range of possible values of the scenario parameters.

There are several alternative approaches for identifying such robust solutions. One is to rather select the vessel with the $k$-longest paths through the network, thus avoiding the dependence on one or a few specific realization of the future. Another alternative is to apply a replacement path algorithm, capturing alternative contractual options resulting from the cancellation of a planned contract. A third alternative is to apply a Monte Carlo based sampling of future contractual scenarios.

The actual optimal design will depend on the sampled scenario. Based on 200 randomly generated scenarios as illustrated in Fig.5, the distribution in “winning” counts is given in Fig.6. Here, we see that no particular vessel is the optimal one in all generated scenarios. Rather, we obtain a first feedback on which designs seem to be better able to handle alternative realizations of an uncertain future, thus pointing towards possible robust solutions to be further investigated.
5. Summary and conclusions

In this paper we have discussed alternative approaches for generating future contract scenarios to be used as part of a Ship Design and Deployment Problem. This can be used to develop vessel designs design specification that balances the cost related to providing vessel capabilities, towards the additional revenue of being able to select from a larger number of more highly paid contracts.

The scenarios are generated partly based on explicit expectations of future market opportunities, such as offshore field developments with specific capability requirements, and partly by drawing realistic single contracts from a given distribution. In addition, actual known contracts, such as the typical startup assignments for a new vessel, can be added to the scenarios as well. We further discuss how this approach can be extended to handle the obvious uncertainty related to the actual realization of the future, in terms of identifying design specifications that are likely to perform well across a number of possible scenarios.

References


Optimal Hull-Form Design Subject to Epistemic Uncertainty

Matteo Diez, INSEAN, Rome/Italy, m.diez@insean.it
Daniele Peri, INSEAN, Rome/Italy, d.peri@insean.it

Abstract

The paper presents an application of ship design optimization subject to epistemic uncertainty. Generally, uncertainty that results from intrinsic variability (e.g. stochastic environmental or operating conditions) is termed aleatoric. Conversely, uncertainty is described as epistemic when it results from a lack of knowledge about a quantity whose true value (continuous or discrete) exhibits no actual variability (e.g. lack of information in the current stage of the design). Following a Bayesian approach, all the uncertainties are assigned a probability density function (PDF). Specifically, quantities with aleatoric uncertainty may be modelled as random variables through constructing their PDFs from observed data. Conversely, parameters with epistemic uncertainty may be assigned a PDF and the probability is inherently the degree of belief in a proposition. The present work addresses the epistemic uncertainty related to the location of the ship centre of gravity (CG), when optimizing the hull form. Following a Bayesian approach, a probability density function (PDF) of the CG location is defined. Two complimentary formulations are presented and applied to the hull form optimization.

1. Introduction

The paper presents a hull-form optimization subject to epistemic uncertainty. When handling a design optimization problem, it is likely to deal with two general types of uncertainty. The uncertainty is named aleatoric, e.g. Najm (2009), Iaccarino (2008), if it results from an intrinsic and stochastic variability. This type of uncertainty includes the variability on the environmental and operating conditions, as well as different and probabilistic mission profiles. Conversely, the uncertainty is termed epistemic, Najm (2009), Iaccarino (2008), when it results from a lack of knowledge about a quantity whose true value is not yet defined at the current stage of the design. The latter case includes all those parameters, which are hypothesized (with a certain degree of uncertainty) at the design stage when the optimization is performed, even if they are intrinsically deterministic.

The application presented in this paper can better explain the latter point. Here, the epistemic uncertainty related to the ship centre of gravity location (CG) is assessed and included in the hull-form optimization. Specifically, we are in the case of the following hypothesis: a) at the design stage when the hull form has to be finalized, the CG location is still unknown, b) once the entire ship design is concluded, then the CG is deterministically defined and located (the interested reader is suggested to look through the final discussion of the work by Diez et al. (2010), where the present theme was raised by Dr. J. Kuhn). The problem is solved as follows. Adopting a Bayesian approach, a probability density function (PDF) of the CG location is defined, and the PDF is inherently the degree of belief in the proposition. The decision problem is formulated following two complimentary approaches.

In the first formulation, the expectation and the standard deviation (with respect to CG variations) of a suitably defined performance measure are considered as the optimization objectives. This first problem is twofold. On one hand, the task is that of finding the hull-form configuration which minimizes the so-called Bayesian risk, thus defining the Bayesian solution to the decision problem, De Groot (1970). On the other hand, the problem is that of finding the hull-form which is more insensitive to the uncertainty realizations. In both cases, the overall optimization task can be interpreted as a robust design optimization (RDO) procedure, e.g. Diez and Peri (2010). Specifically, the method requires the evaluation of the expected value and standard deviation of the merit factor. These are evaluated by integrating the performance measure over the uncertain parameters domain. The resulting uncertainty quantification (UQ) of the problem is conducted here by Gauss-Legendre quadrature.

In the second approach presented here, the CG location affects a design constraint. The problem is
handled by identifying a probabilistic feasible region where the optimization procedure is allowed to move. For this second formulation, the perspective is that of reliability-based design optimization (RBDO, e.g. Tu et al. (1999), Agarwal (2004), Agarwal and Renaud (2004), and the following procedure is here adopted. The expectation and the standard deviation of the probabilistic constraint are evaluated through the UQ. Then, the one-sided Chebyshev’s inequality (also called Cantelli’s inequality, e.g. DasGupta (2010), is applied to quantify the upper bound of the probability of violation of the constraint itself, thus defining the reliability of the current design. Designs with a reliability index smaller than a critical value are disregarded during the optimization.

For both formulations, the constrained minimization problem is solved using a particle swarm optimization (PSO) algorithm, Campana et al. (2006).

2. Design optimization problems affected by uncertainty

Generally, an optimal design problem may be formulated as a constrained minimization problem. The deterministic approach to optimal design, is described by the following mathematical problem

\[
\begin{align*}
\text{minimize} & \quad f(x, y) \quad \text{for} \quad y = \hat{y} \in B \\
\text{subject to} & \quad g_n(x, y) \leq 0 \quad \text{for} \quad n = 1, \ldots, N \\
\text{and to} & \quad h_m(x, y) = 0 \quad \text{for} \quad m = 1, \ldots, M
\end{align*}
\]

where \( x \in A \) is the design variables vector (which represents the designer choice), \( y \in B \) is the design parameters vector (which collects those parameters independent of the designer choice, like, e.g., design specifications, environmental and operating conditions), and \( f, g_n, h_m : \mathbb{R}^k \rightarrow \mathbb{R} \), are respectively the optimization objective, the inequality and equality constraints functions. Depending on the application, different kind of uncertainties may arise in the optimization process, Beyer and Sendhoff (2007), Park et al. (2006), Zang et al. (2005). Specifically, in the latter problem the following uncertainties may occur (the interested reader is also referred to Diez and Peri (2010).

a) Uncertain design variable vector. When translating the designer choice into the “real world,” the design variables may be affected by uncertainties due to manufacturing tolerances or actuators precision. Define a specific designer choice \( x \) and define \( \xi \in \Xi \) the error or tolerance related to this choice; \( \xi \) may be described as a stochastic process with probability density function \( p(\xi) \).

b) Uncertain design parameters and specifications. Design parameters and specifications may differ from the deterministic design conditions \( \hat{y} \) (see the minimization problem of Eq. (1) because of intrinsic variability (aleatoric uncertainty) or because of a lack of knowledge at the current design stage (epistemic uncertainty). Therefore, the design parameters vector is assigned a probability density function \( p(y) \).

c) Uncertain evaluation of the function of interest (also called “errors”). The evaluation of the functions of interest (objectives and constraints) may by affected by uncertainty due to inaccuracy in modelling or computing (errors). Collect the original deterministic objective function and constraints in a vector \( \varphi := [f, g_1, \ldots, g_N, h_1, \ldots, h_M]^T \), and assume the assessment of \( \varphi \) for a specific deterministic design point, \( \varphi := \varphi(x, \hat{y}) \), be affected by a stochastic error \( \omega \in \Omega \) with a probability density function \( p(\omega) \).

Combining the above uncertainties, we may define the expected value of \( \varphi \) as

\[
\mu(\varphi) = \int_{\Xi B \Omega} \varphi(x + \xi, y + \omega) p(\xi, y, \omega) d\xi dy d\omega
\]

where \( p(\xi, y, \omega) \) is the joint probability density function associated to \( \xi \), \( y \) and \( \omega \). It is apparent that the
expectation of \( \varphi \) is a function of the only designer choice. Moreover, the variance of \( \varphi \) with respect to the variation of \( \xi, y, \) and \( \omega \) is

\[
\sigma^2(\varphi) = \iint \left\{ \varphi(x, y) + \omega \right\} - \varphi(x^*) \left\{ \varphi(x, y, \omega) d\xi dy d\omega
\]

resulting, again, a function of the only designer choice \( x \). Eqs. (1) and (2) give the general framework of the present work. Specifically, in the following we will concentrate on the uncertainty related to some geometrical parameter which is independent of the designer choice, thus referring to uncertainties of the \( b \) type. Specifically, the vertical location of the centre of gravity (CG) is assumed to be unknown at the current stage of the design (epistemic uncertainty). Following a Bayesian approach, a probability density function is assigned to the vertical CG and the expectation and the standard deviation of the relevant functions reduce, in this case, to

\[
\mu(\varphi) = \int_B \varphi(x, y) p(y) dy
\]

\[
\sigma^2(\varphi) = \int_B \left[ \varphi(x, y) - \varphi(x) \right]^2 p(y) dy.
\]

being \( y \) the uncertain quantity under analysis.

3. Robust decision making for optimal design subject to epistemic uncertainty: robust design optimization (RDO)

In this section, the formulation for robust design optimization (RDO) subject to epistemic uncertainty is presented. Assume that the optimization objective in the problem of Eq.(1) is associated to a general loss (like, for instance, the performance loss with respect to a given target). Under the hypothesis of uncertain CG location, we may refer to \( f(x, y) \) as the loss associated to the designer choice \( x \), when the condition \( y \) on the CG occurs. Therefore, the expectation of the loss \( f \), evaluated through the integral of Eq.(2), may be defined as the risk associated to the decision \( x \) under the distribution \( p(y) \), De Groot (1970). It follows that the designer should choose, if possible, a decision \( x \) which minimizes the risk (expected loss).

Specifically, if we consider the Bayes risk, i.e. the lower bound of the expected loss for all the possible choices in \( A \),

\[
\tilde{f}^* := \inf_{x \in A} \mu(f)
\]

we look for the Bayes decision of the problem against the distribution \( p(y) \), for which the risk equals the Bayes risk. The Bayes approach to the designer decision problem may be formulated as follows.

\[
\text{minimize w.r.t. } x \in A, \quad \mu(f) := \int_B f(x, y) p(y) dy
\]

subject to

\[
\sup_{y \in B} \left\{ g_n(x, y) \right\} \leq 0 \quad \text{for } n = 1, \ldots, N
\]

and to

\[
\mu(h_m) := \int_B h_m(x, y) p(y) dy = 0 \quad \text{for } m = 1, \ldots, M
\]

In the latter formulation, the inequality constraints are treated in the worst case, following a conservative approach. An alternative method is that of considering the inequality constraints as probabilistic inequalities, and assessing the reliability of the design with respect to the constraints violation. The latter approach is the main idea behind reliability-based design optimization (RBDO), e.g., Tu et. al. (1999), Agarwal (2004), Agarwal and Renaud (2004), and will be addressed in the next section.
The optimal designer choice \( x = x^* \) of the problem of Eq.(3) is that which minimize the expected loss of the system performances with respect to the variation of the design parameters collected in \( y \). It may be noted that in the present context, the design specifications (CG location) are given in terms of a probability density function.

The problem of Eq.(3) may be enriched by considering the standard deviation of \( f \) (with respect to the uncertain parameters variation) as a second or as an alternative objective function. The latter approach improves the insensitiveness of the final design to the operating conditions variation:

\[
\text{minimize w.r.t. } x \in A, \; \mu(f) \text{ and/or } \sigma(f) \\
\text{subject to } \sup_{y \in E} \{g_n(x, y)\} \leq 0 \quad \text{for } n = 1, \ldots, N \\
\text{and to } \quad \mu(h_m) := \int h_m(x, y) p(y) dy = 0 \quad \text{for } m = 1, \ldots, M
\]

### 4. Reliability-based robust design optimization (RBRDO)

The main idea behind reliability-based robust design optimization (RBRDO) is that of defining probabilistic constraints in the optimization task for RDO. Specifically, the inequality constraints that appear in the problem of Eq.(4) are replaced by probabilistic inequalities as follows:

\[
\text{minimize w.r.t. } x \in A, \; \mu(f) \text{ and/or } \sigma(f) \\
\text{subject to } \quad P\{g_n(x, y) \leq 0\} \geq r \quad \text{for } n = 1, \ldots, N \\
\text{and to } \quad \mu(h_m) := \int h_m(x, y) p(y) dy = 0 \quad \text{for } m = 1, \ldots, M
\]

where \( P\{Z\} \) is the probability of the event \( Z \) and \( r \) is the reliability associated to the \( n \)-th inequality constraint. Here, the probabilistic inequalities are handled using the one-sided Chebyshev inequality (also known as Cantelli’s inequality, e.g., DasGupta, 2010). Specifically, the probabilistic statement that appear in the problem of Eq.(5),

\[
P\{g_n(x, y) \leq 0\} \geq r
\]

is replaced by the inequality

\[
\mu(g_n) + k\sigma(g_n) \leq 0
\]

where \( k \) is a real positive number. The one-sided Chebyshev inequality (Cantelli’s inequality) in this case reads

\[
P\{g_n \geq \mu(g_n) + k\sigma(g_n)\} \leq \frac{1}{1 + k^2}
\]

Combining Eq.(6) with Eq.(7) it may be shown that the minimum value of \( k \) ensuring a reliability \( r \) is given by

\[
k = \sqrt{\frac{r}{1 - r}}
\]

The above function \( k = k(r) \) is invertible. Therefore, once a real positive coefficient \( k \) is chosen, is is possible to determine the corresponding reliability \( r \). It is

\[
r = \frac{k}{1 + k^2}
\]
Tables I and II give some relevant value for the functions \( k = k(r) \) and \( r = r(k) \) respectively (see also Figs. 1 and 2).

Fig. 1: \( k \) as a function of the reliability \( r \).

Fig. 2: \( r \) as a function of the coefficient \( k \).

Table I: Relevant values for \( r(k) \)

<table>
<thead>
<tr>
<th>( r )</th>
<th>( k )</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.9</td>
<td>3.00</td>
</tr>
<tr>
<td>0.91</td>
<td>3.18</td>
</tr>
<tr>
<td>0.92</td>
<td>3.40</td>
</tr>
<tr>
<td>0.93</td>
<td>3.64</td>
</tr>
<tr>
<td>0.94</td>
<td>3.96</td>
</tr>
<tr>
<td>0.95</td>
<td>4.36</td>
</tr>
<tr>
<td>0.96</td>
<td>4.90</td>
</tr>
<tr>
<td>0.97</td>
<td>5.69</td>
</tr>
<tr>
<td>0.98</td>
<td>7.00</td>
</tr>
<tr>
<td>0.99</td>
<td>9.95</td>
</tr>
<tr>
<td>0.999</td>
<td>31.6</td>
</tr>
</tbody>
</table>

Table II: Relevant values for \( k(r) \)

<table>
<thead>
<tr>
<th>( k )</th>
<th>( r )</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>0.5</td>
</tr>
<tr>
<td>2</td>
<td>0.8</td>
</tr>
<tr>
<td>3</td>
<td>0.9</td>
</tr>
<tr>
<td>4</td>
<td>0.941</td>
</tr>
<tr>
<td>5</td>
<td>0.962</td>
</tr>
<tr>
<td>6</td>
<td>0.973</td>
</tr>
<tr>
<td>7</td>
<td>0.98</td>
</tr>
<tr>
<td>8</td>
<td>0.987</td>
</tr>
<tr>
<td>9</td>
<td>0.988</td>
</tr>
<tr>
<td>10</td>
<td>0.99</td>
</tr>
<tr>
<td>20</td>
<td>0.998</td>
</tr>
</tbody>
</table>

5. Numerical application to hull-form optimization

The current application is related to the seakeeping of a containership. The S175 containership, already adopted for intensive seakeeping activities by the ITTC seakeeping group, is here adopted as base hull. Modification of the ship hull is performed using a free-from deformation (FFD) approach, Sederberg and Perry (1986). A box, split into 5x2x3 slices (longitudinal x lateral x vertical), is bounding the hull surface, and 9 design variables are used for the modification of the lateral coordinates only. As a consequence, the keel line is fixed, as well as the lateral footprint.

A single geometric constraint is enforced during the optimization: the displacement is allowed to vary within 12,400 t and 12,600 t, where the baseline value is 12,500 t. The original deterministic objective function is defined as the vertical acceleration, \( a \), at the extreme bow of the ship and at the waterline. The acceleration is computed for a forward speed of 16 knots, with head sea. The spectrum type is that of Pierson-Moskowitz, with a characteristic period \( T_1 \) of 11 s, and \( H_{1/3} \) equal to 4.5 m.

An uncertainty on the height of the centre of gravity (CG) of the ship (\( Z_G \)) is hypothesized, with a variation between 5.4 m and 6.4 m, from the keel line and with uniform probability. Here, three different approaches are pursued and the associated solutions compared. Specifically,

- a first solution is obtained by minimizing the vertical acceleration with a prescribed value of \( Z_G \), in the middle of the interval of variation (i.e., deterministic approach with \( Z_G = 5.9 \));
• the second formulation involves the minimization of the variance (and, therefore of the standard deviation) of the vertical acceleration subject to the uncertain parameter $Z_G$;

• a third problem is solved minimizing (as before) the variance of the vertical acceleration under the uncertain parameter $Z_G$, and, at the same time, considering a probabilistic constraint on the vertical acceleration itself (RBRDO problem). The acceleration is required to be smaller than $34 \text{ m/s}^2$ with a probability of failure smaller than 0.01%. Consequently, the prescribed value of $k$ in the one-sided Chebyshev (Cantelli’s) inequality is $k = 4.36$ (see previous section).

Fig. 3 shows the different solutions obtained. Specifically, the section views are compared. The original hull, with straight sides, is depicted on the top left. The expected value of the vertical accelerations are respectively $34.87 \text{ m/s}^2$ for the original hull, $32.55 \text{ m/s}^2$ for the deterministically optimized hull, $34.06 \text{ m/s}^2$ for the hull with minimum variance (standard deviation) and, finally, $34.00 \text{ m/s}^2$ for the hull obtained through RBRDO. In the latter case, the expected value for the constraint $g := a/a_{\text{max}} - 1$ is $\mu(g) = -1.06 \times 10^{-4}$ and the standard deviation results $\sigma(g) = 7.52 \times 10^{-6}$. As a consequence, the probability of constraint violation is nearly zero. Moreover, the numerical values of the expectation and the standard deviation of the acceleration, pertaining to the original hull and the three optimized hulls are reported in Table III.

---

**Table III: Expected value and standard deviation for the four different hulls**

<table>
<thead>
<tr>
<th>Hull</th>
<th>$E(a)$</th>
<th>$\sigma(a)$</th>
</tr>
</thead>
<tbody>
<tr>
<td>Original</td>
<td>34.87</td>
<td>$1.07 \times 10^{-2}$</td>
</tr>
<tr>
<td>Deterministic</td>
<td>32.55</td>
<td>$1.46 \times 10^{-2}$</td>
</tr>
<tr>
<td>Min. variance</td>
<td>34.06</td>
<td>$7.11 \times 10^{-3}$</td>
</tr>
<tr>
<td>RBRDO</td>
<td>34.00</td>
<td>$8.69 \times 10^{-4}$</td>
</tr>
</tbody>
</table>

---

Finally, Fig. 4 shows how the different optimization procedures affect the variation with $Z_G$ of the vertical acceleration of the final solutions. It may be noted how the deterministic optimization produces the final shape with the higher dependence of the vertical acceleration on $Z_G$, while the opposite effect is obtained when the variance (standard deviation) is minimized, as expected.
Fig. 4: Dependence of the vertical acceleration on the vertical position of the centre of gravity $Z_G$.
Values are shifted in order to have zero value in the middle of the interval of variation.

6. Concluding remarks

A probabilistic Bayesian framework for ship design subject to epistemic uncertainty has been presented and discussed. The overall formulation has been applied to the hull-optimization of a containership aimed at seakeeping performances. Specifically, both RDO and RBRDO (based on the one-sided Chebyshev inequality) have been applied to the seakeeping optimization problem. The extension of the present methodology to address either resistance optimization, either multiobjective problems will be addressed in the future work.

References

AGARWAL, H. (2004), Reliability based design optimization: Formulations and methodologies, Ph.D. Thesis, University of Notre Dame, South Bend


DIEZ, M.; PERI, D. (2009), Global optimization algorithms for robust optimization in naval design, 8th Int. Conf. on Computer Applications and Information Technology in the Maritime Industries, (COMPIT), Budapest.


IACCARINO, G., (2008), Quantification of uncertainty in flow simulations using probabilistic methods, in Non-equilibrium gas dynamics: From physical models to supersonic flights, VKI Lecture Series

KENNEDY, J; EBERHARD R., (1995), Particle Swarm Optimization, IEEE Int. Conf. on Neural Networks, Perth.


Virtual Reality Supported Assembly Planning in the Shipbuilding Industry

Hermann Lödding, Axel Friedewald, Martin Heinig, Sebastian Schleusener,
Hamburg University of Technology, Hamburg/Germany,
{loedding;friedewald;heinig;schleusener}@tuhh.de

Abstract

Within the research project Power-VR, an assembly planning prototype has been developed to assist the user during the whole review process. The user is provided with an automatic session preparation as well as with methods for investigating several assembly issues. Examples are the review of a specific ship section as well as the verification of a part assembly under consideration of assembly dates of the affiliated parts. Dynamic concatenation of model data with meta data during the session offers the possibility to automatically filter the parts of the model according to different kinds of information such as the delivery dates. The prototype further supports the finding and verification of assembly sequences by providing necessary filter information, e.g. about the next part to assemble. Also an automatic model preparation for collision control is procured. The prototype can be handled immersively during the session. To further improve the control of the available functions, the use of a tablet PC is suggested. This device allows for a more flexible interaction with the system than the traditional flystick and provides additional information, adapted menus and buttons.

1. Introduction

The propagation of 3D CAD modeling in the shipbuilding industry shows the acceptance and the benefit of using computer software for the planning and construction of a vessel. As CAD software is a construction tool, the need for alternative tools such as Virtual Reality (VR) is given, e.g. for design or assembly reviews. Especially in the outfitting assembly, which is characterized by high complexity, often narrow compartments and numerous manual assembly operations, an efficient planning and reviewing software can greatly decrease planning time and reduce planning errors, Lödding and Friedewald (2010).

VR has found its way to the daily business in the automotive and aviation industry and as the costs for VR decrease, it becomes more attractive for the shipbuilding industry, Lödding and Friedewald (2010). A barrier for the implementation is still the lack of good support for the planning of outfitting processes in an efficient manner considering the high number of assembly parts. The shipbuilding process is characterized by very short development times and an assembly of complex products including miscellaneous supplier parts, Nedess et al. (2008). The supplier parts are numerous and often arrive too late. Such variations in the delivery situation can be the reason for changes of the optimal assembly sequences. Shipyards have to react to these short-term events quickly, without compromising on quality and with as little effort as possible.

The benefit of using VR in such a one-of-a-kind production can be a better and faster planning, which leads to a reduction of assembly times, assembly errors and therefore to a reduction of costs. Overall the competitiveness increases. To achieve these results, a highly customizable VR solution is needed, because every product is different and has therefore different requirements. The solution also has to be able to cope with the short processing times in the design department, Lödding and Friedewald (2010). To check the needs for changes in the assembly sequences and to develop new sequences with VR, the current status of the planning process has to be represented first, then it has to be adapted to the new circumstances. With conventional VR solutions, this is a very time consuming process and would not generate satisfying results.

2. Assembly planning in Virtual Reality

The key for the successful use of VR technology is to adapt the VR system to the business processes, Nedess et al. (2005), Bohuszewicz (2007). To achieve this goal, a general VR process model for the
shipbuilding industry has been developed. Nedess et al. (2008, 2009a) divided the process in six phases. The result is shown in Fig. 1.

![Fig. 1: The VR process, Nedess et al. (2009a)](image)

To classify the different phases, phase 1-4 cover the session preparation, phase 5 is the actual performance of the session and phase 6 is the documentation of the results. This classification shows the importance of the session preparation to the whole process of a VR session.

The usage of non-adapted standard VR systems leads to many manual activities during all phases of the process, especially during the preparation of the session with the manual selection of all relevant model parts. This selection depends largely on the objectives of the scenario and needs to be conducted within the VR software. Therefore the most time consuming part is the session preparation, Nedess et al. (2009b). To simplify the usage of VR in the shipbuilding industry, the preparation effort for the different sessions must be greatly reduced, Nedess et al. (2009a). Nevertheless the whole VR process must be supported and a concept for covering all the phases of the VR-Process was introduced with the Virtual Production-Workflow-Composer, Nedess et al. (2009b). This ViP-Composer uses specified engineering templates for the adaption of the software and also supports the user to perform the desired VR review. One additional goal of the approach was to enable departments without a profound VR knowledge to use VR solutions, Nedess et al. (2009b).

The usage of meta data can play an important role during a VR session. As the support for the usage and visualization of model specific meta data is still not satisfying, Nedess et al. (2008), an improved approach for importing, showing and using meta data in the context of the VR-Session should be developed.

The Power-VR project funded by the German Federal Ministry of Economics and Technology is a cooperation of several shipyards, suppliers and research institutes. In this project, the scenario of lately delivered parts, that causes the necessity for changes of the assembly sequence for a certain area of the vessel (Fig. 2). The investigation shows that current VR solutions have problems in supporting the necessary replanning efforts, because they do not provide proper methods for model preparation and handling. To improve the planning situation with VR reviews, a good support for the VR session is needed; otherwise the effort would exceed the benefits.

The aim is to find a new assembly sequence which enables a collision free assembly of the delayed component by identifying and rescheduling the parts, which would be assembled between the planned and the new delivery date and block the alternative assembly path. The replanning requires easy to handle and quickly available VR support.

The 3D CAD model contains all the assembly parts in their final assembly position but does not consider any assembly status, path or sequence. In comparison to initial assembly planning, the model preparation is more complicated because the model has to be reduced to the specific area of the vessel, containing only the already built in assembly parts. A manual selection of these parts is very time
consuming and error-prone. Therefore, an automatic session preparation has to be supported. This requires the concatenation of the additional information with the 3D model. Also the assembly sequence of the parts and the delivery dates have to be connected.

When the problem of a delayed part shows up, finding a solution is time-critical. In the VR process, the session preparation is the most time consuming part. Therefore, a scenario-based support for an automatic session preparation is very important. This has to cover the visualization of different assembly states on demand and a visualization of the planned assembly sequence of parts. Planning data like delivery dates must be available and usable during the session for the model preparation. The user has to be guided through the session by giving him additional information like the next step to be done or important information about assembly parts such as the weight or other relevant part properties. Tools for rescheduling need to be developed to support the finding and verification of a new assembly sequence using VR.

In this approach, the session preparation is based on dynamic information which can change. The whole process needs to be flexible and adjustable to such changes all the time. In addition, the required information has to be available at any time in an aggregated, computable database. The end of the session should result in a significant and suitable documentation of the new found assembly sequence.

Another common problem is that the person who needs the VR session is not a VR expert and needs to explain the scope of the session to the VR expert. In a more suitable approach, the VR operator (user without profound VR knowledge) prepares the functional site of the session in advance by himself and only needs the VR expert for VR specific topics. The same issue shows up during the performance of the session, when changes to the VR scene have to be made like loading an additional section of the model, or when an adaption of the assembly sequence. To defuse this discrepancy, an extended control concept is necessary to give the VR operator more functional control over the VR software.
3. Extended Virtual Production-Workflow-Composer

To suite the requirements of the scenario "delayed parts" the flexibility of a VR session must be increased. The former concept did not include a dynamic session handling, in which the customization of a session could be done during the execution. The first version of the ViP-Composer adapted the menus of the VR software but did not cover the model preparation for the session. This still needed to be done manually without possibilities for dynamic changes or automatic adaption to the session objectives.

For the usage in the VR software the existing 3D-CAD models of the vessels are converted. The correctness and the completeness of the model should be ensured as a step before beginning the VR process. A model generally consists of many thousand files and the relation between them is of static nature like the filename, the place in the model hierarchy or the coordinates in the model.

Meta data enable further connections between model parts and can group them by a specific attribute. Lödding and Friedewald (2010) introduced a classification of meta data for the shipbuilding industry which exposes a number of relevant dynamic meta information in addition to static meta data, Fig. 3.

![Fig. 3: Static and dynamic meta data](image_url)

The possibility to include meta data directly in the geometry data format is suitable only for static meta data. This implies that a different approach with a flexible coupling is needed for the connection of dynamic meta data and model data so that a change of meta information does not require a new revision of the model data.

The extended ViP-Composer concatenates the meta data and the model data dynamically during the VR session, Fig. 4, Lödding et al. (2010a). The meta data is loaded from a database into the composer and connects the data with the model objects in the scene graph on the basis of a unique key, which could be for example the part name or number. If the meta data changes, a simple reload updates the linked data. Prerequisite is an aggregated, up-to-date database and the possibility of assigning meta information to the model data, which contain the same unique key for the several parts.

The meta data is then available in the session and can be displayed. Not only has the amount of information increased, but it is also possible to select model objects based on dynamic attributes and hence manipulate the session model automatically using meta data values.
Fig. 4: Concatenation of model data and meta data

The ViP-Composer was extended with a functionality to create filters. A filter sets a constraint for a value of a specified meta data attribute of a part. This selects only the entries in the meta data list, whose attribute does not violate this constraint. Multiple filters can be combined and applied, so that for example all parts can be selected, which are in section one of the vessel and will be assembled in the next two weeks, Fig. 5. To automate an operation on the filter result, the ViP-Composer can execute different actions. An action works on the model objects in the VR scene, as the displaying or the coloring of the filtered objects. The combination of filters and actions enable an automatic session preparation and a dynamic session customization during the session, which will be shown with the help of the earlier described scenario of delayed parts.

The ViP-Composer assists the user during the phases of the VR process. The first phase is the definition of objectives. In this case, it is the change of a planned assembly strategy for a certain area of the vessel. This leads to an adaption of the planned assembly sequence, if the assembly path of the delayed part will be blocked by other parts. The ViP-Composer offers two types of templates, professional templates and VR templates. The professional templates include the objective specific sets of necessary filters and actions. They provide support for collision handling as well as the possibility to display and change assembly sequences. The VR templates offer predefined menu configurations and additional tools like a crane geometry or optimization tools, which may be needed during the VR session, but are not directly related to the objective, *Nedess et al.* (2008).

The next phases are the import and preparation of the data for the VR session. The user can now set the parameters of the filters according to the desired inspection scope (like the section of the ship and the time period) and choose the action “load parts”. This can be done without any knowledge of VR or the use of the specific VR software because the filters are defined in a xml structure. The session relevant model parts are now selected via the meta data and automatically loaded into an empty session. This significantly reduces the session preparation time and avoids selection errors caused by a manual selection of the parts. The pre-selection of the relevant parts without loading the complete model has another advantage: Especially in the shipbuilding industry, the used models are in general very big and can exceed the performance limit of the soft- or hardware, *Nedess et al.* (2008). The filtering process significantly reduces the model size and thus increases the system performance.
The preparation of the VR scene in this example consists of a combination of two filters and the action “assembly sequence”. In the earlier phase, all relevant parts have been loaded into the session. However, the model is not yet prepared for the investigation of an assembly sequence. The two filters select all parts that are assembled between the planned assembly date of the delayed part and the newly scheduled assembly date. The “assembly sequence” action puts the result in the assembly sequence table, ordered by the planned sequence derived from the assembly dates. It also prepares the parts for collision control and turns the delayed component transparent.

The user can now perform the VR session and view the animated assembly sequence, see which part is delayed and can stop the animation at any time to perform a disassembly of this part with the help of collision control. It is also possible to animate the assembly sequence step by step in both directions (forward and backward in time), which is controlled by an additional menu. The assembly sequence is adaptable via the assembly sequence table. Changes are directly visible within the assembly sequence animation and can be checked for collisions. The newly derived assembly sequence gets validated immediately as shown in Fig. 6.
If the meta data contains additional information to a certain part in the assembly sequence, this information is displayed in the 3D scene during the sequence. As outlined in the scenario description, the session should be dynamic, in case new conclusions require additional model parts in the session or meta information changes. Reloading meta data as well as adjustments of applied filters remain possible during the entire session.

Fig. 6: Validation of assembly sequences

Fig. 7: Extended ViP-Composer
The last phase is the **documentation** of the obtained results. The newly derived assembly sequence is documented in the VR scene itself as an animation and is also recorded in the assembly sequence table. It can be saved back as meta data and can easily be distributed. Overwriting existing meta data without permission or review of the person in authority for these information should be avoided; therefore a revision control for meta data is suggested.

Fig. 7 illustrates the concept of the extended ViP-Composer with the usage of meta data. The extended ViP-Composer supports all phases of the VR process and provides improved session handling possibilities. The described scenario of delayed parts is completely supported. However, a new interaction concept is needed to fully benefit from the specified approach. The next chapter covers this topic.

### 4. Extended interaction concept

The first phase of the VR process introduced in chapter 3 is the definition of objectives for the VR session. This is assisted with functional and VR specific templates. The functional templates cover the engineering aspects of the VR session and the VR specific templates cover the aspects of the VR session tool. This differentiation is necessary, because a user of a tool not always is an expert of this tool. This issue was explained in Lödding et al. (2010b), where a classification of user groups according to the profile of requirements or competencies has been made for production simulation. The groups are simulation operators and simulation experts, transferred to this context, VR operators and VR experts.

The VR operator knows exactly what should be done during the session and how the results should look like, but he does not know how to do it. The VR expert assists the VR operator in executing the session, but as he does not exactly know about the investigated problem, the aid can only be as good as the communication between them. Therefore the VR operator should be able to prepare and handle a session.

This approach was pursued with the development of a VR construction kit, which defines templates with predefined configurations for VR sessions, Nedess et al. (2009d). For the session preparation, the VR operator can parameterize the chosen templates and can afterwards easily operate with the extended and configured menus and tools defined in the template for the VR software. The traditional desktop setup with a mouse and a keyboard enables the necessary interactions. During a session, the VR operator works immersively with a flystick to interact with the software. The flystick offers only the selection of functions via an immersive menu. Even if the menu is extended with additional functions, the limitation of the static interaction with the session remains, because there is no possibility to enter parameters or complex commands.

Transferred to the filter model of the ViP-Composer, this means that a filter could be activated or deactivated. The parameters, for example the assembly date, could however not be changed immersively. Nedess et al. (2008) recommend, that tools and menus of the VR software need to be structured and directly available during a VR session to maintain the ongoing problem discussion. In this case the VR operator needs to either delegate this operation to a second person at the PC or has to leave the immersive workspace, which in both cases violates the suggestion. The extended interaction concept covers this “discontinuity of handling” during the ongoing session.

The requirements for the new concept are based on the idea of an immersive dynamic session handling. A VR operator should be able to customize the actual session by himself without leaving the immersive workspace. Hence the new interaction device needs to be portable and lightweight. It should offer the possibility to enter commands or text and should be able to display additional information, which reasonably should not be projected in the 3D scene. An easy access to often used functions or menus without selection via the immersive menu will be helpful. This is faster and has the advantage, that the 3D model is not obscured by the immersive menu which can be beneficial during an animation for example. Further requirements can be voice control for easy and fast input of text or the use of the orientation sensor for navigation in the 3D scene. The applicability of a tablet PC for
session control was investigated. An overview of the interaction possibilities compared to the traditional flystick is given in Table 1.

<table>
<thead>
<tr>
<th>Device</th>
<th>Tracking</th>
<th>Orientation sensor</th>
<th>Voice control</th>
<th>Input of text</th>
<th>Additional user interface</th>
<th>Display of information</th>
</tr>
</thead>
<tbody>
<tr>
<td>Fystick</td>
<td>+</td>
<td>0</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>Tablet PC</td>
<td>0</td>
<td>+</td>
<td>+</td>
<td>+</td>
<td>+</td>
<td>+</td>
</tr>
</tbody>
</table>

The easy handling of a tablet PC and its touch display offer the necessary capabilities for the extended interaction concept, Fig. 8. Based on the performance of the VR session with the ViP-Composer in chapter 3, an improved handling of the session with a tablet PC has been prototypically implemented. This is shown with the help of the introduced scenario about delayed parts.

In contrast to the conventional session handling, the tablet PC allows to control the functions of the ViP-Composer in the immersive workspace. The defined filter and actions from the session preparation and the model itself can be manipulated during the session, Fig. 9. The necessary parameters can be entered or another action may be selected. This is important, if the session needs to be different from the initial preparation state, e.g. if additional model parts need to be loaded. Meta information is displayed on the tablet PC and does not obscure the VR scene, but provides additional information. The functions of the assembly planning like starting and stopping the assembly sequence animation are provided via the touch screen and are easily accessible, Fig. 9. They do not need to be selected over the immersive menu and do not disturb the ongoing assembly sequence. The usage of the tablet PC is intuitive and does not need special VR knowledge. Rescheduled dates of assembly parts as well as comments can be entered via the tablet PC.
5. Conclusion

The shipbuilding process is characterized by very short development times. Therefore, VR sessions need to be done more quickly and easily than with current solutions. This article shows that the extended ViP-Composer provides complete support for the entire VR process. This includes a concept for the concatenation of dynamic meta data and model data. This allows for a fast and automatic model manipulation as well as the usage of planning information. This significantly reduces time and occurring errors during the session preparation and execution.

The provided planning information can be used for the assembly sequence planning and can also be changed during the VR session. The ViP-Composer immediately visualizes the rescheduled sequence as an animation and directly conducts a collision control. The results and documentation of the changes can be saved as meta data. The introduced extended interaction concept describes the possibility to control the ViP-Composer during a VR session from the immersive workspace. In comparison to the traditional flystick, the tablet PC improves the manipulation, display and input possibilities for the VR operator. The need for VR expert knowledge is reduced. A prototype of the extended ViP-Composer is currently evaluated in cooperation with a German shipyard.

References


LÖDDING, H.; FRIEDEWALD, A.; WAGNER (2010b), Rule-based resource allocation – an approach to integrate different levels of planning detail in production simulation, 9th Conf. on Computer and IT Applications in the Maritime Industries (COMPIT), Gubbio, pp.203-212.


Reduced Beam Model – The Seek after Adequate Accuracy with Feasible Performance in Global Structural FE-Analysis of a Naval Vessel

Roope Kotiranta, Surma Ltd, Helsinki / Finland, rmk@survivability.fi

Abstract

Despite the ever increasing performance of the computing hardware, a detailed structural analysis of a ship may still take relatively long time. Especially if the FE-analysis is used for getting feedback to the actual design process, the performance issues have a crucial role. Assessment of the effects of an underwater explosion is an example of a case for which an alternative calculation procedure could be needed. This paper describes an approach using a reduced beam model, aiming at fast and robust structural analysis with feasible performance to accommodate global assessments requiring a model of the whole vessel. A central part of the described method is the utilization of a modern general purpose ship product model as the platform and the source of information used in the FE-analysis. Another noticeable feature of this approach is that the beam model generation is automated to enable real and practical integration of the analysis with the design process.

1. Introduction – why reduced?

Anyone ever involved in a ship design project knows that with even a conventional ship it is a complex process including multidisciplinary problems, especially so, when aiming for new novel designs. The designers faced with these challenges nowadays usually have a plentiful supply of different numerical simulation tools to help towards a feasible solution, Fig. 1. Especially when naval vessels are considered, the spectra of numerically assessed features is wide, including e.g. radar cross sections, under water acoustics and other signature reduction related phenomena.

![Fig. 1: Numerical tools in a design process. [http://brlcad.org/wiki/Documentation]](http://brlcad.org/wiki/Documentation)
As understandable, most of the aspects subject to numerical simulation are coupled together, either via direct physical dependencies or less directly resulting from some other ‘weaker’, yet practically equivalent connection. This means that achieving a balanced design, requires subsequent, if not simultaneous solving of different models. This leads to a conclusion that although computational capacity is ever increasing, so is the demand from the numerical simulations. A global FE model of a frigate size vessel shown in Fig. 2 gives an idea of a typical model used for global response calculations with respect to under water explosions.

Fig. 2: Global FE model of a frigate size vessel

As it is, the increase of demand seems in this case always to be bigger than that of the capacity, thus calling for some form of 'down shifting' regarding these simulations to enable taking all the necessary aspects into the iterative design loop.

This paper discusses an approach which aims to lower the computational demand of the dynamic structural analysis. The method used for this purpose is the so called reduced beam model. In the following chapters the main principles and the benefits as well as the disadvantages of this procedure are reviewed. One of the key features, namely the automated generation of this model, based on a modern product model, is also shortly described. Some planned validation and development plans are also briefly discussed, hence ending the paper with a few summarizing observations.

2. Reduced Beam Model

2.1. Background – the motivation

Before going any further, a few words on SURMA are probably in order. Survivability Manager Application (SURMA), is meant to be a practical tool enabling the designer to include survivability as any other feature, hence capacity, endurance, cost, etc. in the iterative ship design process. Even if the speed and robustness of selected calculation procedures reflects this goal, the fidelity and resolution of these assessments should at least facilitate reliable comparisons between different designs. Thus the absolute values from the design perspective are not necessarily needed.

Relating to the above mentioned considerations, simplifications are applied when ever thought
appropriate. One aspect, which due to its fairly complex nature would otherwise require huge computational resources, is dynamic structural analysis. In SURMA, this is currently taken care of by presenting the structure of interest as an equivalent single degree of freedom system.

Despite many benefits of the SDOF system, hence rapidity and stability, it still possesses certain limitations. Probably the worst of these is the one inheriting from the definition, that is to say it has only one degree of freedom, thus it's fairly hard, if not impossible to connect reasonable SDOF systems into a physically consistent entity. Another shortcoming is that when analyzing extreme cases, the collapse mode has to be predefined, since the SDOF system has no 'built in' capabilities to predict this. To overcome at least these two defects, it was decided to replace the SDOF system with finite three dimensional beam elements, which are created following basically the same principles as with the SDOF system.

2.2. Automated creation

SURMA is using NAPA product model as the modeling and simulation platform. This enables the integration of a bit more exotic analysis into the conventional design environment, but also the automation of many repetitive and tedious tasks. Such an undertaking is also the construction of the reduced beam model. Effectively the same procedure that is utilized for generating the SDOF models, can be used here. It follows the path illustrated in Fig. 3. Basically the product model is divided into some natural discreet entities, which can be presented with beam elements. The configuration and properties of these beams can then automatically be derived from the structural definition of the product model.
The beauty of this scheme is that all the information incorporated into the product model can also be applied in the reduced beam model. This information may include the structural arrangement, material data, and non structural weight distribution, just to mention a few. An example of a fairly high resolution beam model is shown in Fig. 4.

![Fig. 4: A high resolution beam model created from NAPA product model](image)

3. Usage – possibilities and blind spots

3.1. Benefits with respect to intended use cases

As the development of SURMA is driven by the needs of a military solution for combat survivability assessment, one of the scenarios SURMA needs to tackle is the under water explosion. Because the beam model constitutes a coherent physical entity representing the whole ship, global aspects, such as the whipping response can now be analyzed. With the formerly used SDOF system this would have been impossible.

Although the first whipping modes of the hull girder can also be fairly accurately calculated using a 'stick' model, i.e. a collection of beams created by dividing the girder into discreet parts along its length. Catadi (1989), the reduced beam model offers more. One interest with respect to whipping, are the accelerations it excites in the various sensitive equipment on-board, and a fair estimate for this is achievable when combining the reduced beam model with the equipment arrangement of the NAPA product model, which SURMA also utilizes. An example of this kind of equipment arrangement is illustrated in Fig. 5.
Since the reduced beam model is still rather a simple numerical model, it exhibits certain advantages over the more complicated ones. The first evident virtue is the computational efficiency, which easily imaginable from the comparison between the reduced beam model and an FE model created for the same purpose presented in Table I.

Table I: Comparison between reduced beam model and FE model

<table>
<thead>
<tr>
<th>Model</th>
<th>Elements</th>
<th>Nodes</th>
<th>DOFs</th>
</tr>
</thead>
<tbody>
<tr>
<td>RBM</td>
<td>1279</td>
<td>1138</td>
<td>6828</td>
</tr>
<tr>
<td>FEM</td>
<td>798973</td>
<td>70621</td>
<td>423726</td>
</tr>
</tbody>
</table>

Another remarkable benefit is that the beam model doesn't exhibit as many local modes as can be encountered when assessing the natural modes with a detailed FE model. Thus it's less laborious to detect the real global modes using e.g. the method based on modal masses.

Yet one noticeable feature of the reduced beam model is the numerical stability. The used beam elements are immune to problems like shear or volume locking, which can cause difficulties with shell and solid type elements, *Hallquist (2006)*.
3.2. Known limits

Although a list of assets can easily be recited, it clear that some penalties must be paid to reach this. One mode of collapse, which might be encountered even with 'normal' loading conditions of a ship is buckling. It can be clearly understood from the illustration in Fig. 6 that a simple beam presentation is insufficient to capture this phenomena.

When dealing with weapons effect related damage, the reduced beam model behaves in very conservative way, because the loss of structural strength of even one beam can have drastic local and global effects with respect to the neighboring elements. It is also to be noted, that the mapping of concentrated loads, e.g. pressure loads from explosions isn't as straightforward as with a model consisting of shell elements. Also the above mentioned accelerations experienced by the equipment can be expected to be on the conservative side, due to the fact that the coarser the discretization, the stiffer the model and the smaller the degree of structural and numerical damping existing in it.

4. Validation and future development

4.1 Why to validate

Although the theoretical aspects related to the reduced beam model seem clear and well understood, experimental testing is the only test bench which may or may not convince an engineer of the usefulness and reliability of any calculation method.

The first experimental validation case currently planned is the modal analysis of a small patrol vessel, in which the first global modes will also be measured on-board after exiting the actual vessel with a hydraulic shaker. Beside the measurements, the analysis will also be performed with two more sophisticated numerical models.

Another aspect of interest from the view point of combat survivability is the assessment of air blast damage. For this purpose, half-scale trials with fairly primitive structures are currently conducted by the Finnish Navy. Unfortunately, due to schedule issues, no results are available at the time when writing this paper.

4.2 A glance forward

SURMA currently employs an external non-commercial FE solver, which is used for solving the FE analysis with the reduced beams. Although this has a negligible effect on the computational performance, the first planned step ahead is to incorporate a similar solver into the SURMA core code. Besides decreasing the execution time, this also makes the data exchange between the simulation and the product model seamless, hence more robust.

5. Conclusions

This paper presented the motivation and an approach to find a practical compromise between simplified and highly sophisticated numerical structural analysis, aiming to reduce the computational demand of this kind of simulations. A reduced beam model was suggested as a feasible option, and the benefits and tradeoffs of this method were discussed on practical as well as more theoretical level. The need and the plans for certain practical applications were reviewed and finally the imminent scheme of development was shortly described.

References

http://brlcad.org/wiki/Documentation


Discrete Event Production Simulation and Optimisation of Ship Block Erection Process

Jean-David Caprace, University of Liège, Liège/Belgium, jd.caprace@ulg.ac.be
Clarice Trevisani Da Silva, University Federal of Rio de Janeiro, Rio de Janeiro/Brazil, clarice@peno.coppe.ufrj.br
Philippe Rigo, University of Liège/Belgium, ph.rigo@ulg.ac.be
Floriano Carlos Martins Pires, University Federal of Rio de Janeiro, Rio de Janeiro/Brazil, floriano@peno.coppe.ufrj.br

Abstract

Nowadays, shipyards are making every effort to efficiently manage equipments and resources such as labourers, gantry cranes, transporters, steel and block stock yards, etc. The block erection scheduling of a gantry crane has thus far been manually performed by a manager of a shipyard. Such a scenario leads to undesirably long times for producing scheduling results. In addition, the quality of the scheduling results may not be optimal. To improve the overall process, block erection discrete event simulations have been developed in this study by using techniques of optimization. The first results presented in this paper are promising even if some future work must be realized.

1. Introduction

Production simulation is a very useful tool concerning the possibilities of gains in the process of production and as result, cost reduction. In order to achieve an optimum integration design vs. production, it is necessary to model not only the ship but also the shipyard facilities and integrate them into a single simulation model. Best results are achieved when this model is linked to other optimization systems. The simulation allows finding the best workshop layout and assembly sequence according to the building strategy of the ship.

2.1 Production simulation in shipbuilding industry

The simulation of shipbuilding process can be useful to assess, decide and communicate manufacturing planning's, allowing a dynamic and transparent review of the production. The technique can help the project definition of the vessels, or the assessment of production, according of different types of vessels, Kasemaker et al. (2006). During the last decade, shipyards, research centres and universities started to use this powerful tool to analyze shipbuilding operations. The group SimCoMar (Simulation Cooperation in Maritime Industries) is an example of an initiative to accelerate the development of simulation in the industry, helping North American and European shipyards. The Flensburger Nordseewerke Emden shipyard, the universities TUHH (TU Hamburg-Hamburg), DUT (Delft University of Technology) and Anast (University of Liège), and the Center of Maritime Technology (CMT) in Germany are participating in this initiative. Besides SimCoMar, other partnerships have been established between shipyards and universities such as the University of Seoul South Korea, Japan's Kinki, Michigan University, and Federal University of Brazil (LABSEN laboratory).

In recent years, the Dutch and German shipbuilding industry is seeking to reduce delivery times, production costs and increase product quality, using the process simulation. Some German yards are well advanced in the use of simulation and integration solutions to environmental planning processes, such as Meyer Werft and Flensburger.

2.2 Discrete Event Simulation Software's

Currently many simulators are available commercially. Some of them were compared with information obtained from manufacturers, users (Internet discussion groups), for articles published in
congresses and simulation for manuals and textbooks. Table I lists some features of each program as the application price, model visualization, popularity, etc.

Table 1: List of DES software

<table>
<thead>
<tr>
<th></th>
<th>ARENA</th>
<th>PROMODEL</th>
<th>PLANT SIMULATION</th>
<th>FLEXSIM</th>
<th>QUEST</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Application Price</strong></td>
<td>Good</td>
<td>Good</td>
<td>Very poor</td>
<td>Poor</td>
<td>Very poor</td>
</tr>
<tr>
<td><strong>Easy to Learn</strong></td>
<td>Poor</td>
<td>Very poor</td>
<td>Poor</td>
<td>Average</td>
<td>Average</td>
</tr>
<tr>
<td><strong>Custom Extensions</strong></td>
<td>Average</td>
<td>Average</td>
<td>Good</td>
<td>Poor</td>
<td>Very Good</td>
</tr>
<tr>
<td><strong>Technical Capacity</strong></td>
<td>Average</td>
<td>Very poor</td>
<td>Very Good</td>
<td>Average</td>
<td>Very Good</td>
</tr>
<tr>
<td><strong>Model Visualization</strong></td>
<td>Poor</td>
<td>Very poor</td>
<td>Very Good</td>
<td>Average</td>
<td>Very Good</td>
</tr>
<tr>
<td><strong>Graphical User Interface</strong></td>
<td>Poor</td>
<td>Poor</td>
<td>Average</td>
<td>Average</td>
<td>Poor</td>
</tr>
<tr>
<td><strong>Modularity</strong></td>
<td>Good</td>
<td>Good</td>
<td>Good</td>
<td>Average</td>
<td>Good</td>
</tr>
<tr>
<td><strong>CAD connection</strong></td>
<td>Average</td>
<td>Average</td>
<td>Good</td>
<td>Good</td>
<td>Good</td>
</tr>
<tr>
<td><strong>Thechnical Support</strong></td>
<td>Poor</td>
<td>Very poor</td>
<td>Average</td>
<td>Very poor</td>
<td>Average</td>
</tr>
<tr>
<td><strong>Popularity (forums)</strong></td>
<td>Good</td>
<td>Good</td>
<td>Poor</td>
<td>Good</td>
<td>Poor</td>
</tr>
<tr>
<td><strong>Compatibility with others softwares</strong></td>
<td>Very Good</td>
<td>Very Good</td>
<td>Very Good</td>
<td>Very Good</td>
<td>Very Good</td>
</tr>
<tr>
<td><strong>Reuse of models and objects</strong></td>
<td>Good</td>
<td>Good</td>
<td>Very Good</td>
<td>Poor</td>
<td>Very Good</td>
</tr>
<tr>
<td><strong>Statistical Analysis</strong></td>
<td>Good</td>
<td>Average</td>
<td>Very Good</td>
<td>Good</td>
<td>Good</td>
</tr>
<tr>
<td><strong>Pre and Post Processing of Data</strong></td>
<td>Very Good</td>
<td>Very Good</td>
<td>Very Good</td>
<td>Very poor</td>
<td>Very poor</td>
</tr>
<tr>
<td><strong>Special Features</strong></td>
<td>Input Analyzer, Output Analyzer</td>
<td>Stat-fit, Output Result</td>
<td>DataFit package, Bottleneck analysis, Sankey Diagram (material flow analyses)</td>
<td>Button with direct connection to the Excel program to import data, Gantt Chart, Financial reports</td>
<td>Kinematic geometry to associate with machinery and transport systems</td>
</tr>
</tbody>
</table>

Considering the criterion of animation, the software Arena has two-dimensional representation and users must acquire a specific module to have the three-dimensional visualization. In the simulator Promodel the most common representation is also a two-dimensional, but according to forums the three-dimensional visualization can be configured and it is considerably more complex than the two-dimensional. The programs Flexsim, Plant Simulation and Quest have three-dimensional visualization. All programs have modules for optimization. In some software's such Arena, Plant Simulation and Promodel, the modules are coupled to data processing. Modules checking and tracking errors are common in all simulators discussed.

Devices for identifying bottlenecks and streams are offered by Plant Simulation. The software Quest has a module that provides kinematic motion of machinery and equipment making the visualization more realistic. Most simulators studied shows good compatibility with programs from Microsoft's Windows platform. The Active X technology, present in some simulators, allows the integration of models with data from spreadsheets, text files, and others format of files.
2.3 Difficulties to apply production simulation in shipyards

Despite some success stories for the application of production simulation in shipyards such as Flensburg and Meyerwerft, it seems that most of shipyards in the world are not using a day to day production simulation.

The quality of production simulation which requires accurate, reliable, repeatable and understandable results depends mainly on the quality of the input data. Moreover, the implementation of simulation models necessarily involves the manipulation of large amounts of data from both the ship and the manufacturing environment. It follows several types of problems, often very cumbersome, tedious and time consuming to solve. Here are presented the major problems that we may encounter:

- **Lack of available data** – Production time estimates usually make use of previously designed and constructed ships of similar type and size. Unfortunately, access to relevant data may be hindered by proprietary restrictions, interdepartmental communication gaps, etc. In certain cases, when the shipyard is entering a new market, no relevant data is available.

- **Insufficient data definition** – Fields of Data Bases (DB) are rarely defined precisely enough to permit direct comparison of data from different studies or reformulations to suit overlapping or otherwise differing definitional boundaries, Koenig (2002).

- **Inconvenient data format** – The data may be in a hard copy and not in an electronic DB, necessitating time-consuming keypunching. The data hierarchy may be unfamiliar, and require resorting a hierarchy appropriate to the present project. The constant evolution of the production tool implies a constant evolution of the data structure so that it is very difficult to compare past data with actual projects.

- **Unknown validity of data** – The data may itself be an estimate and not a record of the actual value. The value may have changed because a supplier has gone out of business or has moved out of country. The year in which the data was generated may not be known, so that the impact of inflation is in question. The data may be provided by a consultant, and its validity unknown.

- **Inaccessibility of data** – Object oriented databases are often used inside CAD/CAM tools such as TRIBON, CATIA, etc. in order to store the ship models. For design purposes it is the most appropriate solution because of the hierarchical structure of the product (eg. Scheme→Plates→Holes). Nevertheless, for general-purpose queries on the same information, pointer-based techniques will tend to be slower and more difficult to formulate than relational databases. In fact there is an intrinsic tension between the object encapsulation, which hides data and makes it available only through the interface methods of the CAD/CAM tool. So that, the extraction of the data from the CAD/CAM tools to a traditional relational database is required for production simulation purposes. It is a time consuming task mainly because the extraction of the data induces the implementation of macro using the specific export modules of the design tools.

- **Quality of the data** – It is clear that the effectiveness of any simulation ultimately comes down to the quality of the data used in the simulation. While more advanced simulation models are being developed, in our experience, the collection of data to drive these models often seems poor. Furthermore, the more advanced models require richer data sets.

- **High quantity of data** – The design and production description of a ship contains an amazing quantity of information. For example, it is a question of several million of rows with several hundred attributes for the steel structure description of a passenger vessel of 300 meters in length (eg. about 200000 steel items). It is much more when you are considering the outfitting.

- **Data integrity** – During the management of data it is very important to have the assurance that data are consistent, correct and complete. It may happen that this is not the case so it is necessary to report these data as outliers.

- **Data temporal heterogeneity** – The continual changes in technological processes and business practices invalidate the data of previous ships.
Beside the problems involving the data management, the following other issues can also explain why the majority of the shipyards don’t yet use production simulation to manage their planning's:

- The cost of some production simulation software is really a drag for some shipyards and especially for small and medium shipyards.
- Despite the last development of simulation toolkits for shipyards, modelling time remain a cumbersome task when the shipyard is implementing production simulation for the first time. It is realistic to tell that it can reach 6 months to 1.5 years for a complex full shipyards model.
- High skilled people are required to develop and maintain the simulation models up-to-dates with the technical evolution of the shipyard. It required a big investment from the shipyard.
- It is still difficult to integrate the simulation models with the CAD/CAM systems (DB management). There is not yet efficient commercial interfaces to share the data between the CAD/CAM software's and the production simulation software's.

Nowadays the Brazilian shipbuilding industry has Greenfield shipyards. The use of simulation models, in this case, can provide vital information to optimize investments in facilities, resources and equipments. The study of processes and systems by Discrete Event Simulation tool can promote practices more efficient to achieve the long-term strategies of the new shipyards and to ensure their competitiveness and responsiveness. Following the author opinion, Brazil is thus the ideal country to convince the shipyard to use DES system everyday.

2.4 Potential use of simulation in the Brazilian shipbuilding

According to Lloyd’s Register, in 1980 Brazil was the world’s 2nd largest shipbuilding nation behind Japan. However, the industry collapsed in the following decades due to local economic factors such as hyperinflation, high interest rates and the ending of state subsidies. By 1999, no ships over 100 tons were being built and the industry had shrunk to only 2000 workers nationwide, Paschoa (2010).

- Alusa Galvão Shipyard (PE)
  - fabrication of offshore drilling units (ships and platforms semisubmersíveis - drilling units), offshore support vessels (Supply Boats) and modules for oil platforms (topside modules).
- EISA Alagoas Shipyard (AL)
  - Besides ships, the shipyard will produce naval platforms and will do repair services.
- Enseada de Paraguaçu Shipyard (BA)
  - The construction of the drillship for Petrobras and topside assembly units manning the decks, are among the main interests of investors.
- Jurong Shipyard (ES)
  - Converting the hull of a tanker to the platform P-62, which will operate in the Roncador field, Campos Basin, with capacity to produce 180 thousand barrels per day from 2013
- Inhaúma Shipyard (RJ)
  - Conversion of ships into FPSO (Floating Production and Storage), now held abroad. Still, will serve as a support base for ferries owned by Petrobras, and use the area to support various operations.
- OSX Shipyard (RJ)
  - FPSO, TLWP, Fixed Platforms and Drillships
- Promar Shipyard (PE)
  - LNG
- Antlantico Sul Shipyard (PE)
  - Cargo ships - tankers, bulk carriers, ore and general cargo - as well as offshore platforms, drilling rigs and vessels for the oil and gas industry.
- Engevix Shipyard (RG)
  - N/A
However, an amazing revival has occurred in the last decade in response to large deepwater offshore oil and gas discoveries. For political reasons, the Brazilian Government through its state-sponsored oil company Petrobras and its shipping subsidiary Transpetro have used these oil discoveries as a vehicle for job creation. Wherever possible the Brazilian government has required as many of the requisite vessels and oil rigs to be built within the country. This has resulted in a shipbuilding boom. Today, the industry has a national workforce of over 45 000 with approximately 80 booked orders for a variety of ships and rigs, França (2009) and Paschoa (2010).

Fig. 1 shows the different Greenfields shipyards of Brazil. There are recently constructed or in plan to be constructed soon. The Brazilian shipyards can be divided into:

- Shipyards under planning (Greenfield) or construction and shipyards that are making retrofitting or extension of existing workshops – Layout planning
- Shipyards in operation – Production planning

2.4.1 Layout planning

The simulation for layout planning facilities can improve the evaluating of investments and of long-term strategies. Fig. 2 shows the layout of one Brazilian shipyard under planning.

One of the most important advantages of simulation of steel processing shops is the possibility to test different equipment, different suppliers and accounting costs (acquisition, installation, etc.). Different processes (automatic, semi-automatic or manual) can be studied and lines can be integrated (cutting and fabrication of flat panels, for example), reducing costs and integration time.

![Fig. 2: Example of shipyard under planning (Atlantico Sul Shipyard – PE/Brazil)](image)

Testing different positions of machinery and material flow allows the definition of a configuration that minimizes the distances and movements before the machines are installed. After the installation of certain equipment, the repositioning could be infeasible. The simulation permits analyze inventory levels, avoiding stops of production. The assembly blocks can be studied according of different strategies for building. Different methods can be investigated considering the inclusion of advanced outfitting.
Sharing resources such as gantries, cranes and trucks can also be checked. Productivity and time, considering different demands, can be estimated more accurately by providing greater support to managers. In pre-erection, large blocks of different sizes can be modelled. The physical space and resources can be defined depending on the size of blocks.

The workload in accordance with different types of vessels can be evaluated as the operational implications, such as proper inventory levels of intermediate products, and equipment parameters (speeds, etc.). The simulation of the erection could provide important information to determine the best strategy and choose the most appropriate resources. The simultaneous construction is another issue that could be addressed.

2.4.2 Production planning

![Example of shipyard in operation (Eisa S.A – RJ/Brazil)](image)

Unlike most applications in industries with series production, the main added value of the use of production simulation in shipbuilding is obtained in the support of the production planning and control, and not on the layout planning. Fig. 3 shows a layout of a Brazilian shipyard.

The existing shipyards need to constantly refine their processes and techniques to establish competitive conditions. These shipyards must adapt their operating strategies in order to achieve lower costs and production times. Transport systems for workshops can be tested under different parameters. For the steel processing process, different sequences and cutting planes can be evaluated, reducing the setup times of equipment and allowing a better use of resources.

The production of curved panels and sub-assemblies can be balanced, and different assembly methods can be studied. The sequences of production (daily or weekly) can be planned in order to optimize the
production. Any gaps between the planned schedule and the simulated schedule can be analysed and solved before that the real production take place. In the pre-erection and erection process, the constraints and conflicts between the transport systems can be predicted and the time of constructions can be estimated considering risks and uncertainties.

2.5 Coupling production simulation and optimization

Nowadays, more and more applications of simulations and optimisations are used in production planning to increase production performance and competitiveness of shipyards, Steinhauer et al. (2006), Kim et al. (2007), Souza et al. (2008) and Bentin et al. (2008).

In the context of production planning, the performances achieved with an overall production strategy can be assessed according to different criteria, such as lead time and manufacturing costs. The typical issues arise during the production are the balancing of working load and working force, the detection of bottlenecks and the maximization of resources utilization.

The production scheduling consists in establishing the best fabrication strategy (that can be represented by a production parameters system) in order to minimize both lead time and manufacturing costs. Those parameters can be quantitative, such as human resources or production facilities features, or qualitative, such as manufacturing sequence, workload dispatching on different working areas or priority strategies.

If the consequences of the variation of only one quantitative parameter on the production performances are relatively easy to foresee without the help of simulation, it becomes quickly much more complicated if several parameters are simultaneously modified. Optimisation based on production simulation models can be used to find one of the best set of values to minimize both lead time and manufacturing costs.

Production simulation coupled with optimisation tools used during the design stages can enhance the productivity of shipbuilding industry. Advantages are among others:

- New policies, production procedures, decision rules, production flows, organizational procedures, transportation systems, and so on, can be assessed without committing resources for their acquisition
- Hypotheses about how or why certain phenomena occur can be tested for feasibility before the production
- Insight of the interactions between production variables can be obtained
- Insight of the variables importance on the production performance can be obtained
- A production simulation study can help in understanding how the system operates rather than how individuals think the system operates
- The “What’s happen if” questions can be answered. This is particularly useful in the design of new production systems.
- We can do the evaluation of very complex systems where analytic solutions are not known and for which production simulation is the only possible approach
- Production simulation models often have a visual interface, sometimes with graphic animations and this fact makes them more reliable to the eyes of managers

3 Case study - Erection sequence optimization

3.1 Erection sequence issue

After the block splitting, the next scheduling stage to be performed is the definition of the optimal erection sequence. The erection process is a very complicated and highly networked operation involving decision-making interlinked with a lot of structural items. Manual solutions are often
The main issue of erection sequence is that this process follows a huge number of implicit physical and production rules. During the definition of block sequence, consideration must be given to:

- Physical constraints, such as some blocks are supposed to support other ones and have therefore to be positioned before.
- Planning and production control constraints, such as the desire for constancy of work inside the workshop.
- Block assembly constraints, such as the minimum time between the laying of blocks. This time is required in order to tack and weld the block on the ship. Another very restrictive constraint is that it is usually impossible to insert a block between two blocks already erected. Indeed, it would increase the complexity of block assembly stage. Moreover, the required gap necessary to insert the block is not compatible with the minimum welding gap.
- Erection constraints, such as the first’s blocks to be placed. The blocks contain the engines are often the first blocks to be placed because they require time for assembly and outfitting much higher than others.
- Erection strategies, such as the laying of ship blocks starting from the middle, fore or aft part of the ship; by layers or by slice or finally with a pyramidal strategy.

Each of these sets of constraints comes from a different constituency within shipyard, and the definition of block sequence has traditionally involved a process of iterative definition, review, and negotiation. Depending on the shipyard, this process may be well defined or somewhat inaccurate. Even when the process is well defined, it involves multiple channels and cycles of communication, and as a result it can be not only lengthy but also subject to errors and omissions that results in less than optimal block sequence.

The intent of this study is to examine how various computer-based analysis and simulation techniques might be used to improve the efficiency of the block sequence definition process.

![Fig. 4 : Workflow of the optimization process](image-url)
3.2 Presentation of the overall optimization workflow

Fig. 4 shows the overall workflow of the optimization process. Three different codes are involved in the optimization loop. Firstly an in-house development to generate feasible sequences, secondly a DES software (Quest - Delmia - Dassault) to evaluate the lead time of the production process, and finally an optimization platform (ModeFRONTIER).

3.2.1 The feasible sequence generator

The purpose of this module is to generate one/several feasible sequence according to the assembly technical requirements (production rules). The sequence is filled up with the blocks one by one. It is based on successive decision stages. The algorithm is launch recursively to choose the next bloc in the sequence.

The algorithm determines at each bloc selection step the neighbour blocs of the partial solution. Among them, the algorithm chooses only the blocks fulfilling the technical constraints. Finally, he can select heuristically one of the block providing a technical feasible sequence.

If we wanted to generate all possible sequence, there would factorial \( n \), where \( n \) is the number of blocks. One of the advantages of the technical constraints is the fact that they are extremely selective, and the number of feasible sequences decreases hugely.

The principle is the following. Blocks are selected one by one to be erected on the dry dock. As \( n \) blocks have to be erected, \( n \) decision steps have to be executed. At each step of the selection process, another block must be chosen among the blocks not already welded. For that purpose, a list of potential neighbours, which could be chosen as the next block in the sequence, because satisfying the technical conditions, is filled up, Fig. 5. Finally, a block is selected heuristically in the list of potential neighbours satisfying the technical conditions to be the next in the sequence.

This erection sequence generator has several advantages like the:

- Automation of the block erection sequences
- Very fast process (< 1s)
- Consider 3 different erection rules (by layer, by slice or by pyramid propagation)
- Generation of multiple feasible sequences with the same starting point (first block)
- Possibility to start with different initial blocks (or sequences)
- Possibility to add other production rules
- Input and Output text files
- Independent Java modules (Multi Platform)

Nevertheless some limitations are remaining. It seems very complex to take into account all production rules simultaneously during the construction of the erection sequence. It follows that some situations are not yet solved by the algorithm.
For each value of the input data (erection rule, first block to be placed), this in-house development can generate heuristically several feasible block erection sequence. Then the sequences are passed to the DES in order to evaluate the lead time of the production process.

3.2.2 The DES software

The Quest software has been selected by the authors to perform this study. The QUEST software (Queuing Event Simulation Tool) has been developed by Dassault Systems group. One of advantage of this simulation package is the possibility to reuse procedures, geometries of products and resources which are stored in a user library. Another point to be highlighted is that message boxes are showed, during the simulation run, containing indications of the element with problems and the type of error, helping the users to verify the model. The user can also modify or rewrite a selected behavioural rule and can even add new rules. This open architecture allows the user to control model behaviour at a very detailed level. In most cases, the behavioural rules defined by SCL (Quest Simulation Control Language) are sufficient. However, modifications can be made where ever necessary.

After the generation of one erection sequence the DES is launched in order to evaluate the lead time of the erection process. The study focuses the erection of a Suez Max Tanker as shown in Fig. 6.

![Production simulation of the block erection](image)

Fig. 6: Production simulation of the block erection

3.2.3 The optimization software

The ModeFRONTIER software has been selected for this study. ModeFRONTIER is a multi-objective optimization and design environment designed to couple CAE/CAD/CAM, CFD, simulation software to various optimization algorithm. It is developed by ESTECO Srl and provides an environment for product engineers and designers. This optimisation toolbox is a GUI driven software that wraps around the CAE tool, performing the optimization by modifying the value assigned to the input variables, and analyzing the outputs as they can be defined as objectives and/or constraints of the design problem.

The logic of the optimization loop has been set up in a graphical way, Fig. 7, building up the workflow structure presented in section 3.2 by means of interconnected nodes.

Lead time of the block erection process has been defined as objective function. Two design variables have been defined respectively the first erected block and the erection rule (by layer, by slice or by pyramidal propagation). No constraints have been added in the optimization workflow because the constraints are already included in the sequence generator algorithm. Simplex algorithm and a genetic algorithm have been tested during the optimization process.
3.3 First results

The first intermediate results of the case study are presented in this section. Fig. 8 shows the objective function after 350 iterations. It takes about 2 min for each simulation run i.e. about 12 h for 350 iterations. The reader would be easily noted that unfortunately no convergences are reached.

The authors are currently investigating the causes that of this results. The most probable source of this behaviour is that the feasible sequence generator module has a heuristic comportment that is too random for the optimization algorithm. The main advantage of the actual version of the model is that we are generating only the feasible sequence avoiding a lot of non feasible design during the optimization. The drawback is that the constraints are not implemented in the optimization software but in the feasible sequence generator module.

Despite the no convergence, an interesting outlook of this first analyse can be highlighted here. The block erection rule by layer takes about 6% more time than the other two erection strategies whatever the first erected block selected.
4. Conclusion and future work

The aim of this study is to improve the efficiency of the block sequence definition process and the optimality of the resulting block sequence regarding the lead time of the erection process. The selection of a right erection sequence seems to be a great potential to improve the manufacturing lead time. Nevertheless, the first intermediate results presented in this study are not yet valuable. A new approach is required to modify the modelling of the problem and/or the comportment of the optimization algorithm. Authors are in contact with the optimization software company in order to find an appropriate solution to solve this issue.

References


BENTIN, M.; SMIDT, F.; PARTH, S., (2008), *Process modelling and simulation using CAD data and PDM in early stage of shipbuilding project*, COMPIT’08, Liege

FRANÇA, B. (2009), *Brazilian shipbuilding revival*, Conf. Modern Shipbuilding, Ottawa


KIM, Y.S.; LEE, D.H., (February 2007) *A study on the construction of detail integrated scheduling system of shipbuilding process*, J. Society of Naval Architects of Korea, 44:1, 48-54


A Particle Swarm Algorithm-Based Optimisation for High-Strength Steel Structures

Sören Ehlers, Aalto University, Helsinki/Finland, soeren.ehlers@tkk.fi

Abstract

Crash resistant ship structures with high post-accidental residual strength are more and more important as a requirement of Goal Based Standards. A first step in the development of structures, which fulfil these requirements, is the rational identification of crashworthy structural concepts. Therefore, efficient optimisation algorithms are needed, besides reliable numerical collision simulations. Particle Swarm Optimisation (PSO) algorithms tend to improve the objective fast, which makes them a good choice for lengthy nonlinear finite element-based collision simulations. As an application of PSO and numerical collision simulations a Liquefied Natural Gas (LNG) tanker is optimised for crashworthiness. Furthermore, one concept using normal shipbuilding steel only and one concept using a combination of normal- and high strength steel is investigated. Hence, it will be shown if the benefits from high strength steel outweigh the increase in cost. Therefore, it will be shown if the combination of normal- and higher strength steel can result in increased crashworthiness, even though the high strength steel has a lower failure strain when compared with the normal steel.

1. Introduction

Collision accidents account for about 20 per cent of all serious accidents and thereby present a significant risk in the maritime transport environment. Hence, various efforts are undertaken to reduce this risk, e.g. active and passive safety measures. General risk analysis related to collisions of ships involves the assessment of the accident probability and the consequences to identify a satisfying equilibrium between safety measures and the stakeholders’ expenditure, Klanac and Varsta (2011). However, the feasibility of these safety measures can only be achieved if they are enforced by regulatory bodies or proven to cause a financial gain, respectively no additional capital investment, for the stakeholders. In this respect, it would be the most cost-effective solution to reduce the collision probability. However, the eventual unforeseen event will occur and, than it is up to the ship structure to withstand the deformation energy without breaching to avoid cargo outflow or water inflow. Hence, the scope of this article is to identify ship side structures with increased crashworthiness and to assess the influence of crashworthiness on the production and repair cost.

Crashworthy ship side structures have been identified on the basis of qualitative finite element simulations using novel crashworthy side structures, Klanac et al. (2005), Ehlers et al. (2007). However, so far only Royal Schelde, Graaf et al. (2004), has successfully implemented such novel side structures, so far into inland waterway ships and barges. They followed a procedure similar to the GL assessment procedure for alternative structures, Zhang et al. (2004). By this means, they simulated a collision incident in a quasi-static fashion with a rigid striking bow. A rigid bow results in the absorption of the available energy by the struck ship alone, even though Lehmann and Yu (1998) showed that the deformations of the striking bow absorb up to 42% of the available energy. However, with a rigid bow a comparison of different side structures can be made, as the energy absorbed until inner hull fracture is of primary interest for the conceptual design of the side structure. Usually, conventional side structures are not optimised for crashworthiness. Therefore, Ehlers (2010) presented a procedure to include non-linear finite element-based collision simulations in structural optimisation for the conceptual design phase using the novel material relation until failure according to Ehlers and Varsta (2009). A parametric finite element model was built that assigns the material relation and failure strain according to the element size. A particle swarm optimisation algorithm was used to identify the crashworthiness concept. Therein, he identified the optimum stiffener spacing and type for the most probably striking locations. As a result, Ehlers obtained a crashworthy tanker side structure absorbing 500% more energy when compared with the initial rule based structure at a
weight increase of 18%. Those optimisation-based procedures to obtain crashworthy ship side structures are the only procedures available in the literature at present. Therefore, Kõrgesaar and Ehlers (2010) extended this procedure to cover a realistic bow shape including the superstructure and utilized it to obtain a crashworthy side structure of a liquefied natural gas (LNG) tanker considering the deflection limit of the containment system. As a result, they obtain a crashworthy structure weighing no more than the initial rules-based concept, yet absorbing 80% more energy. Furthermore, they identified that the deflection limit of the LNG containment system represents the physical boundary of the structural deformation space. In other words, new concepts are needed to increase the crashworthiness even further, yet remaining within the limits of the overall structural deflections.

Therefore, this article investigates the possibility to utilize high strength steel (HSS) in crashworthy ship structures even though earlier investigations have shown no improvements in crashworthiness using HSS. Therein, it can be seen that a novel structural arrangement made from high tensile steel results in a full penetration of the side structure by the striking vessel. Hence, it is commonly concluded that the increase in tensile strength found in HSS cannot compensate for the lower fracture strain at failure. However, in this article it will be shown that a rational combination of normal steel and HSS can lead to an increase in overall crashworthiness. As an example, a crashworthy side structure of an LNG vessel will be presented and the results will be compared to the optimised normal steel concept presented by Kõrgesaar and Ehlers (2010). Furthermore, the influence of the structural arrangement on the repair cost will be assessed.

2. The particle swarm algorithm-based optimisation

A crashworthy conceptual design is a lightweight design that performs well in a collision scenario and fulfils the operational requirements. In other words, it is the design with the highest Energy per Mass (E/M) ratio, Ehlers (2010). This E/M ratio serves as a comparative unit for the optimisation procedure. Additionally, the conceptual design alternatives under one characteristic service loading condition are evaluated to comply with classification societies’ rules, for details on this assessment see Ehlers (2010). The ship motions are not considered in this collision analysis, as the maximum crashworthiness of the conceptual design is of interest. Furthermore, the production cost is assessed for the conceptual design alternatives, to account for the increased cost arising from the HSS utilisation. Additionally, the repair cost is obtained based on the simulated structural deformations. Hence, the objective of this optimisation is to identify the concept with the highest E/M ratio using HSS, yet minimizing the production and repair cost.

2.1 The collision scenario

The example scenario involves a striking and a struck ship colliding at a 90° angle. The striking LNG ship is assumed to be rigid and its bow collides with the struck ship of the same type. Furthermore, two principal striking positions are considered for the striking and struck ship dimensions and reasonable draft variation; for details see Kõrgesaar and Ehlers (2010). The studied LNG is a minimum ballast water concept with a profound V shaped bilge section, which will be in contact with
the striking bulbous bow. Furthermore, the relatively fragile containment system of the LNG ship has deflection limit of 4mm/m, which will be considered during the optimisation.

2.2 The optimisation procedure

A particle swarm optimisation (PSO) algorithm is chosen to identify the conceptual design alternatives. PSO is a population-based optimisation technique developed by Kennedy and Eberhart (1995). Optimisation starts from a feasible initial population and by updating the population of each generation the algorithm searches for the optimum. The best particle in current calculation round shares its location with all the other particles of the next round. Thereby the particles of the next round will be directed towards the best location of the previous best particle. In other words, PSO is a one-way sharing mechanism, which looks only for the best solution. Therefore, all particles tend to converge to the best solution. The PSO parameters for this study are given in Table I. The PSO algorithm is used as a tool to rationally increase the objective function value, even though a global optimum might not be reached at termination. This limitation arises from the collision simulation time of up to 30 minutes per collision simulation. Therefore, the optimisation is terminated at the prescribed maximum number of generations. The PSO algorithm is written in Matlab and is based on Jalkanen (2006).

Table I: PSO parameters

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Swarm size</td>
<td>50</td>
</tr>
<tr>
<td>Number of generations</td>
<td>50</td>
</tr>
<tr>
<td>Inertia at start</td>
<td>1.4</td>
</tr>
<tr>
<td>Dynamic inertia reduction factor</td>
<td>0.8</td>
</tr>
<tr>
<td>Number of rounds to improve solutions before the inertia is reduced</td>
<td>3</td>
</tr>
</tbody>
</table>

2.3 The crashworthiness simulation

The solver LS-DYNA version 971 is used for the collision simulations. The ANSYS parametric design language is used to build the finite element model for the LNG cross-sections with variable structural dimensions; see Table II and Ehlers et al. (2008). The three dimensional parametric model is built between two transverse bulkheads, Fig. 2, and the translational degrees of freedom are restricted at the plane of the bulkhead locations. The remaining edges are free. The structure is modelled using four noded, quadrilateral Belytschko-Lin-Tsay shell elements with five integration points through their thickness. The characteristic element-length in the contact region is 150 mm to account for the non-linear structural deformations, such as buckling and folding. Standard LS-DYNA hourglass control and automatic single surface contact (friction coefficient of 0.3) is used for the simulations, Hallquist (2007). The collision simulations are displacement-controlled. The rigid bow is moved into the ship side structure at a constant velocity of 10 m/s. This velocity is reasonably low so as not to cause inertia effects resulting from the ships’ masses, Konter et al. (2004).

Table II: Strake variables

<table>
<thead>
<tr>
<th>Thickness [mm]</th>
<th>Stiffeners</th>
<th>Stiffener types</th>
<th>Stiffener and/or plate material</th>
</tr>
</thead>
<tbody>
<tr>
<td>10...29</td>
<td>1...17</td>
<td>HP100x6, HP120x8, HP140x8, HP160x8, HP180x10, HP200x10, HP220x10, HP240x10, HP260x12, HP280x12, HP300x12, HP320x13, HP340x14, HP370x13, HP400x16, HP430x15</td>
<td>Normal steel or HSS</td>
</tr>
</tbody>
</table>
2.4 Material modelling

The collision simulation uses the element length-dependent local strain and stress relation until fracture according to *Ehlers and Varsta* (2009) for normal steel, Fig. 3 (left). This element length-dependent material relationship is identified on the basis of optical measurements. The HSS material relation is assumed to follow the same behaviour and is therefore shifted upwards as a result of the difference in yield strength. Furthermore, the reduction in failure strain found in the HSS material is experimentally determined for a dog-bone specimen with a gauge length of 60 mm. The resulting curve, for the full range of element lengths, is assumed to follow the same trend as found for the normal steel, Fig. 3 (right). The failure strain and element length relation is implemented in the ANSYS parametric design language model generation via material 24 of LS-DYNA, which allows failing elements to be removed at the critical strain. The constant strain failure criterion is justified due to the close ranges of triaxiality at failure, *Ehlers* (2010).

![Strain vs stress relation and failure strain vs element length](image)

**Fig. 3:** Strain vs stress relation (left) and failure strain vs element length (right) for normal steel and HSS

2.5 Production- and repair cost

The steel structure production cost of each alternative is calculated with a cost module according to *Rigo* (2001, 2003). The cost is based on a simplified calculation of labour and material costs. The calculated cost is calibrated referring to the cost of a stiffened panel using unitary production costs of the yard. The production cost is calculated as a sum of three components:

\[
\text{Cost}_{\text{total}} = \sum_{\text{panel}} \left( \text{cost}_{\text{material}} + \text{cost}_{\text{consumables}} + \text{cost}_{\text{labour}} \right) 
\]

(1)

Material cost includes raw material cost for the plate and stiffeners. The HSS is assumed to increase the material cost by 14%. Cost of consumables consists of the costs from welding (energy, gas, electrodes and provision for equipment depreciation). Labour cost is based on the workload for surface preparation and welding.
As consequences of the LNG ship collision, the resulting cost to repair the deformed and ruptured side structure is assessed. The steel replacement in a shipyard is assessed as a unit price of steel processed (USD/kg) for flat plates with correction factors for more laborious areas (Romanoff et al. 2007). Typical repair prices for per kg steel are given in Table III for quantities of 60 kg to 20 t. Laborious areas are subjected to certain correction coefficients ($C_{corr}$), Table IV. These steel processing prices include material, workmanship, lighting and ventilation, but exclude staging, tank cleaning, testing the tanks and access work. Coating is also a separate job.

<table>
<thead>
<tr>
<th>Country</th>
<th>USD/kg</th>
</tr>
</thead>
<tbody>
<tr>
<td>China</td>
<td>2.0</td>
</tr>
<tr>
<td>Turkey</td>
<td>3.6</td>
</tr>
<tr>
<td>Greece</td>
<td>10.8</td>
</tr>
</tbody>
</table>

Table III: Typical repair price per kg normal steel

<table>
<thead>
<tr>
<th>Laborious areas</th>
<th>$C_{corr}$</th>
</tr>
</thead>
<tbody>
<tr>
<td>Fore Peak, aft Peak and Ballast Tanks</td>
<td>1.15</td>
</tr>
<tr>
<td>Cargo Tanks and Engine rooms</td>
<td>1.15</td>
</tr>
<tr>
<td>Single curves plates</td>
<td>1.15</td>
</tr>
<tr>
<td>Holland profile</td>
<td>1.25</td>
</tr>
<tr>
<td>Oil and oily tanks</td>
<td>1.20</td>
</tr>
<tr>
<td>Plate thickness less then 10mm and more than 10mm</td>
<td>1.15</td>
</tr>
<tr>
<td>High Tensile Steel</td>
<td>1.14</td>
</tr>
</tbody>
</table>

Table IV: Correction coefficients for laborious repair areas

The repair cost for the different collisions scenarios can be calculated for the obtained deformed ship side structure using the average price from Table III and the correction factors from Table IV according to

$$C = \sum_{\text{repair areas}} C \cdot C_{corr} + C_{staging} + C_{coating} + C_{downtime} + C_{dry_dock}$$  \hspace{1cm} (2)

The material is considered deformed if plastic strains occur within the corresponding finite element. Furthermore, it is assumed that 25% of additional steel needs to be replaced during the repair. The final repair cost does not include the downtime and dry-docking costs.

3. Results of the optimisation

The progression of weight, cost, energy-per-mass ratio and the repair cost are shown in Fig. 4. The results are normalized with the rule-based concept according to Kõrgesaar and Ehlers (2010). The weight of the structural alternatives was increased continuously within a 5% band. The production cost tends to increase with increasing crashworthiness, however, at a significantly slower rate when compared with the energy-per-mass ratio. The repair cost shows a continuous decrease with increasing crashworthiness. The latter becomes possible due to the utilization of HSS, which in combination with normal steel can absorb more energy for a smaller extent of damage. Furthermore, 2500 iterations did not result in a very good convergence, thus implying that a better solution could still be found. However, the total optimisation time of 31 days, together with the increased crashworthiness, are considered sufficient to indicate the benefit from HSS in crashworthy ship structures. Hence, the resulting concept with the highest energy-per-mass ratio is shown in Fig. 5. HSS is positioned around the contact region. This arrangement supports the highly deformed contact region and minimizes the overall deflection of the inner hull, which is bound by the deflection limit of the containment system. For comparison, the normal steel concepts obtained by Kõrgesaar and Ehlers (2010) are shown in Fig. 6 and compared with the resulting HSS concept in Table V. This comparison shows that the large extent of damage found in the ENERGY concept results in the highest repair cost. The repair cost rise, because the increase in energy using normal steel can only be achieved by a spread of damage involving additional structural members in the absorption of energy. The HSS concept however, is able to absorb additional 13% of energy at significantly reduced repair cost, Table V.
Fig. 4: Progression of weight (top left), cost (top right), Energy/Mass (bottom left) and repair cost (bottom right) during the optimisation.

Fig. 5: Resulting crashworthy concept using HSS (marked with *) and normal steel.

Table V: Comparison of HSS concept with normal steel concepts

<table>
<thead>
<tr>
<th>Concept</th>
<th>Mass</th>
<th>E/M</th>
<th>Cost</th>
<th>Repair Cost</th>
</tr>
</thead>
<tbody>
<tr>
<td>Rule based</td>
<td>63.22 t/m</td>
<td>900 J/t</td>
<td>1.33M USD</td>
<td>0.415M USD</td>
</tr>
<tr>
<td>ENERGY</td>
<td>-4%</td>
<td>+69%</td>
<td>+56%</td>
<td>+43%</td>
</tr>
<tr>
<td>HSS</td>
<td>+3%</td>
<td>+82%</td>
<td>+5%</td>
<td>-10%</td>
</tr>
</tbody>
</table>
Fig. 6: Rule based concept (top) and crashworthy concept ENERGY (bottom) identified by Kõrgesaar and Ehlers (2010).
4. Summary and conclusion

This paper presented the optimisation for crashworthiness using a particle swarm algorithm and the nonlinear finite element method. As an example, an LNG vessel with a strict deflection limit of the inner hull, due to the containment system, was optimised for crashworthiness utilizing HSS besides the normal steel. For the normal steel concept it was confirmed that an increase in crashworthiness requires an extent of the damage. The latter is however conflicting with possible deformation restrictions, e.g. the containment system. However, due to the utilisation of HSS the crashworthiness can be increased, while at the same time the extent of damage is decreased, thus it is not conflicting with possible deformation restrictions.

Hence, the utilisation of HSS proved to be a valuable measure to increase the crashworthiness, especially if the possible deflections are limited. Additionally, the production cost is increased slightly, which is however recovered quickly if an accident occurs due to the significantly lower repair cost. As a result, it can be said that the presented optimisation results indicate the benefit from HSS in crashworthy ship structures.

References


EHLERS, S.; VARSTA, P. (2009), Strain and stress relation for non-linear finite element simulations, Thin-Walled Structures 47/11, pp.1203-1217


JALKANEN, J. (2006), Particle swarm optimization of load carrying structures. J. Structural Mechanics 2, pp.23-35


KONTER, A.; BROEKHUIJSEN, J.; VREDEVELDT, A. (2004), A quantitative assessment of the factors contributing to the accuracy of ship collision predictions with the finite element method. 3rd Int. Conf. Collision and Grounding of Ships, Japan, pp.17–26
KÖRGESAAR M.; EHLERS S. (2010), An assessment procedure of the crashworthiness of an LNG tanker side structure. Ship Technology Research 57/1, pp.50-64

LEHMANN, E.; YU, X. (1998), On ductile rupture criteria for structural tear in the case of ship collision and grounding, Practical Design of Ships and Mobile Units (PRADS), pp.149-156


ROMANOFF et al. (2007), Catalogue of the Selected Methods to Solve the Structural, Production and Operation Issues Affecting Design of 3 Application Cases, EU IMPROVE Deliverable 2.6

WOLF, M. (2003), Full scale collision experiment, X-type Sandwich side hull, EU Sandwich project report Deliverable TRD448

Abstract

Conventionally, the rooms and spaces of a ship are either modelled as volumetric entities, or with the aid of bulkheads and decks. According to our knowledge, no simple representation exists where both entities can be modelled independently, and where automated conversion from one view (volumetric) to the other (planes) is possible. This paper introduces a simple yet effective approach, where a ship designer can mix the use of volumes and planes in any fashion. Furthermore, this modelling method is applied in a novel tool to manage ship subdivision constraints. As quite some numerical constraints are known a priori, they can be defined in a list, and assigned to specific subdivision elements. Examples are bulkhead locations or required tank volumes or deck areas. A constraint management tool is developed which evaluates the ship layout design during the design process. The designer will be able to modify or add constraints and the tool will support the designer by managing these constraints during the design process. If the hull form changes, all submitted rules will be updated according to the new main particulars. If one of the constraints does not comply, an adjustment or alternative can be chosen at that moment and the impact of this change is directly visible. The designer can also ask the tool to provide a ship layout design that complies best with the constraints entered. When the Constraint Management program is used, a feasible ship compartment design can be made in a quick manner and the designer is kept from making errors. This means that a correct ship layout model is available on which probabilistic damage stability calculations and weight estimations can be performed in an early stage. This method has been implemented in a computer program, so actual design examples will be discussed.

1. Introduction

In the period 2008 – 2011, in the Netherlands a joint-industry research project, baptised Innovero (see http://www.marin.nl/web/JIPs-Networks/Public/INNOVERO-1.htm), was commenced which aimed at improving the ship design process, and thus decreasing the time required to produce a feasible ship design. Summarised, an infrastructure is developed in this project where a confederation of software agents, conducted by a knowledge management tool (in particular Quaestor, e.g. van Hees (2009), jointly tackle the various design tasks.

An important demand in Innovero is that the range of applicable software agents not necessarily has to be limited to dedicated naval architectural analysis software, but that also general purpose CAD-systems, such as Rhinoceros, Eagle or Autocad, should be considered. The consequences of this requirement are too wide to be discussed in full depth in a single conference paper; instead, there is one subject we will focus on, which is the representation and utilisation of the internal geometry of the ship. The reason is that conventional methods to represent the internal geometry (rooms, spaces, compartments, holds) have been developed in the past for naval architectural analysis tasks, such as the calculation of tank calibrations tables or damage stability. Because these representations are not necessarily the most suitable to be used in the design stage, e.g. with a general purpose CAD-system, they had to be reconsidered for Innovero purposes.

This reconsideration, and subsequently re-design, of representations for internal geometry is one of the two main subjects of this paper. The second one is the design and functionality of a tool that utilises this representation in order to guide the ship designer in complying with a number of geometrical constraints.
2. The ship’s internal shape representation

2.1 Design requirements for the internal shape representation

In the Innovero project, an inventory of requirements for a representation method was made. Most of the requirements are more or less related to the implementation, such as:

- Compatible with (or convertible into a format compatible with) the applied naval architectural analysis software, which is PIAS, by company SARC, of the Netherlands. If representation conversion is required, it should not lead to over-detailed models, because they could hamper calculation efficiency in case of lengthy calculations, such as probabilistic damage stability.
- Ready to be applied in combination with various ship hull representations, e.g. a surface model, a solid model, or a wireframe model. The latter may even be rather sparse, if the hull is only defined upon cross sections.
- The applied method, and its underlying entities, should be sufficiently easy to understand, and comprehensive, in order to be utilised with macros or scripts of general-purpose CAD software.

However, the most important requirement was not related to software as such, but was included in order to assist the ship designer as much as possible. The issue is that for some design or definition tasks it is pleasant, for a human being, if the focus is on spaces, while for other tasks a plane-based focus is more desired. And with ‘focus’ the basic modelling entity is addressed, so, summarised, one time it is easier to model a compartment directly by its boundaries, and in other cases modelling by means of bulkheads and decks is more appropriate. Obviously, planes and spaces are interrelated, so we need to address their duality.

2.2 Methods for the representation of internal geometry

Representation methods, as applied in today’s popular naval architectural software tools, have been investigated in Lee et al. (2003, 2009), where it is concluded that they fall apart in two categories:

- Those where spaces are defined by their boundaries. Such a method is not suitable for our purposes, because they lack an explicit definition of, or reference to, realistic planes\(^1\), so the duality between spaces and planes is not addressed.
- Those where a wireframe model of compartment boundaries is applied. Although in this case some relationship between spaces and planes does exist, their reverse relationship does not, so the duality is not fully addressed.

As a solution, Lee et al. proposed a data structure based on a non-manifold solid model, Fig. 1. This class of solid models, manifold and non-manifold, are discussed into depth in e.g. Mäntylä (1988), but summarised, a non-manifold model can describe an object with only an outer boundary, while with a non-manifold representation also an object with an internal structure can be modelled. Such a ‘solid model’ may seem rather abstract, but in e.g. Koelman (2003) it is shown that practical applications may certainly be built upon them. Nevertheless, these representation methods are rather complex, where the non-manifold method is an order of magnitude more complex than the manifold version. Because simplicity was expressed in sub-section 2.1 as an important requirement, this category of methods was considered to be less suitable for our problem.

Another interesting paper is Alonso (2008), where many implementation details are unfortunately not discussed, but from which it is apparent that the described software system allows for multiple repre-

\(^1\) By a realistic plane we mean a plane, or part of it, that does really exist in a ship. It will be obvious that such a plane does not necessarily has to extend over the full intersection between the plane and the ship hull.
sentation methods. The conceptually simplest method is a compartment bounded by six planes\(^2\), but there is also another method based on, quote, a ‘successive split of an initial space’, with a further detailing: ‘from the first level compartment definition the system allows the iterative subdivision of the created compartments by user-defined planes. This process can be repeated as many times as needed to obtain the desired detail of subdivision’. Certainly an interesting idea.

2.3 The BSP method

Starting from the most important design requirement of sub-section 2.1, the capability to tackle the duality between spaces and planes, we may conclude that the spaces part can be fulfilled by conventional methods. However, for the modelling of planes, realistic planes, the question arises what is the handiest method to represent and define those planes. And with handiest we mean which method intrinsically fits to the ‘logic’, the intuition and the expectation of the average ship designer. In discussions with ship designers and a designer’s panel, held back in 2006, it became apparent that the ‘space splitting idea’ is considered to be rather intuitive; with this method an empty hull form is split in two by a plane, those two resulting spaces are subsequently split in two by other planes etc. etc., until the subdivision is obtained. If we look beyond its particular fields of implementation, this ‘space splitting idea’ is similar to the well-known Binary Space Partitioning (BSP) method, where a space is recursively split in two, resulting in closed cells, in which we see the ship’s compartments. The BSP method (see [http://en.wikipedia.org/wiki/Binary_space_partitioning](http://en.wikipedia.org/wiki/Binary_space_partitioning) for an introduction) originates from interactive computer games\(^3\), but has also been used for modelling purposes.

\(^2\) Quote: ‘In most of the cases, the compartments are limited by six surfaces oriented according to the principal geographical directions of the ship. The aft and the fore compartment limits on the x axis, the port and starboard limits on the y axis and the lower and upper limits on the z axis’

\(^3\) In particular boys games, such as shooter games and football games.
Conventionally, the recursive subdivision is represented by a binary tree, which is also a suitable internal representation in a computer program. An example of a BSP-application in a plane is given in Fig. 2, where the shaded 2D figure on the right is recursively split by the planes $a$ through $f$ (which form the nodes of the tree at the right side of the figure) and the cells 1 through 7 (which form the leaves of the tree).

Examples, and other applications of the BSP, can be found in, amongst others:

- General descriptions of the BSP, its properties and a number of basic algorithms are presented in e.g. Thibault (1987), de Berg et al. (1998), Schneider and Eberly (2003), Naylor (1998), van den Bergen (1999).
- Conversion of BSP to a B-rep solid model is proposed in Buchele and Roles (2001), for objects with curved boundaries, and in Thibault and Naylor (1987) and Comba and Naylor (1996) for polyhedra.
- Boolean operations with BSP’s in Thibault and Naylor (1987) and Naylor et al. (1990).

2.4 The BSP method, applied to internal ship modelling

We have concluded that the BSP approach may fit the requirements of sub-section 2.1; it is capable of representing planes as well as volumes, supports the intuitive method of ‘space splitting’, is conceptually simple to understand, while the referred references of the previous sub-section provide sufficient tools to convert from and to other representations. By the way, the phrases of Alonso (2008), as cited in sub-section 2.2, suggest that their method is somewhat similar to the BSP approach, which gives the comforting feeling that we are not the only ones sailing the uncharted waters of the application of a ‘space splitting’ method in ship compartmentation.

However, a native BSP representation would not always be the best entity to present the ship layout to the ship designer, for the reason that in a BSP tree a compartment or plane may be subdivided in many smaller sub-compartments or sub-planes, which hampers the grand overview of the design. For that reason, a data structure was designed where the program user, the ship designer, is working with the following familiar entities:

![Fig. 2: Two-dimensional example of the BSP tree; from Comba and Naylor (1996)](image-url)
The ‘compartment’, which is an enclosed space within the ship.

The ‘physical plane’, which is a realistic plane within the ship, so a bulkhead or a deck. The physical plane may be bounded, which means it does not extend over the entire space in the ship hull.

The reference plane, which is a virtual and unbounded plane, only intended to speed up modelling and modification action.

The BSP forms the glue between those three entities, and is not available to the program user as a separate entity. This data structure is depicted in some more detail in Fig. 3.

2.5 The computer program

Based on the described data structure and tools a computer program was produced. The program, which is a module of the well-known PIAS suite, is characterised by the following properties:

- If it has two faces; it can either work as a stand-alone design tool with a Graphical User Interface (GUI), or it can act as a server, to serve other processes (or other agents). In the latter fashion it can e.g. provide properties of the internal ship entities, such as the shape of decks or bulkheads, or the tank volumes and moments of inertia.
- It has multiple ways to communicate with other software, which is either the knowledge management system as applied in Innovero, or other end-user software, such as a general-purpose CAD system. Currently, implemented communication methods are file, named pipes and TCP/IP.
- Includes inter-process communication by means of XML.
- Support for export of the ship compartmentation, e.g. in 3D VRML file format, or as a user-defined 2D layout plain DXF, which can serve as an sub-layer or a general arrangement plan or a tank arrangement plan.
- Has multiple import and export options, notably from and to other PIAS modules, such as those for intact or damage stability.
- Includes tools for the management of design constraints. This is the subject of the next section.

In Fig. 5, an example of the GUI is presented. It consists of several dedicated windows; one presents a 3D overview of the hull and its internals, three windows present three orthogonal sections, and there are windows with a list of physical planes, reference planes and a compartment tree. It is interesting to see that the BSP is not included in this GUI, because it is irrelevant for the daily use of the program. However, for debugging purposes, the BSP can be visualised of which Fig. 4 gives an example.
3. Constraint Management

3.1 Introduction

Ship designers have to deal with numerous types of rules and regulations imposed by classification societies and regulatory authorities. These rules prescribe guidelines for the design of a ship in general. These rules generally dictate requirements for the positioning of bulkheads and compartments in the total ship layout, (damage) stability, and freeboard. With many different, mostly empirical, formulas the exact boundaries of these rules are determined. Most of these rules can be found in different rule-books, text or via digital (static) index files.

Besides the regulatory rules as described above, the designer has to take owner requirements into account. These rules generally prescribe requirements for compartment volumes and dimensions, deck areas, speed, noise, building costs etc. of which some are known in advance in the form of a list of requirements and others are gradually specified during the design process.

It would be useful to develop a constraint management tool that collects the different types of design constraints. Besides, it must be able to check if the geometrical model complies with the submitted design constraints. If some of the constraints are not satisfied, feedback on the confliction region(s) must be given to the designer. Even more, a solution proposal should be provided which complies best with the submitted set of design constraints.

3.2 State of the art

In the shipbuilding industry a lot of research is performed on knowledge based design. However, to our knowledge, no program exists which meets the requirements as stated in paragraph 3.1. Some research is done on knowledge-based design in the maritime industry and other sectors; in Augusto and Kawano (1998) a ship structural design is optimised with a nonlinear search algorithm. The focus of this research lies in the optimisation of the hull structural parts. The way of defining the constraints in the form of a penalty function can be used in the same way for the constraint management problem.
In Yu et al. (2010) a new method for offshore platform design is presented. This method allows better reusability and changeability of the compartment layout during the design process by describing the layout model as a parametric design. The method uses a 2D arrangement as a basis and performs parametric changes in this 2D plane, deck and bulkhead heights must be entered externally and are not adjusted during the optimisation routine. The described method works well for simple compartment layouts, but cannot be used for ship compartment design where vertical subdivision is of equal importance as transverse and longitudinal subdivision. Apart from an arrangement evaluation, hydrostatic and FEM calculations are also considered in the described method. The general approach of this implementation may be used for the addition of other constraints like initial stability and trim requirements in the constraint management system.

In Lee and Lee (1997) an intelligent compartment design system is developed which supports compartment design of a crude oil tanker. The nature of the research shows many similarities with the constraint management problem. However, the whole constraint management structure focuses only on design rules of crude oil tankers and is therefore not able to handle general constraints. Mainly IF-THEN statements are used to implement the specific rules, which results in a program with pre-defined constraints that is not capable of handling other types of constraints.

### 3.3 Implementation

#### 3.3.1 General constraint description

In the ‘to be developed’ constraint management tool ideally all types of rules and requirements as mentioned in paragraph 3.1 are taken into account. Purely considered from the calculation-difficulty point of view this will not cause any problem. All evaluations deal with calculations that are not very complex to perform. However, there are some evaluations that nowadays still take a considerable time to calculate. E.g. the different evaluations and calculations to perform a probabilistic stability calculation are not complex, but all the different possibilities that need to be calculated make the calculation exhaustive. With evaluations that still take several hours to calculate, some of the design constraints will not be taken into account at the moment.

For the constraint management project we have restricted ourselves to address the constraint problem as far as it can be included in the design phase, in a processing time that allows interactivity. Implementing the calculation-intensive evaluations will result in long calculation times that will influence the workability in a negative way. For the time being, all constraints that deal with plane positions, areas and compartment volumes are considered only.

A way is found to describe the varying list of design constraints in a general way. It is found that the majority of the common constraints belong to one of the groups:

- amount (e.g. number of bulkheads),
- position (e.g. tanktop position),
- area (e.g. deck area),
- volume (e.g. fuel oil tank volume).

Besides belonging to a group, each constraint boundary can be of the type:

- minimum (at least…),
- maximum (at the most…),
- minmax (in-between … and …).

When the constraint group and type definition is completed, it is known which boundary values should be given to the constraint. Where the min and max type only require one (lower or upper) boundary, the minmax type requires two (lower and upper) boundaries. With the two properties
‘group’ and ‘type’ combined, the majority of constraints can be described in a general way. Some examples are given in Table I.

Table I: Example of constraints assigned to a group and type.

<table>
<thead>
<tr>
<th>Constraint</th>
<th>Group</th>
<th>Type</th>
<th>Min</th>
<th>Max</th>
</tr>
</thead>
<tbody>
<tr>
<td>Fuel oil tank volume at least 900 [m³]</td>
<td>Volume</td>
<td>Min</td>
<td>900</td>
<td>-</td>
</tr>
<tr>
<td>Collision BH between 88.2 [m] and 91.0 [m]</td>
<td>Position</td>
<td>MinMax</td>
<td>88.2</td>
<td>91.0</td>
</tr>
<tr>
<td>RoRo deck area less than 2200 [m²]</td>
<td>Area</td>
<td>Max</td>
<td>-</td>
<td>2200</td>
</tr>
<tr>
<td>At least four bulkheads</td>
<td>Amount</td>
<td>Min</td>
<td>4</td>
<td>-</td>
</tr>
</tbody>
</table>

Finally, the constraint boundary values must be calculated. As mentioned in sub-section 3.1, the constraints occur in all shapes and sizes. Usually they are given in empirical formulas, but also lookup tables and graphs are used to determine the constraints. To be able to implement all these types of constraints, a general mathematical formula editor is required. Instead of adding it to the constraint management tool itself, the strength of the existing formula worksheet MS Excel is used. Within Excel almost all types of mathematical equations can be entered. The use of look-up tables and logical rules (IF…THEN) is also a possibility. By using the Dynamic Data Exchange (DDE) communication channel, a link is made between the constraint management tool and an Excel worksheet. In this worksheet the ship main dimensions are imported and constraint formulas use these parameters to calculate the constraint values. The final outcome of the formula is sent back to the constraint management program as a lower or upper boundary of the constraint.

3.3.2 Linking the constraints to the geometric model

In our new program, a geometric model consists of planes which partition the ship hull space in different compartments (see paragraph 2.4). Horizontal planes will serve as decks, vertical planes as transverse or longitudinal bulkheads. An entity in the current ship layout design contains some general properties (name, number, weight group, etc.) and geometrical information (position, corner points, direction, etc.) of the model components. In addition to that, for each plane or compartment the connected constraints can be chosen from the list of design constraints.

3.4 Mathematical description of the model

With the link between the geometric model and the constraints, all required input is given to perform a model evaluation. In the following part the constraint management problem is described in a mathematical form based on a standard optimisation problem in conformity with the description as in Augusto and Kawano (1998). The standard optimisation algorithm consists of design variables, an objective function and the description of constraints with penalty functions. These last two are evaluated as a whole in the combined penalised objective function. The design variables serve as the input for the optimisation problem and are changed systematically to satisfy the main optimisation goal: minimising the value of the penalised objective function.

This general description of the optimisation problem is used as a basis for satisfying the constraints. Although the common goal of an optimisation problem is to minimise the objective function, the constraint management program of this paper uses it for the satisfaction of constraints. Therefore it is not an optimisation task, but a constraint satisfaction problem, with the objective function only occupying a secondary role.

3.4.1 Set of equations

The design variables specify the design and the goal is to find the variables for which the design is best. Written in the form of a vector, with n independent components, the design variables can be given as:

\[ X = \{x_1, x_2, x_3, \ldots, x_n\}, \quad (4.1) \]
For the constraint management program, the plane positions are the design variables. Every $x$ represents a plane position, and $n$ represents the number of planes that are taken into account. The plane positions determine the ship layout and by judging their position the quality of the compartment design is measured. The planes can be moved in their orthogonal direction only.

When the constraint management tool evaluates the compartment design, there already exists a ship layout. This layout is made by the designer, so this can be seen as the preferred compartment design, given that no other constraints apply. The main goal of the constraint management system is to find a solution that complies best with all the constraints, but which needs the minimum amount of plane movements; after all we do not want to annoy the designer. This last requirement directly determines our \textit{objective function}; while satisfying the constraints, minimise the plane position adjustments to match the layout design best with the designer’s intentions. The \textit{objective function} is described as:

$$f(x) = q \cdot \sum_{i=1}^{n} (x_{i, \text{old}} - x_{i, \text{new}})^{\delta}, \quad (4.2)$$

with $q, \delta$ as scale factors to control the influence of the object function on the penalised optimisation function.

Subsequently, the description of the constraints is described in the following part. Usually, design problems are subject to series of inequality constraints in the form $g(x) \geq 0$. In case of a maximum constraint $x \leq x_{\text{max}}$, this can be rewritten as:

$$g(x) = -x + x_{\text{max}} \geq 0, \quad (4.3)$$

which is the same format as the standard inequality constraint. In paragraph 3.3.1 the various types of constraints were discussed. All these constraints can be written in the standard form and are thereby described digitally.

The compliance of the different constraints is described in the form of a penalty function. As described in Augusto and Kawano (1998) there exist \textit{interior} and \textit{exterior} penalty functions. When ship design constraints are considered, usually a starting point in an unfeasible area is given. The goal is to reach a feasible area regardless where exactly within that area. In that case an \textit{exterior penalty function} is used since it is the only one capable of investigating an infeasible area. The \textit{exterior penalty function} used for the constraint management program is:

$$p(X, r) = r \cdot \sum_{i=1}^{k} \alpha_{i} \cdot \left( \min \{0, g_{i}(x)\} \right)^{\beta_{i}}, \quad (4.4)$$

$$\min \{0, g_{i}(x)\} = 0, \text{ if } g_{i}(x) \geq 0$$

$$= g_{i}(x), \text{ if } g_{i}(x) < 0 \quad (4.5)$$

The penalty value is zero in case the constraint [(4.3)] is satisfied. If not, the squared difference of the boundary and the actual value will be the penalty for that specific constraint.

Finally, the parts given in the foregoing paragraphs are combined in the \textit{penalised objective function} $\phi(x, r)$, given by:

$$\phi(x, r) = f(x) + p(X, r)$$

$$= q \cdot \sum_{i=1}^{n} (x_{i, \text{old}} - x_{i, \text{new}})^{\delta} + r \cdot \sum_{i=1}^{k} \alpha_{i} \cdot \left( \min \{0, g_{i}(x)\} \right)^{\beta_{i}}, \quad (4.6)$$

in which $f(X)$ stands for the \textit{objective function} and $p(X, r)$ for the \textit{penalty function}. The \textit{penalised objective function} describes the total performance of the compartment design. The lowest resulting value of this formula will represent the layout model that complies best with the evaluated constraints. With the $q, \delta, r$ terms, the influence of the different terms relative to each other can be controlled. The $\alpha_{i}, \beta_{i}$ terms determine the scaling of the different constraint penalties to each other only.
3.4.2 Constraint satisfaction

In order to find the minimum of the penalised objective function, a Quasi Newton method is used. The overall minimum of this function will indicate the combination of plane positions that satisfies the constraints best. The strength of approaching the constraint management program this way is that, regardless of the feasibility of the set of constraints, a solution (a minimum) will always be found. By inspection of the value of the penalised objective function it can be seen if the solution found is a feasible or an infeasible one.

In case of an infeasible compartment design, it is difficult to judge which action has to be taken. An infeasible design means that two or more constraints are mutually conflicting. Which of these constraints is the ‘problematic’ one cannot be judged by the constraint management tool itself. After all, the design constraints are all included by the designer in the first place; he is the one to decide what to do in the infeasible situation.

When the combination of constraints is infeasible, the $\alpha_i$ scaling factors in (4.6) highly influence the outcome. By changing the individual scaling factors the dominance of the constraints can be controlled. In order to give the designer control over the scaling factors, the constraint equaliser is introduced. In this window (additional to the main GUI) each constraint is represented by a trackbar. By moving this trackbar up or down, the constraint dominance is increased or decreased (see Figs. 7 and 8). If the designer is able to change the weighing of each individual constraint, the outcome of an infeasible constraint combination can be influenced and preference can be given to a specific constraint. In this way the interpretation of the unfeasible design is left to the designer, but the influence of the entire constraint system is given instantly.

![Fig 5: Graphical User Interface](image)
4. Application examples

In Fig. 5 the main GUI of the new compartment design program is displayed. Visible are the three orthogonal projections and one central 3D view on the ship. Both the tree of compartments (left) and a list of physical planes (right) are visible. The ship layout design of Fig. 6 was created within 15 min. In the colours blue, pink and green the orthogonal planes can be seen. Oblique planes are coloured yellow.

In Figs. 7 and 8 a practical example of the use of the constraint equaliser is displayed. The model consists of a simple compartment design, just for good understanding. Two bulkheads (coloured blue) are linked to the ‘bulkhead aft position’ and ‘bulkhead forward position’ constraints. The tanktop (green) is connected with the ‘tanktop position’ and ‘tanktop area’ constraints. The constraints and their scaling can be seen in the trackbar windows. A fact is that this combination of constraints is infeasible. In order to meet the tanktop area constraint, the forward and aft bulkhead (which bound the tanktop) need to be moved forward and backward respectively. However, their position constraint does not allow them to move further than a specific position, which is 35 [m] for the aft bulkhead and 65 [m] for the forward bulkhead. So moving the bulkheads to increase the area constraints violates the position constraints. The other way around, i.e. moving the bulkheads to their constraint-positions, will violate the ‘tanktop area’ constraint. By shifting the trackbars the dominance of the different constraints can be adjusted. In Fig. 7 the scaling factors are changed to satisfy the position constraints, in Fig. 8 the trackbars are moved to meet the ‘tanktop area’ criterion.
5. Conclusion and subjects for further research and development

The developed Binary Space Partitioning (BSP) method, as discussed in this paper, seems to work very well for designers in the early phase of the design process. The method requires minimum input by the designer. However, he or she has to think ahead to be able to use the right order by which the spaces are split into two adjacent new subspaces. More work has to be and can be done to further improve this new subdividing method. An additional feature would be to also be able to add volumes which can be subdivided later on. This way the superstructures and deckhouses can be included in the design. Also the possibility to re-use (parts of) subdivisions could help to speed up the design process. The topology within a particular class of ships is often very similar or even the same. Also the topology of parts of the design can be identical to existing designs. This way, a new ship could be generated by using pre-defined parametric “building blocks”. In this way, parametric models of alternative design options can be generated quickly and into more detail, providing also the possibility to analyse and optimise the designs when used as input for analysis tools.

The constraint management method as programmed and described in this paper is also a very interesting feature to be explored further. The program as it is right now can deal with the design constraints related to the positions, areas and volumes as defined by the BSP method. However, some valuable additions can be made like an initial stability and trim calculation. The constraint management method has only been used for one particular design topology. This method, however, also gives the designer the opportunity to create alternative topologies while the computer monitors the compliance with the constraints. In addition, a next step forward would be to combine this method with the option of also being able to vary the dimensions and other design parameters of the hull form. This way the effect these changes will have on layout, areas, volumes and capacities as well as on additional performance characteristics of the design like resistance, weight and costs can be investigated and included in the process of finding an optimal solution.

References


A Machine-Learning Approach to Predict Main Energy Consumption under Realistic Operational Conditions

Jóan Petur Petersen, DTU Informatics, Lyngby/Denmark, joanpeturpetersen@gmail.com
Daniel J. Jakobsen, Decision3, Tórshavn/Faroe Islands, dj@decision3.com
Ole Winther, DTU Informatics, Lyngby/Denmark, owi@imm.dtu.dk

Abstract

We present a novel and publicly available data set of high quality sensory data collected from a ferry over a period of 2 months, and investigate state of the art machine-learning methods for prediction of main propulsion fuel efficiency. Neural networks are applied in both instantaneous and predictive settings. Performance results for the instantaneous models and examples and discussions of the practical advantage of the predictive models are given. The presented models have been successfully deployed in a trim optimization application onboard one of DS NORDEN's product tankers.

1. Introduction

Most modern ships have several measurement devices that keep track of its operations, such as the vessel’s speed, fuel consumption, weather conditions, and so forth. Storing, analyzing, and acting upon this data could become a valuable asset for the ship owners, operators, and crew. In this article we present a freely available data set of data collected from a ferry, where the data has been used to develop the models used in a trim optimization application.

Already in 1987 the technical article “Marine Performance Surveillance with a Personal Computer” by Journée (1987) and Journée et al. (2003) described an on-line data collecting system and a mathematical model tuned by the collected data to predict ship performance. This mathematical model is based upon physics and hydrodynamics formulas. In the article by Leifson et al. (2008) model combinations of conventional physical models and artificial neural networks are tested. In the article by Pedersen and Larsen (2009) an Artificial Neural Network is used to predict the propulsion power. Both papers make use of collected operational data from the ship.

Ship resistance evaluation methods can be divided into four groups, from the traditional to the more advanced: Traditional and standard series methods, regression based methods, direct model tests, and computational fluid dynamics (CFD) Molland (2008). The traditional, standard series, and regression based methods typically rely on a set of parameters that describe the hull, for example the wetted area, the Froude number, and so on. A well-know regression based approach to predict speed/power is the Holltrop-Mennen method, Holltrop and Mennen (1982), Holtrop (1984), which is widely applied in the initial design stage. Here the total resistance of the ship is subdivided into additive components, which are estimated based on data collected from model and full-scale tests. Other full-scale and model experiments have been carried out to investigate how different components affect the total resistance of the ship, such as fouling and wind load Townsin (2003), Blendermann (1996). The Holltrop-Mennen methods have been applied for operational optimizations Leifson et al. (2008). These initial design methods may sacrifice details for the sake of fewer parameters, and more robust results. Because of this sacrifice, these methods might not be very well suited for analysis of the ship’s performance after it has been constructed. Computational Fluid Dynamics, CFD, can be used to estimate hull resistance Ruggiero et al. (2007), but these calculations are computationally demanding, and typically take several hours to complete. So it is infeasible to use them onboard directly for optimization applications. It is however possible to do the CFD calculations beforehand for a given set of parameters. This technique has been used successfully for an trim advisory system Hansen and Freund (2009).

In contrast to the physical models, statistical approaches only require knowledge about the ship and the physical setup when it comes to the design of what sensory features to use in the system. The
statistical model learns the relations between the measured signals from the collected training data. In the predictive machine learning approach the main agenda is generalization. This means the ability of the model to give sensible predictions for situations not identical to what has already been observed in the training data. The main factors determining the predictive performance are the relevance of the sensory features, the amount of training data available and the choice and tuning of the complexity of the model to the task at hand. Hybrid physical and statistical models with a proper assessment of the predictive uncertainty of each component may in principle improve over a pure model by suitably interpolating between the two Leifson et al. (2008).

In a system that can optimize the fuel consumption for propulsion, the two key quantities we need to be able to predict are the fuel consumption \( f \) and speed through water \( v \). Ignoring route planning concerns the fuel efficiency is defined as the speed through water divided by the energy (fuel) consumed: \( e = \frac{v}{f} \).

The ship’s state is the set of variables that allow us to predict \( f \) and \( v \). Depending on the application, a number of these quantities can be considered as control variables such as the trim, propeller pitch, RPM, i.e. these can be changed by the crew, while others like weather conditions or physical characteristics of the ship cannot. All relevant state variables are not necessarily measured - there can be several reasons for this, for example by choice, cost, or complexity. But arguably the most important factor in getting good model predictions is to have rich and accurate sensory data. An instantaneous model predicts the current value based upon current measured signals. It therefore does not treat the control variables differently than the remaining variables. Using the model directly as part of an advisory system for optimizing fuel efficiency on-line can be problematic, because a change in a control variable may affect several variables, including inputs to the model and not only the outputs of the model. For example changing the trim could affect the way the autopilot controls the rudder, as the dynamics of the ship are also affected, which would result in an incorrect estimate of the speed and fuel consumption. So a temporal state-space model, Ghahramani and Hinton (1996), for the dynamics of all variables is a more adequate model for this purpose. We will not consider a complete state-space model with hidden states, but instead predictive model, which uses a non-linear neural network to capture the assumed Markovian dynamics of the system, Weigend et al. (1990), Chakraborty et al. (1992), Svarer et al. (1993). This so-called tapped-delay neural network is not explicitly defined in terms of a model with hidden states like the traditional state-space model, but can never the less capture the essential part of the dynamics of the system. Ideally, the tapped-delay model represents the deterministic part of the dynamics. With careful regularization in order to avoid overfitting, we may also use the residual error of the model on the training to fit an additive term in the model to represent the stochastic part of the model due to noise and/or un-observed information. The full model consisting of the deterministic and stochastic part allows us to make efficient non-linear propagation of predictions and uncertainty through time. The instantaneous model and the tapped-delay neural network, which we will focus on in this article, may be considered as an important first step towards a complete state-space model since it actually models \( v \) and \( f \), which in the complete state space view are just two of the state variables.

2. Data collection

The data is collected from a domestic ferry, M/S Smyril in the Faroe Islands, owned by Strandfaraskip Landsins. The ship serves a daily route from the capital Tórshavn to the southernmost island Suduroy. The ferry sails two to three trips back and forth each day, where the duration of each trip is 1 h 55 min. The ship is designed by Knud E. Hansen, and built on the IZAR shipyard, San Fernando, Spain (delivered in 2005). Fig. 1 gives some data on the vessel.

A computer system and some additional hardware is installed onboard the ferry. This system collects the data from the ship, which the models will be based upon. The data is made publicly available, and can be found here: http://cogsys.imm.dtu.dk/propulsionmodelling/.
Fig. 1: Information about the vessel Smyril, and the collected trips in the data set.

The data spans a period of almost two months, February 16th to April 12th, 2010. The map in Fig. 1 shows the routes taken by the vessel during the data collection period. We have made this data available in an attempt to encourage benchmarking within the ship propulsion field. Decision3 has provided the data for the project. To our knowledge, this is the first data set of its type that has been made publicly available. By releasing it to the community, we hope that it will be used for benchmarking similar models, and further work within this field.

The following signals are stored by the system: Port and starboard propeller pitch, port and starboard rudder angle, port and starboard level measurements, fuel density, fuel temperature, fuel volume rate, trim angle, longitude, latitude, speed through water, speed over ground, true heading, wind speed, and wind angle. Each signal is stored in a Comma Separated Values (CSV) file. For more details please refer to the homepage. All of the signals supplied in the data set are common onboard most ships. However the microwave based level measurements devices are quite novel, though they have been used before Atwater (1990). With these devices it is possible to get information about the squat, trim, heeling, and draft of the ship, the waves generated by the motions of the ship, and sea waves around the ship. Much more work could be done interpreting these signals.

The system was setup to start storing data from the point when the ship starts to move and until it stops. Subsequently this might have been too restrictive, as the information just before and after a trip could also contain valuable information, which has not been stored. Smyril has a shaft generator, but it has not been used during the period of data collection. The engine has a constant RPM, and it has therefore not been measured.

If we look at the map in Fig. 1, we see that there are two short trips, where Smyril sails northward from Tórshavn. Here the ship is bunkering heavy fuel oil. The bunkering can be seen on the vessels draft. The variation in the routes taken is due to weather conditions.
3. Methods

3.1 Instantaneous/regression model

We use supervised learning methods for building a instantaneous/regression model to estimate the fuel consumption and speed from a set of measured features, Bishop (2007). Formally the response (or output) variable, \( y \), e.g. the fuel consumption, is modeled as

\[
y = f(x; \theta) + \epsilon,
\]

where \( f \) denotes the model function depending upon parameters \( \theta \) and explanatory (or input) variables \( x \). The residual \( \epsilon \) is the part of the measured signal \( y \) not explained by the model due to noisy measurements and/or model shortcomings. The parameters \( \theta \) should be learned (inferred) from a training set \( \{ (x_n, y_n) \} \) of input output pairs. We will use a probabilistic formulation assuming that the residual has a Normal (or Gaussian) zero-mean and \( \sigma^2 \) variance distribution: \( \text{Norm}(0, \sigma^2) \). We will assume that measurements are independently identically distributed (iid) such that we can write the likelihood as

\[
p(y|X, \theta, \sigma^2) = \prod_{n=1}^{N} \text{Norm}(y_n|f(x_n; \theta), \sigma^2),
\]

where \( y \) and \( X \) are shorthand for the vector of output and matrix of inputs, respectively. We will use an Artificial Neural Network (ANN), Bishop (2007), a non-linear model for \( f \). We briefly review this method below including details for our use of the method, Bishop (2007) for detailed general description. We take a predictive rather than descriptive modeling approach reporting the test (or generalization) performance on a test set. In all our experiments with the instantaneous model we use one third of the data as test set and the remaining two-thirds as training set. We investigate two different ways to make the training-test split. In the first approach test data is selected at random and in the second we minimize the training-test redundancy (or cross talk) coming from adjacent time-windows using complete legs as test and training data units, see Section 4.1 for details.

3.2 Tapped-delay neural network

While the instantaneous model only predicts the current target value, the tapped-delay neural network setup can predict future values.

A tapped-delay neural network model can be described as

\[
Y_n = f(X_n, X_{n-1}, \ldots, X_{n-d+1}, w) + e(n,w),
\]

where \( f \) is in this case a artificial neural network that maps the seen sequence of sample vectors, \( X_n \), into a predicted future sample, \( w \) are the model’s parameters often also called weights, and \( e \) is the error in the prediction. The error is only considered to be additive here. The number of previous samples vectors available to the network or steps is given by \( d \).

We will make some small modifications to the tapped-delay neural network, as to adapt it more to this application. The output of the network is set to predict the next ship state, so that we want the output of the network to be \( Y_n = X_{n+1} \). The controls are added as inputs to the model. The controls are considered to be known, so these are not predicted by the model. The controls are represented by a vector \( U_n \). The variable \( X_n \) will express the dynamic state of the ship. It will contain the ship’s
speed through water, trim, draft, and so forth. The value from the current step, \( X_n \), is added to the right-hand side of Eq.(3). The argument for doing this, is that we suspect the next step, \( X_{n+1} \), to be quite similar to \( X_n \), and therefore the model will only have to learn the difference between the steps. We will concentrate on the simplest tapped-delay neural network first, where \( d = 1 \). Applying these modification to Eq.(3) we obtain a new model given by

\[
X_{n+1} = f(X_n, U_n, w) + X_n + e(n, w) \tag{4}
\]

### 3.3 Data pre-processing

The samples are arranged into intervals or windows. The windows are non-overlapping and there is no gap between consecutive windows on a trip. If measurements are missing completely from a device in a window, the window is discarded. A device might have malfunctioned or stopped providing data for some other reason. Based on the samples within a window, a number of features are generated. The feature extraction process consists mostly of simple mathematical operations such as taking the mean, variance, or derivative of an input. The mean, variance, and derivative features are calculated as

\[
\begin{align*}
\Omega_{\text{mean}}(w) &= \frac{1}{M} \sum_{n=0}^{M} x_n, \\
\Omega_{\text{var}}(w) &= \frac{1}{M} \sum_{n=0}^{M} (x_n - \Omega_{\text{mean}}(w))^2, \quad \text{and} \\
\Omega_{\text{der}}(w) &= \frac{1}{M-1} \sum_{n=0}^{M} \frac{x_{n+1} - x_n}{t_{n+1} - t_n},
\end{align*}
\tag{5}
\]

where \( w \) is a window identifier, \( M \) the number of samples within the window, and \( x_n \) samples (\( n \) being its index) from the selected input signal within a window, and \( t \) the corresponding time stamps. The variance feature will express how much the signals vary within a window. The variance of the surface distance measurements will for example give the model an idea of the waves surrounding the ship. The derivative feature will tell the model if a signal is increasing or decreasing within a window.

The choice of the window size depends on the application. We found that a window size of three minutes represent a good trade-off between robust estimation of data and time-scale for change in the variables for the instantaneous model. For the given data set using a window size of 3 min gives a total of 9001 windows. Others have found a window size of 10 min to be suitable Pedersen et al. (2009), Leifsson et al. (2008). The tapped-delay network example will have a much shorter length, as here we are interested in seeing the dynamic changes. The window size used for the tapped-delay neural network is 15 s.

<table>
<thead>
<tr>
<th>Description</th>
<th>Mean</th>
<th>Variance</th>
<th>Derivative</th>
</tr>
</thead>
<tbody>
<tr>
<td>Speed through water (target)</td>
<td>√</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Fuel consumption (target)</td>
<td>√</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Trim</td>
<td>√</td>
<td>√</td>
<td>√</td>
</tr>
<tr>
<td>Port and starboard pitch</td>
<td>√</td>
<td>√</td>
<td>√</td>
</tr>
<tr>
<td>Port and starboard rudder</td>
<td>√</td>
<td>√</td>
<td>√</td>
</tr>
<tr>
<td>Heeling</td>
<td>√</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Draft</td>
<td>√</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Port and starboard level</td>
<td>√</td>
<td>√</td>
<td>√</td>
</tr>
<tr>
<td>Headwind and crosswind</td>
<td>√</td>
<td>√</td>
<td>√</td>
</tr>
</tbody>
</table>
Table I gives the features used for the instantaneous model tested. The features speed through water and fuel consumption are model outputs, while the rest are for inputs – there is a total of 29 inputs. The draft and heel are estimated from the trim and level measurements. For the tapped-delay neural network example, we will only use a small subset of these.

3.4 Artificial Neural Network

We will use a feed forward neural network with two layers of adaptive parameters and bias units:

$$f(x; \theta) = \tilde{g} \left( \sum_{j=0}^{M} w_j^{(2)} g \left( \sum_{i=0}^{d} w_{ji}^{(1)} x_i \right) \right).$$

The parameters of the model are the weights of the two layers $\theta = (W^{(1)}, W^{(2)})$. Non-linearity is achieved through the activation functions $\tilde{g}$ and $g$, which are usual taken to be sigmoid-functions such as hyperbolic tangent or the logistic function. In our case we use a linear output activation $\tilde{g}(a) = a$, and non-linear activation for the hidden layer $g(a) = \tanh(a)$. The output of the network therefore becomes a linear combination of the nonlinear activation functions.

In order to control the complexity of the model we use regularization of the weights. This can be formulated in a number of ways. Here we view it as maximum a posteriori (MAP) estimation $\theta_{\text{MAP}} = \arg \max_{\theta} p(\theta|D)$, where the posterior is given by

$$p(\theta|D, \sigma_\epsilon^2, \sigma_\theta^2) = \frac{p(y|X, \theta, \sigma_\epsilon^2) p(\theta|\sigma_\theta^2)}{p(y|X, \sigma_\epsilon^2, \sigma_\theta^2)}.$$

and $p(\theta|\sigma^2)$ is the prior distribution of the parameters. We take this distribution to be independent and identically distributed (i.i.d.) for weights and the prior for a single weight to be $\text{Norm}(\theta, \sigma_\theta^2)$. We can formulate the MAP estimation problem as an equivalent cost function minimization problem by taking the logarithm and omitting terms not depending upon $\theta$:

$$E(\theta; \lambda) = \frac{1}{2} \sum_{n=1}^{N} (f(x_n; \theta) - y_n)^2 + \lambda \|\theta\|^2,$$

where $\|\theta\|^2$ is the two-norm or sum of the squared elements of the weights and $\lambda = \sigma_\epsilon^2 / \sigma_\theta^2$.

Optimization of this non-linear cost function for constant hyper-parameter $\lambda$ is performed by minimizing the error gradient found with back-propagation Bishop (2007). Model prediction is computed according to Eq. (6) using the optimized weights which will be a function of both the training data, the hyper-parameters and possibly the initialization of parameters and learning rates. The learning of the hyper-parameter $\lambda$ is done by $k$-fold cross-validation, where $k=10$. In $k$-fold cross-validation we split the training data into $k$ approximately equal sized sets. We perform $k$ training runs using in turn one of the sets for validation. The average test error over the $k$ runs is proxy for the test error for a model trained on $N - N/k$ examples. We scan over a range of $\lambda$ values and select the one with the lowest cross-validation error, see Section 4.

1 The bias terms are included in a compact formulation by letting the summation start from zero and clamping both the zeroth input and the zeroth output from the hidden unit to minus one.
3.5 Performance measures

The accuracy of statistical prediction models may be quantified in terms of for example squared residuals of the model predictions on a test set and should be judged relative to baselines such as the error using the mean fuel consumption or the variance of the fuel consumption. The effect of single variables (for example the control variables) can also be investigated by comparing performance of models with and without these variables. A model which is successful in terms of significant improvement over the baseline can be considered a simulator of the ship in term of fuel consumption and speed. For regression models a convenient measure is the root-mean-square error (RMSE). It has the same unit as the model output variable, and has a similar form to the standard deviation. Given a data set containing \( N \) windows, the RMSE can be calculated as follows

\[
RMSE = \sqrt{\frac{1}{N} \sum_{n} \left( f(x_n | \theta) - y_n \right)^2},
\]

where \( f \) denotes the model function, and \( \theta \) denotes a model parameter vector. The output of \( f \) for a given an input vector \( x_n \) is the predicted value by the model, and \( y_n \) is the value actually observed.

4. Results

4.1 Instantaneous model results

As mentioned in section 3.4 the data set has been split into two sets, a training set and a test set. The training set is 2/3 of the data. This has been done by i) selecting random windows from the original set and ii) randomly selecting whole trips (sets of windows). The test sets are not used in any way before evaluating the performance of the models. The weight decay parameter of the artificial neural network models are found using 10-fold cross validation as mentioned in section 3.4. The training set is split into 10 approximately equal sized sets; this split is done window wise. Fig. 2 shows the RMSE for the testing set using a different number of trips in the train set. Here the fuel consumption is predicted using the data split by whole trips. This is done in order to get an indication of how much data is required before the model can be used. Ten different trip sequences have been used - the variation resulting from this is illustrated by the box plots. The shape of the learning curve is quite normal: a steep initial decrease followed by a slower (typically power-law) decrease.

Fig. 2: Test set RMSE as a function of the number of trips in the training set. The fuel consumption is predicted using the data sets split up by whole trips. The error bars are obtained by ordering the way the trips are included in the data set in 10 different ways.

Fig. 3: The output from the final ANN model and the corresponding target (true) values. The data points are from three selected trips from the test set (split trip-wise). The trips are separated by vertical dashed lines in the plot.
By splitting up the sets by trips, gives us the ability to examine whole trips from the test set, because otherwise it would be likely that data points from a trip would end up in both the training and testing sets. Three selected trips are shown in Fig. 3, and the predictions obtained from the ANN. It can be seen that the model is able to predict the changes within the trip. These plots give a qualitative impression of the models.

Table II gives the performance results obtained for the instantaneous models, along with the cross validation results, which are expressed as a mean value and a standard deviation. As one might have expected, the models perform a little better using the dataset where the windows are shuffled, due to the cross talk mentioned in section 3.1.

A direct comparison with similar work Pedersen et al. (2009), Leifsson et al. (2008) is hard because the sensor data is different. Pedersen et al. (2009) report, as best result with an ANN model, a mean relative error on propulsion power of 1.65%. Our results for ANN models are 1.50% for the fuel consumption using the dataset split by windows and 1.67% if split by trips. Leifsson et al. (2008) report 0.65 knots RMSE for the speed and 60 L/h for fuel. Using the mean relative error is a disadvantage to our model and data, because we have relatively many samples where the speed is low compared to the two other articles.

Table II: Performance results for the instantaneous ANN model

<table>
<thead>
<tr>
<th>Performance</th>
<th>Shuffled windows</th>
<th>Shuffled trip</th>
</tr>
</thead>
<tbody>
<tr>
<td>Test</td>
<td>0.32</td>
<td>41.1</td>
</tr>
<tr>
<td>X-val</td>
<td>0.32 ± 0.012</td>
<td>41.06 ± 0.76</td>
</tr>
</tbody>
</table>

Figs. 4 and 5 show the histograms for the residuals for two modeled feature values. The distributions of the residuals have been approximated by a normal distribution and Student’s t distribution. The Student’s t distribution has longer ‘tails’ than a Gaussian. This makes the distribution much less sensitive than the Gaussian to the presence of a few data points which are outliers, Bishop (2007). The probability density distribution for the speed and fuel residuals is narrower than the fitted normal distribution. Outliers often arise in practical applications either because the process that generates the data corresponds to a distribution having a heavy tail or simply through erroneous data.

4.2 Tapped-delay neural network model and residual modeling

We will keep the tapped-delay neural network example simple, and only include a few parameters in the model. The following control features have been used: Port and starboard pitch, port and starboard rudder, initial port and starboard level, difference between ground and speed through water, headwind and crosswind. The following dynamic variables have been used: Speed through water, port and starboard level, trim, draft, difference in heading. The model is trained on 2/3 of the data using a
range of weight decays values, and the model with the best performance based on the rest of the data is selected (validation set).

The predictive model is evaluated using Eq.(4). The noise, $e(n,w)$, will be simulated using a Gaussian and Student’s $t$ distribution fitted to the residuals on the validation data. We will use the univariate distribution here for each variable. The model can be initialized with some reasonable settings for the dynamic values, and the model’s response to a given input can be found. Including the noise in the dynamic states will give an idea of the sensitive the system is. By examining the plots in Figs. 6 and 7 where the histograms for two of the dynamic variables are given, it is evident that the student’s $t$ distribution is a good representative for the residual noise.

![Histogram for the draft residuals using the predictive model](image1)

![Histogram for the speed residuals using the predictive model](image2)

![Propeller pitch (Input signal)](image3)

![Speed through water (Dynamic variable)](image4)

![Tapped-delay neural network response to a (rectangular) step function for the propeller pitch. Both propeller pitches are set to the same value. The solid gray area gives the models sensitivity to added Gaussian multivariate noise.](image5)
As an example, we have generated 100 dynamic variable samples, and propagated these through the tapped-delay neural network for 90 steps, where each step is 15 seconds. The propeller pitches are changed from being 0 for 30 steps, to being 90 for 30 steps, and then 0 again for 30 steps. The control signal is drawn in the upper plot in Figs. 8 and 9, and the tapped-delay neural network’s response is given in the lower plot. The samples have been ordered after the speed through water value, and the area between the 5 lowest and highest values is given by the solid gray interval in the plots. Fig. 8 gives the response from the tapped-delay network with added Gaussian noise, while the plot in Figur 9 gives the response with added noise from the Student’s t distribution. If we look at the plots, we can see that the uncertainty in the speed prediction grows with time as the noise accumulates. It also seems like the noise is worse with lower speeds than higher speeds – a possible reason for this could be that we have much more data at the ship’s cruising speed, and a smaller part of the data at lower speeds. The system also stops storing data just before the ship stops, so the system newer actually experiences the ship being totally still.

If used correctly this model approach gives us the ability to determine how a change in a control will affect all of the dynamics of the ship; for example how changes in propeller pitch would affect the trim of the vessel. This is a clear advantage compared to the instantaneous model.

![Propeller pitch (Input signal)](image)

![Speed through water (Dynamic variable)](image)

**Fig.9:** Using the same model and control signal sequence as in Fig. 8, but now the noise is drawn from the student’s t distribution

### 5. Conclusion

A non-linear ANN model has been used for modeling of fuel efficiency in ship propulsion. The instantaneous model, using a artificial neural network model, has been used before in similar work Pedersen et al. (2009) and Leifsson et al. (2008). The results obtained are similar to those of previous works; it is however difficult to compare these results, because the data used is different. Difficulties comparing our results with previous work might indicate that there is a need for some publicly available data, which can be used for benchmarking these models and methods. The high quality
sensory data set presented here can hopefully fill this gap, and be used as a common reference, or encourage others to make their data publicly available.

Our ongoing work within this area will focus on improving these models, especially with regard to the problem that several state variables are affected when changing one of them. This is a problem that for example can occur when using these models in a trim optimization application. For example advising a trim that is far from the current trim, will make the inputs invalid when the new trim is reached, and the model will give different results. It is plausible that a state-space model or similar may be able to handle this problem.

References


SVARER, C.; HANSEN, L.K.; LARSEN, J. (1993), *On design and evaluation of tapped-delay neural network architectures*, IEEE Int. Conf. on Neural Networks

TOWNSIN, R. (2003), *The ship hull fouling penalty*, Biofouling 19, pp.9–15

Abstract

This paper presents CAFE, a new and innovative system for rapid generation of ship concepts. The enhanced and simplified man-model interaction utilizes arbitrary inclined guiding planes. These guides can add, reshape, move or split any entity in a 3D model. In addition, guides can be moved along with all attached plates, girders, stiffeners and brackets. Attached elements change their shape and number of stiffeners if necessary. Another important feature is the magnetic surface concept. This enables automatic surface reshaping to fit the shape of another neighbouring surface. Internal structure along with the imported equipment can be visualized by walk-trough visualization capability. The system calculates masses and position of the structural centre of gravity. Furthermore, it automatically generates classification drawings at desired intersection planes of the 3D model. Apart from aiding the ship concept design, this system serves as a FEM pre-post processor. It is capable of fully automatic meshing and refinement of the existing FE mesh. It can apply structural loading including wave loads utilized by Smith correction. Additionally, the system is capable of balancing the loads and reaction forces on boundary conditions in case when a full ship FE model is analyzed.

1. Introduction

At the present time, shipbuilding is in a difficult position. Most of the industry is faced with the problem of profitability. Not only that the price of the ship is reduced due to the strong competition, but also design and building of a ship have to be completed in tight schedules to avoid the excessive penalties. Therefore, any time savings are beneficial in both, ship production and ship design. Good and fast ship concept design is mostly dependant on the ability of the designer. However, an efficient design software tool can aid the designer in speeding up the design process. There are several general purpose CAD tools available that can largely contribute to a fast and an efficient ship concept design, such as CATIA. Furthermore, there are software tools such as CADMATIC, SMARTMARINE, NAPA and SHIPCONSTRUCTOR which are specialized in ship design. Most of these specialized tools are able to perform the concept and detailed ship design with a strong emphasis on excellent solutions that enhance the speed of the detailed design. However, improvements in solutions for the concept design can still be made.

This paper presents CAFE, the system for ship concept design that can enhance the design time efficiency by utilizing innovative ideas for rapid ship modelling.

2. Description of CAFE system

CAFE is a new and innovative user-oriented system for rapid generation of a ship concept design: see Fig.1. The enhanced and simplified man-model interaction aims to reduce the required ship modelling time. Modelling features enable the user to simply model the ship structure that includes plating, openings, girders, stiffeners and brackets. Equipment with its respective mass can either be modelled or imported from other CAD tools. Of course that it is more convenient to import the equipment as 3D blocks and position it to desired coordinates in the ship structure. Additionally, tube layouts and wirings can also be accommodated in the same model. The system automatically calculates masses and position of the structural centre of gravity including ship hull, the installed equipment and piping. Internal structure along with the imported equipment and piping can be visualized by walk-trough
visualization capability in a form of the virtual camera that can explore the inside ship structure. The walk-through capability can be used also during the modelling, not only for visualization purposes. In addition to centre of gravity, the system can also provide volume calculations of the hull and any tanks modelled within the structure. Designers can collaborate in real-time by working on the same model and also communicate by using the built-in chat module. Individual parts of a model can be protected by locking instruction. Each designer claims the ownership on certain part of the model. In case that the other designer needs to change the part of the model which is not under his ownership, he or she has to send the request for transfer of the ownership to himself. The system can automatically generate classification drawings of the structure at arbitrary intersection planes of the 3D model, but also it can create the shell plating drawing. General arrangement drawings can be created in the same way as structural drawings. General arrangement includes plots of the equipment in the form of outlined objects or predefined views (front, side, top etc.).

Fig.1: CAFE system’s computer interface

Several innovative ideas have been implemented into the system in order to enhance the modelling efficiency. These ideas include FEM pre-processing tools, flexible guiding planes and magnetic surfaces. Innovations will be explained in more details in following chapters, while here, only briefly mentioned.

Apart from aiding the ship concept design, this system serves as a FEM pre-post processor. It is capable of fully automatic meshing and refinement of the existing FE mesh for detailed stress analysis purposes. It can apply structural loading including wave loads utilized by Smith correction. Additionally, system is capable of balancing the loads and reaction forces on boundary conditions in case when full ship FE model is being analyzed. FE model, including balanced loads and boundary conditions, can be exported to several FE solvers for assessment.

CAFE is capable of generating a mesh for hydrodynamic numerical analysis. Generated mesh can be exported for the purpose of the assessment.

The system utilizes arbitrary inclined guiding planes. These guides can add, reshape, move or split any entity in a 3D model. In addition, guides can be moved along with all the attached plates, girders, stiffeners and brackets. Attached elements are self-adjustable, meaning that they can change their shape and number or spacing of attached stiffeners if necessary.

This system is featured with another important invention called the magnetic surface concept. This
concept enables automatic surface reshaping to fit the shape of another neighbouring surface. The feature is very easy to handle and it avoids trimming of the surfaces. Avoiding trimming is of great benefit while exporting geometry for other CAD tools.

3. System features

The system is enhanced by innovative features that aid ship modelling. These features are implemented in order to enable faster and easier user interaction with the system. The enhanced interaction with the system brings large time saving during the modelling. Several of these innovative features are described in more details in following subchapters.

3.1 Surface generation

A high speed curve generation algorithm was developed by Chaikin (1974) and adopted by Riesenfeld (1975). The basic idea of the algorithm is the cutting corners principle. The algorithm starts with a set of points which are mutually connected by straight lines. The curve is created by cutting corners at points where two lines are connected. Cutting corners results in converting sharp edges into rounded ones while number of refinements goes to infinity; see Fig.2. In practice, 3 or more refinements at one point are sufficient to get a smooth curve. Original points, that have been cut-off, become control points of the generated curve. It can also be easily proved mathematically that this algorithm ends with a quadratic B-spline, Kenneth (1996).

![Fig.2: Cutting corners principle](image)

The described algorithm is modified in a way that original user-defined curve points become points on the generated curve, Bralic et al. (2010). By using the modified version of the algorithm, a user can define exact points through which curve is generated, thus making less effort for modelling. Additional modifications which are introduced to the original curve are knuckle points. At these points user can define sharp corners or prescribed tangent angle of the curve at the knuckle point.

![Fig.3: Surface definition](image)
A new surface is generated automatically by combining several definition curves created using the
modified algorithm described above. Fig.3 shows a new surface automatically defined as a set of quad
or triangle elements. Number of element segments of the surface is based on prescribed precision.
Several shading display options in CAFE system can aid the designer in fairing the generated surface.

3.2 Guiding planes

CAFE system utilizes orthogonal or arbitrary inclined guiding planes. Orthogonal guiding planes are
parallel to planes defined by pairs of coordinate axes of the coordinate system, while arbitrary guiding
plane is defined by any 3 points in space. These guides are invented to aid the designer while
positioning, splitting, moving or reshaping structural entities in a 3D model. Fig.4 shows a partial ship
model with one active arbitrary inclined guiding plane displayed in transparent colour.

Guiding planes can be used to split any entity in the model. Fig.5 shows symbolic split operation of
the vertical surface entity by guiding plane which is perpendicular to it. Guiding plane is shown on the
left side of the picture displayed in transparent colour. On the right side it can be noticed that guiding
plane is deactivated and that vertical entity is split in two on the position where guiding plane was
positioned.

Besides splitting, guiding planes can be used to move or reshape entities. Fig.6 shows the same
symbolic example from Fig.5 where upper part of the split entity is attached to the guiding plane. By
moving the guiding plane, upper entity is moved along, while lower one is reshaped which in this particular example manifests as stretching.

Fig.6: Moving and reshaping entities by guiding plane

In a ship scale example, guides can be moved along with all attached plates, girders, stiffeners and brackets. Attached elements are self-adjustable, meaning that they can change their shape and number or spacing of attached stiffeners.

3.3 Magnetic surfaces

Magnetic surface concept enables automatic surface reshaping to fit the shape of another neighboring surface. These entities are used when surfaces are modeled, but are not fully aligned to each other. This can be a very useful feature for entities that are connected to the shell plating, e.g. when floors aren’t properly attached to outer shell due to copying the entity from another location. Fig.7 shows an example of snapping the floor to the outer shell. Giving the outer shell the magnetic feature and slightly moving the floor using a computer mouse will result in snapping of the floor to the outer shell. Left side of the Fig.7 shows the floor unattached to the outer shell and viewed from the outside of the model. Right side of the Fig.7 shows the floor when it is properly snapped to the outer shell and viewed from the inside of the model.

Fig.7: Snapping the floor to the outer shell by using magnetic surface feature

Magnetic surfaces can be used to shape curves as well. By using the guiding plane with magnetic surface feature, it is easy to project the circle on the curved surface and to model the tunnel attached to the outer shell such as shown in Fig.8. Procedure is principally shown in figure sequence going left to right.
3.4 Equipment

A ship modelling process is greatly simplified by the ability to import the equipment into the system. The system already possesses a large library of equipment products. However, it is possible to import any 3D part or the equipment designed in other CAD tool and exported in VRML format. While importing the equipment, user needs to specify its mass and centre of gravity. Fig.9 shows examples from system’s equipment library.

The equipment gives the realistic view of the model and, more importantly, provides a spacing control between the structure and the equipment. Naturally, the equipment modelled in CAFE can easily be exported to other CAD tools as well. Properly imported and positioned equipment, such as engines and winches, can be seen on the example of the supply vessel in Fig.10.
3.5 Classification drawings

The system automatically generates classification drawings of the structure including shell plating. General arrangement drawings can be created in the same way as structural drawings. General arrangement includes plots of the equipment in a form of outlined objects. Drawings can be generated by positioning a guiding plane on desired section of the 3D model. Considering the fact that different organizations use different notations and drawing marks, it is made possible to add and adjust them easily. This ensures minimal additions and time spending to complete the classification documentation. Fig.11 shows the example of automatically generated drawing that represents only part of the deck.
3.6 FEM pre-processing

CAFE system is featured with a FEM pre-processing tools. It is capable of fully automatic meshing the modelled entities using arbitrary sized elements. Fig.12 shows the FE mesh of the structure where on the left side is the original model, while right side represents the meshed model. Number of elements used in the system is not limited. The refinement of the existing FE mesh for detailed stress analysis purposes is possible as well; Fig.13.

Fig.12: FE model

The system can apply any structural loading including wave loads utilized by Smith correction. Wave loading is displayed in Fig.14 where only hull form is modelled.

Boundary conditions for FE purposes can be easily prescribed. Additionally, system is capable of balancing the loads and reaction forces on boundary conditions in case when full ship FE model is being analyzed. FE model, including balanced loads and boundary conditions, can be exported to several FE solvers for assessment such as NASTRAN, FINNSAP, ANSYS and LS-DYNA. It is also possible to import the results of the strength assessment done in NASTRAN and make them visible from within the system.

Fig.13: Refined FE mesh for detailed stress analysis
3.7 Other useful features

Very important feature of CAFE system is that it doesn't use relationship database for model storage. Relations between objects are obtained either by connecting nodes or by parent-child relations, e.g. if some finite element is generated automatically from stiffened panel, it will inherit all its properties.

Another positive feature of CAFE is that while system is running, it has two separate databases allocated in the memory. One is used for visualization (modelling) and another as a storage and user management (synchronization). This fact enables an easy handling of collaborative design.

Structural organization of the database enables drastic savings in storage size, e.g. a full model in conceptual phase is about 1 MB, while the same is 1 GB when a relationship database is used for the same model. Reduction of the model file size helps the collaboration in a way that large portions of the model can be quickly exchanged over the internet.

4. Conclusions

CAFE system, presented briefly in this paper, aims to utilize innovative ideas in order to enhance and simplify the man-model interaction. The system was used on several ship design concepts so far. Tests proved that modelling a ship design concept, using CAFE system, is fast and convenient according to the subjective opinions of experienced designers. Furthermore, the potential of this system lays also in a fact of time saving when automatically producing classification drawings. Additional financial benefit can arise from the fact that design offices can generate their own numerical models for assessments and speed up the process, and thus avoid double model definition.

References


Evaluating Evacuation Simulation Results in a Virtual Reality Environment

Benjamin Mesing, Fraunhofer IGD, benjamin.mesing@igd-r.fraunhofer.de
Florian Kluwe, Flensburger Schiffbaugesellschaft, kluwe@fsg-ship.de
Uwe von Lukas, Fraunhofer IGD, uwe.von.lukas@igd-r.fraunhofer.de

Abstract

Safety is considered as one of the key issues in ship design and operation, especially for passenger vessels. This motivates research applying advanced computer graphics technology for ship safety analysis. We have implemented a scenario allowing the evaluation of evacuation simulation results carried out for passenger vessels in a virtual reality (VR) environment. By combining available 2D simulation data with the 3D data of the ship, additional insights with respect to critical points such as stairways and narrow passageways can be gained, which opens up the opportunity to early design improvements. One of the major technical challenges solved in this context, is the real time visualization of the fully outfitted ship. When reviewing the simulation within the virtual environment, the evaluator can navigate intuitively, using 2D-deck plans on a multi-touch device.

1. Introduction

Safety is one of the key issues in ship design and operation, especially for passenger vessels. In that respect IMO rules (IMO MSC/Circ. 1238) require evidence that all passengers can be evacuated safely within a total time of 60-80 minutes. In order to achieve this goal, corridors, stairways, doors and escape ways have to be dimensioned accordingly. The interior design of vessels with respect to the aforementioned elements is highly regulated by rules prescribing minimum dimensions for the individual elements. However, pure fulfillment of these requirements does not necessarily lead to a beneficial and effective overall arrangement of escape routes. Thus, the overall performance of a specific evacuation concept is highly influenced by the designer’s personal experience. If novel or unusual internal arrangements shall be realized, more rational tools for decision making are required. Today, numeric simulation techniques in conjunction with first principle scientific models can provide powerful assistance in assessing such problems.

In this respect, simulation codes are used for the demonstration of compliance with IMO rules already by standard. Those tools, for example AENEAS, developed by Germanischer Lloyd and TraffGo-HT\(^1\) are usually based on a deck-wise 2D-representation of the ship’s internal arrangement, where persons’ movement is tracked in an underlying grid-structure, Fig. 1. These tools are suitable for calculating the overall evacuation time as required for the demonstration of compliance with the rules. Additionally, they provide a general picture of the evacuation process, helping to identify potential bottlenecks in the concept. For this purpose an initial distribution of evacuees is given, whereas different scenarios are tested. This usually comprises a night-condition, where most of the persons start to leave from their cabins and a day-condition where the majority of persons are initially situated in the public areas of the vessel.

2. Novel Approach

However, a detailed analysis of the identified problems and the influence of several contributing factors like smoke, listing vessel and irrational behavior under stress can not be addressed with such type of simulations. For this purpose, advanced visualization techniques can be applied to approach these specific problems.

\(^1\) http://www.traffgo-HT.com/de/pedestrians/products/aeneas/index.html
Here virtual reality can play a major role as it allows for a demonstrative presentation of the simulation results on a low level of abstraction. By combining available 2D-simulation data with the 3D data of the ship, additional insights with respect to critical points such as stairways and narrow passage ways can be gained, which allows for optimization of the interior layout in the early stages of product development, where costs are still low for such changes.

When reviewing the simulation, the evaluator can choose to take the perspective of an evacuee and to follow his movements or to freely navigate through and around the ship which provides a subjective perception of the situation of an evacuee on board of the ship. This again provides the chance to detect deficiencies in the layout early, here mainly with respect to the clear and simple arrangement of alleyways and stairs. All this generates added value for the shipyard by design improvement, safety enhancement and cost reduction.

In this paper we describe the realization of a virtual reality-based review of such a scenario. Within the virtual reality environment the 3D data of the ship together with simple representations of the evacuees is displayed, Fig. 2. The position of the evacuees changes over the time and is based on the simulation data. During the simulation the user can freely navigate through the scene and control the replay of the simulation. To provide a better overview, the 2D-plans of the ship with the positions of the evacuees are also presented to the user. Major technical challenges we had to solve, were displaying the results of the 2D simulation within the 3D model of the ship, providing a user-friendly possibility for navigation and coping with the large amount of objects from the fully outfitted ship.
3. Implementation of the Evaluation Scenario

We have realized this scenario based on the X3D ISO standard, http://www.web3d.org/x3d/specifications/. It is a standard for describing interactive virtual worlds and widely supported among different VR systems. Aside from providing means for describing the geometry, it also provides a programming interface for implementing additional functionality in a supported programming language like Java or ECMAScript. We have used the X3D-based mixed reality framework instantreality for implementation.

The full life-cycle of a VR session generally includes the steps data preparation, authoring of the scenario logic, execution and post processing, Fig. 3.

The two steps for the preparation are described in more detail here:

1. Preparing the data including conversion to a VR format and in most cases reducing the complexity to ensure real time behavior.
2. Authoring includes configuring the VR system to the needs of the scenario, defining the interaction possibilities and adding the scenario specific logic
The first is especially important when dealing with the complex models of the shipbuilding sector. The second is needed, because we strongly believe, that for every specific scenario and every specific user group, customization is required to give the user the most effective tool for his task. In the remainder of this section, we will detail on the steps we have performed to realize the scenario for validating evacuation simulation results.

3.1. Data Preparation Chain

In an initial step, the geometry had to be extracted from the AVEVA Marine PDM system used by FSG and converted into a format understood by the used virtual reality system. In the shipbuilding sector STEP is considered a good choice for extracting data from the PDM system for visualisation purposes, Broas et al. (2009). The experience of FSG confirms that STEP is currently the best choice for exporting from AVEVA Marine, providing good export quality and being well supported by a wide range of tools.

One of the objectives of the scenario is, to find potential obstacles within the passageways that are not considered by the simulation software. This means, that all objects potentially located in the passageways or staircases need to be displayed while other objects could be excluded. The naming scheme at FSG allowed filtering out part of the unneeded assemblies like pipes from the air conditioning system based on their names. The remaining assemblies are extracted as STEP files and converted to X3D using the converter application NuGraf/ PolyTrans from Okino Software.

The export of all relevant data objects within the passenger area still leads to over 17,000 STEP files of varying internal complexity, ranging from simple pipes to wholly equipped cabins. Because the STEP export preserves full hierarchy information, we have to deal with approx. 1 million parts. Such a large amount of objects cannot be handled by most VR systems. This can be explained by the fact that those systems are optimized for displaying a large number of triangles and traversing complex hierarchies significantly reduces the performance of the rendering. However, for our scenario interaction with specific parts is not required, and object hierarchies can be flattened without loss of functionality. Therefore, we configure an aggressive hierarchy optimization within the NuGraf converter, leading to usually no more than a few objects within a single file and a total number of ~35,000 objects.

To keep the number of polygons manageable and because in this scenario, no high precision for the geometry is needed, we convert the STEP files with the coarsest resolution available, which still produces visually acceptable results. The resulting number of polygons (~14 million) prevents a fluent display (i. e. frame rates of >20 fps) on the high quality consumer graphic card (GeForce GTX 285) we have used. To avoid the necessity for expensive high-end hardware advanced visualization techniques can be used to deal with this complexity.

3.2. Visualising the Whole Model

To be able to fluently explore the full model with standard graphic hardware, we utilise a room based approach, which is also used in many graphically demanding computer games. The approach makes use of the fact, that within a single room, mainly the interior of this room is visible. Therefore, not the full model has to be shown but only the room the user is currently exploring. The most common alternative approach – to perform a polygon reduction – is less desirable because it would lead to a significantly decreased visual quality if only the reduced model is used or an increased amount of memory usage and increase load time in case of a level of detail approach, where the full and the reduced models are loaded, and depending on the distance to the viewer either the smaller or the larger model is displayed.

In our realisation the extensions of the rooms are defined manually by the evaluator (18 rooms in total) matching the room structure of the real model, Fig. 4. Based on this input, a list of parts within each room is generated by a custom PML macro from AVEVA Marine. We provide a module for the
VR system, which evaluates this list and controls the display of the objects accordingly. When a user walks into a room, all details of the room are displayed and when he exits, they are hidden again. To provide a general orientation the hull is always shown, regardless of the user’s position. This approach allows for freely exploring the ship and still having a fluent visualisation.

![Fig. 4: Sections for room based visualization (equipment and hull not shown)](image)

The approach can also be adapted, to reduce the amount of main memory required for running the VR session. Instead of merely toggling visibility, the data is loaded/unloaded when entering/exiting a room. This technique allows us to even display the large model on a WinXP32 system with the well-known 2GB-limit of the main memory. However, this technique has a major drawback: When switching rooms, it takes a notable amount of time to load the details from the hard drive.

The room based approach as presented above has two drawbacks:

- Details directly in front of the user are not displayed if they are in another room – even if they should be visible e.g. through an open door or a window.
- Manual effort is required to define the room boundaries.

The first could be solved by letting the areas overlap. However, this again would double the data volume to be displayed when the user positions the camera near the room border. Therefore, we implemented an approach where only the objects within a certain range (e.g. 20 m) are displayed enabling a fluent navigation without requiring to manually select sections to be drawn. Again the approach can be adapted to on demand loading from the hard drive reducing the memory consumption but producing smaller but noticeable delays. Depending on the scenario and the available hardware, the suiting approach for visualisation can be chosen.

### 3.3. Scenario Implementation

In addition to displaying the 3D objects from within the ship, we also display other information relevant to the evacuation scenario. These are:

- the evacuees at the simulated positions within the 3D model of the ship,
- the emergency signs (which were not stored within the 3D model) and
- a simple fog-like representation of smoke.

For displaying the evacuees within the 3D model, two steps have to be performed: First the simulation software has to be interfaced. Since no access API is provided by the TraffGo Software and no life evaluation of the results is required, we base our interface on the file containing the simulation results.
The file is parsed and the programming interface of X3D is used, to implement a script providing replay functionality. Second the evacuees must be positioned correctly within the model. For each evacuee a 3D model is inserted into the scene. To keep the number of polygons low, the representation is kept very simple. The replay script sets the position of those models according to the current time and the positions stored in the file. Additionally a user interface to control the replay script and various other parameters is provided. Fig. 5 illustrates the overall architecture.

![Fig. 5: Evacuation Simulation Control](image)

Fig. 6: Simple smoke effect enhancing appraisal of the situation (different, smoke intensities shown)

Fig. 7: Visibility of emergency exit signs in different lighting conditions (exit signs do not illuminate their surrounding)
A simple approximation of smoke is also implemented. It makes use of a fog effect, where objects in the distance are slowly blended to grey. The “density” of the fog can be modified through the user interface, Fig. 6. Additionally the intensity of the global illumination can be modified so the effect of the self lighting emergency exit signs can be experienced, Fig. 7.

A special focus of our implementation is on the user interface. Interaction in VR is usually considered a difficult task for sporadic users and no standard technique has yet been established. We therefore chose an approach, where an additional multi-touch 2D device is used for interaction. The device can be e.g. a smart phone or a tablet PC held in one hand or a larger multi-touch table positioned in front of the VR screen. The user sees a standard 2D user interface on the display with buttons and sliders to control the different parameters of the scene. Additionally, 2D plans of the different passenger decks together with the evacuees’ positions are displayed, Fig. 8. The 2D elements can be freely arranged, by rotating, moving and scaling using well known and natural two-finger gestures.

![User interface for navigation and system control](image)

For navigation we utilize the 2D plans for a map based travel approach. Map based navigation is an established navigation technique for single-floor structure for many years, Darken and Sibert (1993) and has recently shown to be an effective navigation aid and control mechanism for multi-floor structures, too, Chittaro and Venkataraman (2006), Jung et al. (2009). In our interface the displayed 2D plans can be selected by touching them at a predefined spot. Once selected, touching a point on the plan with a single finger changes the position of the viewer in the 3D scene to this point. A second finger controls the direction the user is looking to, Fig. 9. This way, the user can explore the ship by moving his fingers across the 2D plan. Of course classic 3D navigation devices like a GamePads or a SpaceMouse can also be used for navigation. We expect this combination of 2D and 3D allowing for a very efficient usage, since the evacuees’ positions are also displayed on the 2D plans, the evaluator always has an overview over the whole situation. He can select narrow spots and examine them in the VR view. As an additional mean for examination it is possible to cut the ship open by hiding all objects from a certain height. However, due to the performance limitations mentioned above, it is currently not possible to fluently display the outfitting objects in this case.

In addition to the plan-based navigation the evaluator can also choose to take the perspective of an evacuee. This mode is activated by picking one of the avatars and can be used to follow his movements and experience the situation from his perspective.
We have presented the scenario to different people involved with planning of escape routes and running evacuation simulations at the FSG shipyard. Because no VR installation was available on-site, we presented the scenario using a tablet PC as input device and displaying the 3D-model using a standard projector.

The general statement was that the scenario provided a realistic impression of the evacuation situation. The very simple display of smoke, significantly helped appraising the evacuation situation by making clear, how limited the orientation is in a smoke-filled room. Though the benefit of using this tool for evacuation layout for an experienced designer was considered to be only small, the benefit as a means of communication with non-experts was emphasized. During the evaluation we pinpointed several areas marked as free within the plan for the 2D evacuation simulation, which were actually blocked by equipment, Fig. 10.

For the user interface, the combination of using 2D plans for directly navigating to a target point and then using a GamePad for smaller changes of the position from there was considered the optimal solution for the tablet PC. The display of the tablet PC was not large enough to comfortably make small movements on the deck plans. Also the possibility to freely arrange deck plans and the other
user interface elements (allowing for an overlap) was not intuitive to use. Due to experiences in the other domains, we expect both points to be evaluated differently when a larger touch interface (e.g. the multi-touch table used for testing) is available, Jung et al. (2009).

4. Conclusion and Future Work

Displaying evacuees at their simulated positions within a virtual environment provides a realistic impression of the evacuation situation. The display of environmental conditions like smoke or reduced lighting significantly improves the appraisal of the situation of the evacuees. The display within a virtual environment especially helps when discussing the layout of escape routes with non-experts and for validating the 2D plans used for the evacuation simulation.

For realization of the scenario the large amount of data must be prepared in such a way, that the virtual reality system is able to display it fluently. We have outlined several approaches for dealing with all the data required for the evacuation scenario. We have also presented an intuitive interface for navigation based on 2D deck plans of the ship, which in combination with a GamePad allows even inexperienced users to easily navigate in the virtual ship.

For the future use, a more realistic simulation of environmental effects like smoke and illumination would be beneficial. Such effects are often used in virtual training environments for firefighting, St. Julien and Shaw (2003), and require a realistic smoke simulation for the ship as well as setting the light sources according to the positions of the lamps within the ship. Additional benefits could be achieved by using the scenario for training the crew for correct behavior in emergency situations.

Acknowledgement

The work presented here was conducted in context of the research project POWER-VR, funded by the German Federal Ministry of Economics and Technology (BMWi) based on a decision of the German Bundestag.

References

BROAS, P.; SALONEN, T.; SÄÄSKI, J. (2009), Simulations, virtual and augmented reality technologies for ship life-cycle engineering. 8th Int. Conf. on Computer and IT Applications in the Maritime Industries, Budapest, pp.486-97


St. JULIEN T.U., SHAW, C.D. (2003), Firefighter command training virtual environment. Conf. on Diversity in Computing. ACM, Atlanta, pp.30-33

Introducing Highly-Efficient CAE Pre- and Post-Processing Solutions in Maritime Design

George Korbetis, BETA CAE Systems S.A., Thessaloniki/Greece, ansa@beta-cae.gr
Dimitris Georgoulas, BETA CAE Systems S.A., Thessaloniki/Greece, ansa@beta-cae.gr

Abstract

In our days, the CAD and CAE models in maritime products development become increasingly complex while more analyses are necessary before a new design is ready to be constructed. Additionally to the standard assessments, extensive calculations are often needed to ensure the product’s performance characteristics and minimize the failure risk through its lifetime. For the fast employment of such analyses with CAE simulation tools, the use of high efficient pre- and post-processing software becomes essential. This work showcases how ANSA pre-processor and µETA post-processor fulfill this requirement of the Maritime Industry by offering sophisticated tools for advanced simulation techniques, automation capabilities and robust performance.

1. Introduction

The scope of this paper is to present an efficient way to set up CAE analyses, for several disciplines and load cases, starting from a common CAD model. Three case studies are demonstrated where a handysize class double skin bulk carrier is subjected to static, full scale collision and CFD analysis. The recommended process includes, among others, model simplification and idealization, meshing, element quality improvement, loads and boundary condition definition. The behavior of the bulk carrier model can be improved according to the above analyses using optimization techniques. The use of the capabilities of ANSA pre-processor in model shaping and parameterization for the set up of such optimization processes is also demonstrated.

2. Process management

During the design process of a product, the CAD model is distributed to the CAE departments to perform simulations. However, continuous updates of several model parts can cause problems to the analysis flow since the CAE models should be updated with the new version. Furthermore, the complex analysis process should be standardized in order to eliminate any dependency of the model quality to the engineer’s expertise. The standardized process should be shared among the engineers and thus to exchange the analysis know-how. A specialized tool for the process organization is provided within the ANSA pre-processor, the Task Manager, Fig. 1.

All analyses that the bulk carrier will be subjected can be defined in the ANSA Task Manager. All the actions needed to define the FE models for the different load cases or disciplines are set in this tool, in a step-wise sequence. Running the Task Manager sequence, ANSA realizes every Task Item and performs the corresponding action on the model. When needed the user is prompted to interact. Furthermore, the Task Manager checks if every Task is defined correctly.

The process starts from the collection of the CAD data that is common for all analyses. Starting from the geometric model, ANSA is able to create different representations for each part of the assembly, which suit the requirements of the different analysis types. Thus, a part can have representations with different geometrical detail level, meshing parameters, element quality criteria, etc. according to the analysis needs. When the Task Manager runs to create the FE model for a specific analysis, it composes the assembly by collecting the appropriate part representations.
The meshing parameters and quality criteria are prescribed for each part, assembly or region within the ANSA Batch Meshing tool, creating meshing scenarios. These scenarios can be related to each analysis so a part can be meshed with different parameters providing a part representation for each analysis as shown at Fig. 2. A meshing scenario is defined for the sagging case study which applies mesh with different element length to various parts of the assembly, Fig. 3.
3. Sagging case study

The first case study is a static analysis of a full structural model of a bulk carrier. The model is subjected to sagging loading conditions when the ship’s holds are fully loaded. The target of this analysis is the determination of the maximum stresses on critical areas. Geometrical model and specifications are presented on Fig. 4 and Table I.

Table I: Ship model specifications

<table>
<thead>
<tr>
<th>Type of vessel</th>
<th>Handysize class double skin bulk carrier</th>
</tr>
</thead>
<tbody>
<tr>
<td>Length</td>
<td>169 m</td>
</tr>
<tr>
<td>Breadth</td>
<td>25 m</td>
</tr>
<tr>
<td>Depth</td>
<td>18 m</td>
</tr>
<tr>
<td>Lightweight tonnage</td>
<td>9500 t</td>
</tr>
<tr>
<td>Deadweight tonnage</td>
<td>26000 t</td>
</tr>
<tr>
<td>Number of holds</td>
<td>6</td>
</tr>
</tbody>
</table>

3.1. Model set-up

The cargo ship model is relatively large so, coarse mesh should be applied to avoid very long simulation time. In addition, geometrical simplifications should be applied on the model. The first action to simplify the model is to fill small holes (diameter < 0.4 m) that are not significant for the model behavior. Such holes are automatically identified by their diameter and filled. This action improves the elements quality while reduces the number of elements, Fig. 5. The process of filling holes is prescribed at the meshing parameters of a Batch Meshing Scenario that is applied to the respective parts by the Task Manager. Thus, the whole process runs in batch mode without the need of user interaction.
The second simplification action is the replacement of longitudinal stiffeners by beam elements. This action reduces significantly the number of the small elements that represent the stiffeners. The beams that are applied have the same characteristics and behavior with the replaced stiffeners. Beams replacement is an automatic process in ANSA that is able to replace the whole model’s stiffeners with little interaction. Beams are offset and oriented to fit the stiffener position while they are connected to the shells of ship model, Fig. 6.

The meshing parameters and quality criteria are defined in the ANSA Batch Meshing tool. When the Task Manager runs the process the mesh is applied to the whole model. The re-meshing algorithms ensure that the generated mesh fulfill the prescribed quality criteria. The user can identify critical areas on the model where accurate results should be extracted. A different meshing scenario that creates a local refinement is applied on these areas, Fig. 7. Meshing information and quality criteria are shown at Table II.

<table>
<thead>
<tr>
<th>Table II: Ship model specifications</th>
</tr>
</thead>
<tbody>
<tr>
<td>Global element length</td>
</tr>
<tr>
<td>Local element length</td>
</tr>
<tr>
<td>Number of shell elements</td>
</tr>
<tr>
<td>Number of beam elements</td>
</tr>
<tr>
<td>Quality Criteria</td>
</tr>
<tr>
<td>Skewness (Nastran)</td>
</tr>
<tr>
<td>Aspect ratio (Nastran)</td>
</tr>
</tbody>
</table>
The first step of the analysis is to apply still water loading conditions, in order to obtain the proper weight distribution for the analysis that will follow. The ship is at rest in a state of equilibrium between its own weight and cargo payload and the resultant buoyancy. The weight is calculated in ANSA from the mesh net, the properties of the shell and material characteristics, while the payload is applied as pressure inside the holds, using cargo's density. In this case all holds are considered full. The buoyancy is applied as hydrostatic pressure in the elements bellow waterline and varies linearly with water depth, Fig. 8.

The mass of auxiliary structures that doesn’t contribute to ships strength are added by three sets of non-structural mass at the bow, stern and middle respectively. This amount of mass is distributed among the sets in such portion to achieve balance without having trim angle in the still water load case. This is achieved by moving the center of gravity in such a position in relation to the center of buoyancy that the resultant force produces zero moments along the ships length and width. This procedure can be performed automatically with a special tool of ANSA that adds mass to specified areas of the model in order to achieve a target total mass and a target center of gravity. The areas where the added mass is distributed are shown at Fig. 9.
The next step of the analysis definition is the application of sagging loading condition. An 8 m height trochoidal wave is used and the balanced position is calculated by iteratively adjusting the draught and trim until the resultant net force and moment of the ship is ideally zero. The definition of the wave, the balance and buoyancy are again calculated by a special tool developed using the ANSA Scripting Language. Buoyancy is applied as PLOAD4 on hull elements, Fig. 10.

3.2. Analysis results

The model is solved with Nastran and the run lasted about 1 hour and 30 minutes in a dual core processor while the results are presented in µETA post-processor. The maximum developed Von Misses stresses in the cargo hold area are about 140 MPa, lower than yield stress of steel. High stress concentrations occur at the third and forth hatch coaming end brackets but the scantlings of the ship can be considered adequate since there appear no critical stresses. The standardize statistics tool can give an overview of the hull behavior while the areas of interest can be easily identified and displayed using annotations and iso-functions. Results from µETA are shown on Figs. 11 and 12.
4. Ships collision

The second case study tested in this paper is the collision between two identical bulk carriers. Collision mechanics are usually separated into external dynamics and internal mechanics. External dynamics deals with the rigid body global motion of the vessels and the effect of the surrounding water, while the internal mechanics is concerned with the structural failure response. In this case study only the internal mechanics is taken into account. To observe the behavior of both the holds and the bow, both ships are modeled as deformable bodies. Collision angle is chosen to be 90° and strike location at amidships. The initial velocity of the striking ship is 6 knots while the struck ship is standstill. Both vessels are loaded.

4.1. Model set-up

A part of both models that are not significant for the analysis are substituted by rigid bodies entities (CONSTRAINED_NODAL_RIGID_BODY) which contain the mass and inertia of the substituted model parts. The “Rigidize” treatment that is automatically applied within ANSA, minimizes significantly the calculation time and simplifies the model. At the striking model the middle and aft parts are substituted from rigid bodies while at the struck both fore and aft parts are substituted by such bodies, Fig. 13. For this case study the LS-DYNA explicit solver will be used.

The meshing parameters and quality criteria for the collision analysis are prescribed at the ANSA Batch Meshing Tool. Both models are meshed with mean element length of 0.13 m. However at the collision area of both models fine mesh of 0.06 m is applied to ensure accurate results. Transition areas are also provided to connect coarse and fine mesh. Since the beams that represent the stiffeners are pasted on the shell elements, a re-meshing action on the shells updates the beams definition. This is an automatic process in ANSA which redefines any entity is attached on shells after their re-meshing. At the area of local refinement new beams are created. This technique eliminates the need of redefining the beams in every change of the model mesh, Fig. 14.
Velocity and gravity are applied to the striking model. The balance of the striking model is defined by the applied gravity and a rigid wall at the model bottom. Thus, the model slides on the rigid wall until hits the target. The struck ship is restrained in all degrees of freedom by constraining the nodes movement of the outer hull on the side that is not going to be hit. Contact entities of type AUTOMATIC_SURFACE_TO_SURFACE are defined between the two models and AUTOMATIC_SINGLE_SURFACE for each of the models. The last ones are defined to eliminate any penetration between the shells of the model itself during the collision. All the above entities are prescribed as sequential steps at the Task Manager. These actions are realized when the Task Manager is invoked. Meshing information and quality criteria are shown at table III. The two models are assembled in one FE model as shown at Fig. 15.

<table>
<thead>
<tr>
<th>Table III: Ship model specifications</th>
</tr>
</thead>
<tbody>
<tr>
<td>Global element length</td>
</tr>
<tr>
<td>Local element length</td>
</tr>
<tr>
<td><strong>Striking ship</strong></td>
</tr>
<tr>
<td>Number of shell elements</td>
</tr>
<tr>
<td>Number of beam elements</td>
</tr>
<tr>
<td><strong>Struck ship</strong></td>
</tr>
<tr>
<td>Number of shell elements</td>
</tr>
<tr>
<td>Number of beam elements</td>
</tr>
<tr>
<td><strong>Quality Criteria</strong></td>
</tr>
<tr>
<td>Skewness (Nastran)</td>
</tr>
<tr>
<td>Aspect ratio (Nastran)</td>
</tr>
<tr>
<td>Crash time step</td>
</tr>
</tbody>
</table>

Fig.15: The assembly model

4.2. Analysis results

If the external dynamics had been taken into consideration, then the impact energy would have been consumed in both the structural deformation and water resistance. In this study, in which only the internal mechanics are taken into account, the whole impact energy is absorbed by the deformable vessels. As a result, this load case can be considered as the worst case scenario. The results though showed that only the outer hull is penetrated by the bow's bulb while bow's upper edge is severely damaged. The spring back effect started 2 s after the impact moment. Striking force, velocity and damage are shown at Figs. 16 and 17. The time needed to complete a 4 s simulation, was approximate 24 h using a cluster of 16 processors.
5. CFD analysis

The third case study is a CFD analysis of the same cargo ship. The computational domain of this simulation consisted of the cargo vessel geometry and an open ocean area. The vessel geometry has been meshed with a variable size surface mesh with additional refinement on the hull region, Fig. 18.
The water level has been modeled with a separation surface between the two fluid regions, running across the hull at a mid-distance. The boundary layer, generated on the vessel and the fluid surface separation, consist of five layers of prisms, generated in aspect mode with a growth factor of 1.05 and first height of 10 mm. A hexa hybrid mesh has been generated on both fluid domains with a total size of nearly 11 million hexas and polyhedrals, Fig. 19. The whole meshing process of the vessel and the fluid domains is elaborated through the ANSA Batch Meshing Tool.

Local refinement is applied at areas of interest such as the bulbous bow and the propeller and rudder. Special entities of ANSA, the SIZE BOXES are defined in these areas. Fluid domains that reside inside the SIZE BOXES volume are meshed with a prescribed element length. SIZE BOXES can have any shape in order to simulate any refinement area, Fig. 20.

6. Optimization

The model behavior according to the above defined analyses can be improved by defining an optimization process which can alter model shape and properties. Such properties can be parts thickness, stiffeners cross sections, material properties, etc. The shaping of the model is achieved through the ANSA Morphing Tool. This tool provides several ways of morphing FE or geometrical models while ensures smooth, predictable and controllable results. Since the ANSA Morphing Tool is able to control a ready to run FE Model, there is no need of re-defining any entities like mesh, loads, boundary conditions, etc. The model is ready to run just after shaping. Thus, the defined shaping or parameterization process can be easily coupled with an optimizer and run in batch mode within the optimization loop.

Morphing in ANSA is performed by special entities, the Morphing Boxes. These entities are created around the area of the model to be modified. As the shape of the Morphing Boxes can be modified in several ways, the model surrounded by them follows the modification and the shaping takes place. Design variables can drive parametrically the morphing process. This enables the connection of any shaping action to an optimization process through the design variables. In this example morphing is performed on the bulbous bow of the ship. A design variable drives the position of the bow along the Z axis. In this CFD model the Morphing Boxes control the ship geometry, the surrounding fluid and layers. After morphing there is no need to redefine the mesh so the model is ready to run to the solver, Fig. 21.
Model behavior and validity check for different shapes can be performed by the DOE functionality provided in ANSA. A Full Factorial algorithm creates several experiments upon user’s request. In this example three design variables are defined to alter the bulbous bow shape. A DOE study runs to define several experiments as shown at Figs. 22 and 23.

7. Conclusions

ANSA is able to set up efficiently several CAE analyses for different disciplines and load cases. Process organization and standardization is possible using the Task Manager tool. Great scatter can be achieved in simulation results caused by slight changes in model parameters. Special tools that needed in marine design have been developed using ANSA Scripting Language. Applications like wave creation, mass balance, vessel balance on a wave profile, buoyancy calculation and cargo loading can be automated using these tools. µETA is a versatile post-processor which provides sophisticated tools for results reporting and evaluating. ANSA also provides powerful model shaping and a versatile optimization set-up tool which is able to automate the whole definition of the CAE model. The Optimization Task also provides easy coupling with most of the commercial optimizers.
References


Abstract

The UK MoD has in collaboration with QinetiQ GRC, developed a software tool to facilitate initial options and pricing investigations for a bespoke Class of Naval Auxiliary ships. Two commercially available software packages, Paramarine and modeFRONTIER, have since been used to enhance the tool to conduct an exploration of the potential design solution space at a pre-concept stage for a further ship Class. Although described in this paper in the context of Naval Auxiliary ship design, the tool has been developed in a generic manner with the intention that other ship types can be investigated.

1. Introduction

The UK Ministry of Defence (MoD) is replacing a significant proportion of its Royal Fleet Auxiliary (RFA) ships as they near the end of their service life. The new ships represent a considerable design challenge due to their complex operational requirements. They will be more capable than their predecessors, and will incorporate more demanding safety regulations and modern standards, which when applied to areas such as munitions holds and accommodation spaces, drive up the overall size and cost of the ships. Currently there are no existing designs that satisfy the operational requirements of the new ships, nor the mix of commercial and military standards and equipment to be incorporated. It is therefore anticipated that the new ships will be of bespoke design and build.

In procuring the new ships, the MoD is required to demonstrate that the design requirements being put out to tender are technically feasible and represent good value for money. Equally important is the ability for the MoD to engage in dialogue with partner shipyards during the open competition process as an ‘intelligent customer’ with the ability to understand the implications of changes to the requirement, design, or programme within limited timescales.

With the exception of the new Class of Tankers which will be entering service in the next few years, the requirement for the follow-on RFA ships is in its infancy. It is likely that the overall capability and specific features of these later ships will have to adapt to changes in associated ship procurement programmes and increasing budgetary constraints. The tool described in this paper has been developed with the aforementioned factors in mind to provide an approximate initial view of the potential range of designs and their associated cost drivers.

2. Requirements of the design tool

The concept design tool produced previously for the MoD in exploring the new Fleet Tanker Class from a cost vs. capability perspective Cooper et al. (2007) required further development to provide the speed of use, flexibility, and scope of investigation required for the assessment of the follow-on RFA ships, which are far more complex and ambiguous in their overall configuration. The following improvements to the existing tool were subsequently identified as being necessary:

- Ability to investigate radically different ship arrangements
- Ability to explore a greater proportion of the design space
- Increased automation of the design process to reduce design timescales
- Simplicity of use
- Flexibility of use
- Ability to quickly replicate designs and trace assumptions
- Ability to add and refresh data and assumptions held within the tool
The overarching aims of the new tool were defined as follows:

- To explore the design space for a given set of capabilities
- To identify the implications of different capabilities on unit production cost and through-life cost
- To enable informed design decisions to be made during subsequent, more detailed concept design development

Of the requirements listed above, the anticipated need to automate a greater proportion of the design process represented the most challenging aspect of the tool’s development, but was considered an essential step in enabling the tool to explore the large number of designs required, as the timescales associated with the previous, partially-automated approach, were not conducive to the wider scope of exploration required.

The design tool produced previously for the MoD was only automated in the aspects of the tool that lent themselves particularly well to automation. The designer was still required to make the important and often highly involved decisions on overall ship topology, architecture, propulsion arrangement etc. Unfortunately, the interaction of the designer in this process resulted in the design exploration becoming prohibitively slow, thus restricting the large number of potential design options that could be investigated. The complexities of automation in relation to wider exploration in initial ship design are summarised by Andrews et al. (2010) where the applicability of open design processes, referred to as ‘glass box’ methods for which the designer has full control of the design process and a clear understanding of the design drivers, are weighed up against ‘black box’ approaches where the design process is controlled by a predefined set of rules. It can be concluded that the requirements of a particular design tool govern the choice of design method, and the consequence of the need to perform a wider design exploration over a limited timescale is the need for a greater degree of automation. This presented a challenge in that it was clear that an automated approach was necessary to perform the wide range of design exploration necessary in the limited timescales available, while at the same time maintaining the required ability to investigate radically different ship arrangements, a characteristic not conducive to ‘black box’ automated design methods.

3. Adopted design process

3.1 Consideration of layout

The adopted design process described in this paper utilises an embedded layout module which, at a basic level, develops a 2D profile “sketch” of the ship arrangement based on the main features of the design (cargo spaces, flight decks, machinery rooms, hull/superstructure interactions such as uptakes, etc.). The configuration of the features included in the profile arrangement are controlled by logic defined by the designer, enabling a range of variations to the arrangement to take place, the type and extent of which are pre-determined by the designer. The 2D profile is later converted into a 3D model in Paramarine (www.grc.qinetiq.com) based on predefined assumptions on the upper deck and below deck beam occupied by the various features, Figs. 1 and 2. The layout module is the main component of the tool that the designer is involved in defining at the start of the process, thus providing significant control over this key part of the design process and differentiating the tool from an entirely hard-wired, “black box” type approach. It is appreciated that the adopted approach to addressing overall ship configuration does not represent an ideal solution, but is considered to be a significant step forward from entirely non-architectural approaches, and is well suited to the pre-concept exploration stage where greater focus is placed on design timescales and investigation of as wide a range of design options as possible. The integration of the Paramarine 3D model at this stage of the design process also significantly improves the accuracy of the tool due to the geometric information generated and the software’s ability to then assess a wide range of performance aspects.
Fig. 1: Basic 2D profiles (from top: Tanker, Sea Basing Ship, Fleet Support Ship) generated in MS Excel showing predefined ship size and layout drivers (highlighted areas represent excess space which can be subsequently minimised as a design objective)

Fig. 2: Basic 3D model (tanker) generated in Paramarine from 2D profile

3.2 Achieving a design balance with an automated design process

Initial development of the tool focused on the use of a simple, automated design process incorporating iterative loops to feed-back design information and re-configure the design in the case of design constraints not being satisfied; the intention being to reduce the number of unbalanced designs generated, Fig. 3. The design process was initiated by defining a set of capability variables (Range, Speed, Payload, etc.) within a ‘Master Input Variables’ module. These variables fed into the various interconnected stages of the design process to generate a design based on the predefined logic incorporated into the tool. Use of the tool revealed that the highly interdependent logic within the process made it difficult to trace the assumptions and steps taken in the development of each design.
Fig. 3: Iterative design process deemed unsuitable for the automated design approach

The major drawback however, was that certain designs were unable to converge on a balanced solution, forcing the process into infinite calculation loops as the system repeatedly converged and diverged. This characteristic proved to be problematic in performing an automated design exploration as the software controlling the process was unable to move onto the next design unless a lengthy time-out option was specified, negating any benefit gained by using this approach.

Fig. 4: Non-iterative design process adopted to generate individual ship designs

For the automated design process described in this paper, a non-iterative approach was considered to be more appropriate, Fig. 4. The issue of changing the various input variables to generate balanced designs has been achieved by using modeFRONTIER (www.esteco.com) to control the inputs to the design process. This software also checks each design generated against the predefined constraints in order to ascertain whether a design balance has been reached. The result is a large number of designs...
for any one capability, of which typically, a significant proportion could be unbalanced. The main
benefit, however, is that a design solution (whether balanced or unbalanced) is always reached, and
because non-convergent designs are prohibited, the design software is less prone to instability. Where
a number of balanced designs are generated for any one capability, designs that best satisfy the
objective criteria (e.g. lowest cost) can be identified.

To move from an iterative design process to a non-iterative process, some significant changes to the
overall flow of design information had to be made. Ship speed, for example, was previously an input
which generated an iterative loop in the process (ship speed required > resistance calculation >
installed power calculation > machinery and fuel weight > displacement > resistance calculation). To
remove this loop, ship speed was removed from the inputs and replaced by an initial installed power
estimate (installed power estimate > machinery and fuel weight > displacement > resistance
calculation > ship speed check). The final ship speed check is performed at the end of the process to
assess whether the required ship speed had been achieved, Fig. 4. Theoretically, the two approaches
should produce the same solution, however the non-iterative method typically requires a large number
of designs to be generated before arriving at the solution which best satisfies the objective criteria.

3.3 Exploring the potential design solution space

As a standalone system, the non-iterative process described previously is useful for the rapid
generation of individual, low definition concept designs, by manually changing inputs into the design
process until the objective criteria have been satisfied and a design balance is reached. To enable
exploration of the wider design solution space for a given range of capabilities, modeFRONTIER has
been used to control the inputs to the process with the view to satisfying a predefined single or multi-
objective aim, such as to reduce build cost or to minimise the bunker fuel requirement, or both.

ModeFRONTIER allows the input variables to the design process, Fig. 4, to be systematically varied.
This is achieved by firstly defining a Design of Experiments (DOE) which produces a spread of input
data then used to generate an initial ‘population’ of designs. ModeFRONTIER then filters the designs
generated by considering the design constraints and objective criteria specified. Designs found to
satisfy the specified design constraints are used to generate further designs using genetic algorithms
within modeFRONTIER, which seek to develop improved designs based on the objective criteria
specified by the designer. Designs found to fail the design constraints are rejected.

The designer is responsible for selection of the design variables to be explored. Input variables for this
example were split into two categories, capability variables, and design variables. It is important to
differentiate between the two when it comes to setting up the design space exploration in
modeFRONTIER as explained later. Output variables are also split into two categories, design
constraints, and design objectives. Examples of the different types of input and output variables are
provided below:

<table>
<thead>
<tr>
<th>Input Variables</th>
<th>Capability Variables</th>
<th>Design Variables</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Cargo Volume</td>
<td>Length to Beam Ratio</td>
</tr>
<tr>
<td></td>
<td>Speed</td>
<td>S/Structure Volume Fraction</td>
</tr>
<tr>
<td></td>
<td>Range</td>
<td>Cargo Block Length</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Output Variables</th>
<th>Design Constraints</th>
<th>Design Objectives</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Speed Req = Speed Achieved</td>
<td>Minimise Cost</td>
</tr>
<tr>
<td></td>
<td>Stability Exceeds Standard</td>
<td>Minimise Bunker Fuel</td>
</tr>
<tr>
<td></td>
<td>Volume Req &lt;= Volume Provided</td>
<td>Minimise Unusable Space</td>
</tr>
</tbody>
</table>

351
For the exploration work conducted on behalf of the MoD, 4 capability variables and 15 design variables were specified. To enable modeFRONTIER to identify and remove unbalanced designs from the exploration, 10 design constraints considered pertinent to the analysis were specified. To enable the genetic algorithms within modeFRONTIER to generate subsequent, improved designs, from the designs deemed to be balanced, 2 design objectives were specified (minimise build cost and minimise bunker fuel). Tolerances on the design constraints were set to allow designs coming close to the criteria to be included. It was considered that the application of exact criteria at this stage could greatly compromise and restrict the design exploration process due to the approximate nature of the designs generated. Likewise, the application of too many constraints, such as those relating to detailed aspects of performance, was considered to be ineffective due to such assessments requiring detailed analysis using information either not yet available, or not fully developed at this stage.

Having specified the input and output variables, the next step is to specify the range and increment over which the input variables are to be changed by modeFRONTIER. Length to Beam Ratio, for example, could be varied in increments of 0.5, between an upper limit of 8 and a lower limit of 6. This process is performed for both the capability variables and the design variables.

A similar process is performed for the output variables. Design constraints are set-up in modeFRONTIER by specifying the maximum, minimum, target, or acceptable range of values for the output data. Design objectives are then set-up to monitor the key outputs of the design process and identify designs that pass the specified design constraints and offer the best solution in terms of the specified objectives.

For the exploration described, each design variable had typically anywhere between 2 and 20 possible settings. Performing a full factorial investigation for the DOE (i.e. exploring every possible permutation of inputs) would therefore be a prohibitively lengthy process. There are a number of ‘space fillers’ within modeFRONTIER to assist the designer in generating an initial population of designs. For the work described, a Sobol distribution of 10 designs was used in combination with a random distribution of 5 designs. The resulting initial design permutations were used in the DOE for each capability option. ModeFRONTIER was then set-up to produce 2 further designs generations from the initial DOE population using the genetic algorithms within software in accordance with the specified design objectives to give a total of 45 designs for each capability option (108 capability options specified giving a total of 4860 designs). For the purposes of the work described, the NSGAII algorithm was used. The total number of designs could be increased to improve coverage, however for the 2-5 minute timescale per design, the selected total was appropriate to the overall timescale available. The tool has been set-up to be configurable for any number of processors, providing flexibility in terms of the IT resources available to different projects. For the work described, the exploration was performed using a single desktop computer with modeFRONTIER and Paramarine installed.

For very simple design space explorations, it is possible to conduct the exploration process using a single instance (open program) of modeFRONTIER. However, for the number of capability and design variables used for the study described, it was found that modeFRONTIER would tend to focus on combinations of capability and design variables which repeatedly satisfied the design constraints, and hence produce subsequent generations of designs based on these permutations alone, resulting in an uneven and inconsistent exploration of the potential design solution space. It was deduced that a more diverse exploration of each of the capability options was necessary, where the design variables could be varied consistently for each combination of capability variables. This was achieved through the use of a second instance of modeFRONTIER (hereafter termed the “outer loop”) to individually send capability variables to the first instance of modeFRONTIER (hereafter termed the “inner loop”) responsible for exploration of the design variables. In this configuration, modeFRONTIER is operating in what is termed “batch mode”, whereby an separate DOE and subsequent “evolution” of designs is produced for each capability option based on the specified range of design variables, constraints, and objectives. The inner loop is therefore responsible for exploration of the design (non-capability) variables and the outer loop is responsible for sequentially sending every possible
permutation of the capability options (required cargo volume, required speed etc.) to the inner loop, Figs.5 to 7. This method ensures every capability option is explored in a consistent manner, and that the same number of variations is explored for each option, thus providing a consistent and even coverage of the potential design solution space. Tests of the system in batch mode have also demonstrated that modeFRONTIER can more readily identify trends in the output data and therefore generate subsequent designs that provide further benefit in terms of the design objectives.
4. Decision making tools

A large quantity of data is available to the designer once modeFRONTIER and Paramarine have completed a design generation sequence. The use of appropriate decision making tools is therefore vital in attempting to understand the output data. One particular tool available within modeFRONTIER enables any number of the input and output variables to be displayed on a multi-axis graph. Individual designs are displayed as individual lines passing through each axis. Sliders upon each axis allow the designer to constrain the range of data displayed on each axis. For example, to identify the lowest cost designs, the slider on the cost axis is moved to the bottom of the scale. As the slider is moved, the designs falling outside the range of the slider are removed (lines disappear), leaving only the lowest cost designs and the associated variables which generate each design. This is particularly useful as an interactive tool during design and requirements management meetings.

To enable investigation of trends in the output data, QinetiQ GRC have developed an Excel spreadsheet into which all the output data from the Paramarine/modeFRONTIER process can be entered (design data is held within modeFRONTIER as a single, large table showing all the inputs and outputs of the design generation sequence). The spreadsheet generates a graph, Fig. 8, displaying the through life cost and unit production cost (average and lowest) of each capability option explored. Capability options can be re-ordered on the graph to help identify trends in the data.
Fig.8: Trend analysis of 108 capability options explored showing through life cost and unit production cost of permutations offering lowest cost.

5. Validation and verification

Validation and Verification (V&V) of the Paramarine / Excel (design) component of the tool has been partially completed. Design decisions and assumptions are not made within modeFRONTIER therefore this component of the tool was excluded from the V&V process. Validation has here been defined as the ability of the tool to correctly reproduce known results, and verification as the confirmation from an independent expert that the tool has been correctly set up. A separate validation study has been performed on the design component of the tool Ministry of Defence (2008) and has demonstrated that the tool is capable of reproducing existing ship designs of similar type (AO, and AOR Class Auxiliaries) to a satisfactory degree of accuracy, with overall dimensions being within 6% and displacements within 8% for the ship comparisons made. Verification of the design component of the tool is yet to be carried out, this requiring considerable time and expert independent input due to the size and complexity of the tool.

The cost component of the tool has been verified by an independent overseer with considerable design costing experience, and was deemed appropriate for comparative cost purposes. A senior member of the programmes costing team used by the MoD has also scrutinised the model and the material and labour rates used within (rates have since been updated). The cost component of the model has not yet been validated. However, this would be a simple exercise for an independent expert due to the simplicity of the model.

6. Conclusions

The design space exploration performed using the tool has enabled a significant proportion of the potential design solution space to be explored at a pre-concept level for the range of capability options specified. The tool has identified the approximate impact of different platform capabilities on estimated through-life cost and unit production cost by means of the rudimentary cost model incorporated into the tool. This work has enabled subsequent, more detailed designs, to be generated at a greater level of fidelity with an improved understanding of how the different capabilities and design variables influence cost. In addition, due to the extensive range of Naval Architectural
assessment tools and Early Stage Design features within Paramarine, the subsequent design work can be progressed in the same software using the basic designs generated by the design space exploration.

An automated, non-iterative design process was found to provide a stable and effective approach to the rapid generation of large numbers of ship design variations (4000+) and the embedded, user defined layout module, enabled the generation of believable ship configurations, adding confidence to the designs produced by the tool and a further degree of accuracy to the results. The use of “inner” and “outer” loops in the design space exploration workflow has facilitated a more diverse exploration of the potential design solution space by forcing the sequential exploration of the different capability options.

The developments made to the tool since its initial application have facilitated the overarching aims and improvements required of the tool as listed in Section 2. These are summarised again below, with notes describing the relevant features incorporated into the tool:

Required improvements implemented into the existing tool:

- **Ability to investigate radically different ship arrangements**
  A layout module defining a basic 2D profile “sketch” of the ship is set-up by the designer for each individual ship type explored to define the layout logic and layout variations to be incorporated into the design space exploration. This ensures that the layout of the design influences the overall design process to a greater extent than for a purely numerical design process.

- **Ability to explore a greater proportion of the design space**
  modeFRONTIER has been used to perform individual explorations of the design variables in Paramarine corresponding to the different capability options under investigation. This ensures that a greater proportion of the design space is explored, and enables the exploration to be performed in an even and consistent manner.

- **Increased automation of the design process to reduce design timescales**
  The entire design process has been automated, enabling each design generated to be produced in less than 5 minutes and for thousands of design options to be generated over the course of a few days. The designer is responsible for setting up the design process and formulating the assumptions within it. It can be argued therefore that the tool enables a greater level of control of the design process than that of a hard wired “black box” process. The design process is also flexible to the computational facilities available meaning that additional hardware and software can be used to further reduce timescales if deemed necessary.

- **Simplicity of use**
  Separate “modules” have been used to ensure correct “flow” of data through the design process. Each module has been set-up in a straightforward and transparent manner. The move away from an iterative design process, containing many interconnected variables and feedback loops, has further simplified the tool. Although efforts have been made to simplify the various components of the tool, it is considered that a competent Naval Architect with a firm understanding of design methods would be required to update and operate the tool.

- **Flexibility of use**
  The tool has been set-up for the MoD to explore Naval Auxiliary ship types, however, the way in which it has been set-up is amenable to any ship type. The Excel “modules” act as a knowledge base, using only the information relevant to the design being investigated. The layout module can be set-up to explore any ship or marine platform topology.

- **Ability to quickly replicate designs and trace assumptions**
  Due to the parametric nature of Paramarine, Excel, and the automated design approach used, it is possible to rapidly replicate designs and investigate all the assumptions used in their development by replaying the inputs to a particular design as a macro. All assumptions are held within the Paramarine/Excel modules making auditing of the tool relatively straightforward.
- **Ability to add and refresh data and assumptions held within the tool**

  The “knowledge base” of information held within the Excel spreadsheets has been arranged such that additional information on weight, space, equipment, layout, and cost can be easily incorporated into the existing tables. Likewise, assumptions on weight scaling methods, layout logic, etc. can be updated as required.

The overarching aims of the new tool were summarised as follows:

- **To explore the design space for a given set of capabilities**

  It has been possible to explore a considerable proportion of the design space for the capability options specified. Due to the automated nature of the design process used, exploration of the entire design space is not deemed possible for complex design tasks due to the quantity of information and assumptions involved.

- **To identify the implications of different capabilities on build and through-life cost**

  Of the capability options investigated, it was possible to identify the capability with the greatest influence on through life cost and unit production cost, and establish the relative cost implications of the other capabilities. The simplicity of the weight based cost model made it difficult to determine the true cost implications of certain capability options that required a more detailed cost assessment using additional cost metrics.

- **To enable informed design decisions to be made during subsequent, more detailed concept design development**

  The design space exploration enabled subsequent concept design studies to take place with an improved understanding of the cost drivers associated with the ship types under development.

The design space exploration tool described has been developed to help identify key cost and technical design drivers at a basic level. It has not been developed to provide detailed unit production costs, through life costs, or costs to be used for budgeting purposes. The automated process contains a wide range of pre-defined assumptions and logic, therefore designs and design data generated by the tool will feature a considerable degree of uncertainty and are unlikely to represent the ideal solution to any given set of capabilities. A further drawback of the automated design process is that design opportunities which often present themselves to the designer during a manually instigated, architecturally driven design process, will be missed. The human element of ship design allows creative decisions and balanced judgements to be made as the design develops. The tool described is not intended to, and cannot, replace subsequent concept design investigations typically performed during the preliminary design stages. It is recommended, however, that information from subsequent concept designs be fed back into the tool to update the assumptions within and to validate its output.

The scope of the design space exploration tool described is limited by the ability of the designer to incorporate the necessary decisions and assumptions into the process. This can represent a considerable amount of work in itself, and the timescale required for this preparatory stage must be factored into use of the tool. The exploration work revealed that there are many types of investigation that are poorly suited, or indeed impossible to explore using the automated design process, either due to complexity of set-up, immaturity of required information, or lack of resolution in the cost model. Logically, the weight based cost estimation approach used is capable only of establishing useful cost data if the options under investigation are well matched to the cost model metrics. As with any tool of this type, its application to certain investigations must therefore be thoroughly considered prior to its use, and additional resolution must be added to the tool where necessary to ensure useful output.

### 7. Further development

The tool is still in a developmental stage. A large proportion of the work invested in the tool was in establishing the connectivity between the various pieces of hardware and software, primarily the development of links between Paramarine and Excel. There remains considerable scope for further development of the tool, with a greater focus its design elements, and the fundamental aim of
increasing the proportion of the design space that can confidently be explored. In the first instance, it is anticipated that additional functionality within the cost component of the tool could provide the greatest level of enhancement, allowing investigation of a greater number of capability options, and improving the accuracy of the costs generated.

Although developed for the MoD, the tool described in this paper is intended to be flexible to different projects and ship types. Further development is however necessary if it is to have utility over a wider range of applications. The next stage of development should see the tool stripped down to its primary components, namely generic “inner” and “outer” loop design space exploration workflows using an appropriate multi-objective optimisation and design environment (currently modeFRONTIER), and a generic design process containing the required modules and dataflow sequence (Paramarine/Excel), but otherwise undefined, enabling the designer to set-up the design process and design space exploration in accordance with the requirements of the study in hand.

Experience gained in the use of the tool has shown that computational time varies between approximately 2-5 minutes per design. Based on the calculations within Excel being practically instantaneous, and the combined assessments within Paramarine taking typically less than 1 minute, the actual time taken to generate individual designs should be considerably less than the timescales seen. Further investigation of this issue is necessary to improve the efficiency of the tool and therefore increase the number of designs that can be explored.

Acknowledgements

Acknowledgements are extended to Seth Cooper of NAVSEA for his major contribution to the initial formulation of the design process and management of the tool’s initial development. The author would also like to thank the members of the Naval Architecture and Software Teams within QinetiQ GRC for their contribution to the development of the tool, as well as the UK Ministry of Defence and EnginSoft UK for their input during its development:

References

ANDREWS, D.J.; McDONALD, T.P.; PAWLING, R.G. (2010), Combining the Design Building Block and Library Based Approaches to improve Exploration during Initial Design, 9th Int. Conf. on Computer and IT Applications in the Maritime Industries (COMPIT), Gubbio.


MINISTRY OF DEFENCE (2008), AfSup CDT Design Process Validation, MoD DE&S Afloat Support Directorate
Non-Disruptive Development of a Next-Generation CAD Application Program

Bastiaan Veelo, SARC B.V., Trondheim/Norway, Bastiaan@SARC.nl
Herbert Koelman, SARC B.V., Bussum/Netherlands, H.J.Koelman@SARC.nl

Abstract

There is a limit to how far an application program can evolve in incremental steps. At some point in

 time, progression requires radical changes: a new generation that parts with the limitations of its

 legacy. In our case, due to richness in features, a complete rewrite of our hull design software would

 lead to an unattractively long down-time, which is why we have pursued a more efficient allocation

 of our programming resources. This is the report of an approach in which we keep both the production

 version and the development version fully functional within the same executable, providing a non-

 disruptive transition from one application generation to the next, while building on proven

 foundations.

1. Introduction

Almost two decades ago, SARC started research and development of a new computer method and

 application program for the design and fairing of ship hulls, called Fairway. Fairway would remedy

 the fundamental disadvantages and limitations of NURBS surfaces, a technique that is still at the core

 of most CAD software today, Koelman (2003). Since its release on the maritime market in 1995,

 Fairway has been steadily improved and extended, so that it currently has a diverse feature set

 including

- User-defined geometric master-slave relations between curves
- Projection of spatial curves onto the defined model
- Hydrostatic analysis
- Target and model sectional area curves (SAC)
- Shape transformations
- Developable surfaces
- Shell plate expansion
- Multiple solids and model variants
- Boolean operations on solids
- Surface quality assessment methods.

Additionally, external modules have been developed that build on Fairway technology, such as

- A hull server for the interpolation of arbitrary hull lines upon the request of another process
  (RPC fashion)
- An import module that finds matching topological relations in a curve cloud for the
  conversion of loose ship lines into a solid model
- PhotoShip, an application for the reconstruction of 3D digital models of physical objects
  based on photographs.

At the turn of the millennium a new research project started in Trondheim to address another common

 problem in CAD: the fact that the extent, or scale of influence, of geometric manipulation decreases

 with increasing model detail. Because of this, the cost of producing variations on a design generally

 increases rapidly as the design progresses. The project produced an experimental version of Fairway

 demonstrating that the principles of spatial deformation can be used to achieve smooth shape

 variations of arbitrary extent, on models of any detail, Veelo (2004a,b). However, we found that a

 commercial implementation imposed requirements on user interface, on graphical performance and on
internal code structure that Fairway did not meet at that time. This extension, however useful, exceeded the possibilities of incremental code evolution. We had hit the code barrier.

2. Breaking the code barrier

In 2006, the time was found right\(^1\) for a radical modernization of Fairway, and we considered our options. One possibility is to start from scratch and implement the next generation from the ground up. The advantage is that limiting legacy code can be eliminated and that you are free to use any modern programming language, library and technology. The disadvantage is that you will spend a long time implementing data structures, algorithms and features that already exist in the proven implementation of the previous generation. In addition comes time needed for testing and debugging. It is important to note that the new application is not production-ready until its feature set is on a par with the previous generation. The approach is known to work, even for large projects, as illustrated by the rewrite of CATIA V5 in 1999 by Dassault Systèmes, which could afford to put one thousand programmers to the job\(^2\).

We do not have that kind of resources, and with the variety of functionality already present in the production version of Fairway, we would not be able to rewrite all of it from scratch without the result being outdated by the time we would be done. We needed a different option. Fortunately, we were not unprepared.

2.1. Anticipatory measures

In the past, strategic decisions had been made at SARC that were to offer flexibility and security regarding future technological developments and economical changes. These were

1. the use of a standardized programming language,
2. a generic menu library (“genmenu”), and
3. an abstracting Graphical User Interface (GUI) library (“Ywin”).

2.1.1. Language and compiler

Since 1980, Pascal has been the predominant programming language in use at SARC. Pascal has a number of advantages for engineering purposes over the contemporary alternatives Fortran and C:

- It enforces structured programming. This leads to software that is better structured, and hence may contain less bugs and is easier to maintain.
- It has a relatively small number of statements, yet is a very powerful language. So for people not 100% of the time engaged in programming it is easier to comprehend than one of the other languages. Moreover, the language does not invite to whizkiddery.
- The language is straightforward, also from a compiler point of view, which means it is quick to compile to efficient binaries.

An early phenomenon regarding Pascal was that different compiler vendors extended the language in different ways, leading to a variety of Pascal dialects. In 1993 the International Organization for Standardization standardized Extended Pascal (ISO 10206), and rather than using a non-standard dialect of one specific vendor and thereby creating a dependency on its financial well-being, SARC opted for standard Extended Pascal. Prospero Software was the first to implement that standard, and its compiler still serves its purpose very well. Even so, Extended Pascal never enjoyed the hype and popularity we have seen in some other languages, which is why we have to turn to another language in our selection of third-party support-libraries, as you will see in a moment.

---

1 As a matter of fact, it happened during a comfortable evening chat at COMPIT’06 in Oegstgeest.
2 According to a CATIA instructor from IBM.
2.1.2. Abstraction

Ywin and genmenu are in-house developed libraries that increase our programming productivity and help to maintain uniformity of user interfaces. They are used across our entire product line, not just Fairway. Ywin is an abstraction to the Microsoft GUI and system libraries. One important role of Ywin is that it shields our programmers from the frustrations of having to deal with the Microsoft win32 API, which we consider to be of great relevance for productivity. Genmenu is a very flexible event-driven library that enables us to quickly create menu structures, input forms and spreadsheet-like alphanumerical interfaces. But the most important aspect in the context of this article is that Ywin and genmenu form a division between our own computer code and the underlying system libraries. In principle, we can migrate all our products to a different supporting system by changing the internals of Ywin and genmenu.

2.2. First approach

Our first approach for crossing the code barrier was to write a drop-in replacement for Ywin and genmenu, using a modern high level GUI library. This would open the road for more easily incorporating the power of a modern GUI into our abstracting libraries, which would benefit all our software at once. It would also help making Fairway ready for its leap forward.

2.2.1. GTK+ and GPC

We selected GTK+ as the GUI library to build on, mainly because it is written in C which is easily callable from Pascal, and because it is object-oriented nonetheless. GTK+ is also used as the basis for several other abstracting libraries, which is encouraging. As we were writing the library (now called YGTK) from the ground up to the specifications of Ywin and genmenu, our first Pascal test programs were small, using the currently available subset of functionality. Initially, we compiled these programs with the GNU Pascal Compiler (gpc), which is a free compiler that supports most of the Extended Pascal standard. Because gpc produces object code of the same format as the C compiler gcc, this allowed us to link the Pascal parts and the C parts into the same executable, and use the same debugger to debug both parts. The project came along well, but when the time came to hook up the first real Pascal application to YGTK, things turned out to be more complicated than expected.

2.2.2. Complications

Firstly, the application pulled along a load of Extended Pascal modules that it links to, and gpc failed to compile that collection. Reasons were that gpc does not cover enough of the Extended Pascal standard, and our modules made use of some local Prospero extensions and platform-specific APIs. Instead of working out all the incompatibilities and the ones that were yet to be discovered, we decided to compile Pascal with Prospero and C with gcc. Due to different object formats these parts had to be separated into different DLLs from now on. This came with its own set of difficulties, including a compiler bug (or two) that had to be resolved and the writing of an ingenious piece of Assembly code to allow C code to call back into a nested Pascal procedure without messing up its call stack.

Also, while GTK+ originates from the GNU project and Linux world, it eventually became clear that it had not managed to flawlessly deal with the peculiarities of the MS Windows platform yet. Most notably were problems with the kerning of letters on screen and output to certain ink-jet printers consisting of just blocks of black. These problems got fixed, but not before we had long dropped the project.

Sadly, it was only after having implemented most of Ywin’s functionality, that we uncovered a new

---

3 http://www.gtk.org/
4 http://www.gnu-pascal.de/
set of inter-dependencies with low level code. In particular it was the hard-copy output that was problematic. Over the years, Ywin had grown beyond a clean abstraction of just GUI code. We had to conclude that it was too difficult to separate plain GUI calls from the rest and that it would take too long to port all low level code. As it was not getting us much closer to our primary goal, being the next generation Fairway, we decided to stop there and consider a different approach.

2.3. Second approach

What we had been successful at was solving the DLL issue, and we were confident that we could tie together anything with a C interface. Our next approach was to shift the borderline between old and new: write a completely new interface for graphical presentation and interaction, using modern solutions, and only keep the proven data structures and routines that operate on them in Pascal. This approach has given us much of the same freedom as we would have had when writing a new application from the ground up, while not having to reinvent the wheel. It has almost been like writing a new application, but with a flying start. The result indeed looks and feels like a new application.

2.3.1. Qt and Coin3D

When shopping for support-libraries we soon landed on the Qt framework for the user interface\(^5\) and on Coin3D for hardware accelerated graphics\(^6\), both written in C++. Qt is very rich in features, has solid support for the MS Windows platform and generally makes programming in C++ more enjoyable. Coin3D is an implementation of Open Inventor, a retained mode high level object-oriented framework that abstracts OpenGL, with an interface to Qt.

Both Qt and Coin3D share a very important feature: they provide full access to their source code, with a license to change it. We have made use of that opportunity on several occasions, both for implementing extended functionality and for fixing problems faster that their support departments. This combines the advantages of building on other people's excellent work with the flexibility and security of having done everything ourselves.

3. Linking old and new

We started by isolating the core Fairway modules in a DLL, but due to entanglement with Ywin and other modules we were quickly heading for the same problems as earlier. The solution was both simple and advantageous: to embed the entire Fairway application in one DLL, including Ywin, genmenu and other modules that it links to. A small C++ main function provides the entry point for the executable and no changes were required to the original code.

The new C++ code lives in a second DLL, and both old and new code can call each other across DLL boundaries, Fig.1. The Qt event loop is easily stopped and started, which makes it possible to switch back and forth between user interfaces without having to leave the application or save the model file. This has the great advantage that every feature of the new interface is usable as soon as it is ready, because for everything that is not implemented yet one simply switches into the other interface. This has allowed us to start testing and using the second generation Fairway long before it is finished or feature-complete. This has produced valuable feedback and allowed early adjustments of the design. There is no down-time, the new generation just grows steadily alongside the preceding generation, until it is mature enough to replace it.

The advantage of shared core code between both the production version and the development version also works the other way. On several occasions we did improve the core Pascal code in support of the new generation, and as both generations are using the exact same core procedures, these improvements were directly available in the production version, without back porting of any kind.

\(^5\) http://qt.nokia.com/
\(^6\) http://www.coin3d.org/
The two user interfaces work quite flexibly in parallel, Fig. 2, which shows the new interface reusing an old genmenu for configuration of position sets, as an interim solution.

4. The next generation

Now that we had found the technical solution for the next generation Fairway, and had tested capabilities of Qt and Coin3D, we were in the fortunate position to gather around the table and make a preliminary design of the new user interface. These were the points on our wish list.

4.1. Wish list

1. Integration of interfaces. Fairway started with a genmenu-based alphanumerical interface during its early development, and was then extended with a custom graphical modelling interface. Once hardware-accelerated graphics became available to the consumer market, Fairway was given an OpenGL render mode. This mode remained limited to presenting static geometry, because . The three interfaces complement each other but cannot be active at the same time. We wanted the new modelling interface to be hardware-accelerated, dynamic, and provide simultaneous graphical and alphanumerical input and feedback in a unified way.

2. Redesign modelling actions. In the old modelling interface the methods for interaction consist of assorted functions, distributed among different menus. These are to be activated in sequence to perform a modelling action, such as connecting network points with a curve. If a function is activated out of sequence, e.g., choosing “Connect Point” without first having activated a curve and requested the display of points, the system complains with a pop-up message. With well over a decade of experience in working with Fairway we wished to redesign the way modelling actions are performed by the user.

3. Undo and redo. Undo-functionality is easily taken for granted if it is there, but implementing it in a CAD system is a nontrivial undertaking due to the amount and complexity of information that needs to be kept track of. Fairway has had an implementation of undo for a long time, which it uses internally in the process of saving models to file. This system, however, is limited to Euler operators and only covers the topology of the model. For user-level undo functionality, geometrical changes need to be kept track of as well.

4. Minimize mouse centimeters and reduce pop-ups. As we are users too, we want working with Fairway to be a smooth process and non-straining experience. Hence we have the general design objective to keep mouse movements and clicks to a minimum. Pop-up boxes, although effective, disrupt the designer in his work and are often associated with a problem. The old interface pops up a message when an action cannot be performed, like when a curve is locked and therefore cannot be removed. We want the new design to minimize the need for pop-ups.
4.2. Design and development

In bullet form, these are some of the characteristics of the new generation Fairway.

- Actions have become a central entity in the user interface. We have gone from a large set of functions to a small set of actions. They have the following properties.
  - Actions have their own user interface panel on which relevant information and tools specific to the context are presented, Fig. 3.
  - Only one action can be active at the same time.
  - The action panel appears when an action is started and moves out of the way when it is closed.
  - Actions typically have three stages:
    a) Starting stage: Prerequisites such as a selection are checked, possibly with a short note of what the user is expected to do.
    b) Configuration stage: The user changes values in the action panel or uses tools for manipulation presented there. Changes are easily visible because the original state of the model is displayed transparently.
    c) Apply or Cancel stage: When Cancel is pressed the action is terminated and changes are reverted. When Apply is pressed, changes are committed to the model and an undo-moment is created. This helps reduce the amount of geometrical changes that need to be tracked, and is easily comprehensible by the user.
  - Some actions can be applied to a selection set, e.g., changing properties of several curves in one go.
  - Action panels, as well as other UI elements, can be detached from the main window and put floating on top of it or even onto another monitor, Fig. 5. This allows the user to organise the interface to personal preferences and maximize the modeling area.
Fig. 3: Close-up of the floating action panel for curve manipulation and a translation dragger. Points are marked to indicate their weight factor (arrow up and down) both in the table and on screen. Next to the editable coordinates is their color-coded deviation from the curve, followed by the (click-through) intersecting curves. On the status bar the dragger displays operation instructions for its current context, followed by its position.

Fig. 4: Rendered shell and a sequence of selected frames with curvature plots.
Graphical manipulation of positions are performed through specially crafted draggers, Fig.3.

- Draggers are geometrical objects that can be moved in selectable main planes, or constrained in specific directions, independent from viewing direction. (No more translation parallel to the screen.)
- Draggers indicate available degrees of freedom.
- They have a protection against uncontrolled translation in or out of the screen, when one of its axes or planes gets close to parallel with the viewing direction.
- They are easy to grab with the mouse, as they are contained within something that resembles a soap bubble, which reacts to mouse clicks without obstructing the model.

When no action is active, the left mouse button is in selection mode. Points and vertices become available on selected curves, and a click on them restarts the tool that was last used for manipulation. (Previously one needed several functions to accomplish this.)

The middle button, combined with modifier keys, provides all rotation and panning functionality, which is available at any time. (Which was not the case previously.)

Curves light up when the mouse pointer is moved over them and show the following additional information instantly:

- The name of the part it belongs to in the status bar
- A curvature plot with colour-coded gradient
- The position of spline vertices
- Knuckles
- The position and weight of fairing points
- The direction of specified tangents

For all these things one previously had to use one or several functions.

- This pre-lighting indicates what will be selected upon a mouse click. Because the mouse pointer does not need to be exactly over the element, this visual feedback reduces the precision-requirement of hand movements.
- The active action can control which items are selectable, so that illegal selections cannot be made. A different pre-light colour provides early warning to the user and a status message may explain the situation. This saves many a pop-up box.

A tree view presents the structure of the model, showing solids, polycurves and curves, and some of their properties like visibility, lock status and master-slave connections. Selections may also be made here.

All open modeling windows are updated simultaneously, displaying selections, curvature plots and draggers. All windows are active at the same time, the user may start in one and continue in another. (This may seem obvious, but is a real improvement over the previous generation.)

We have support for SpaceNavigator devices from 3Dconnexion, Fig.5.

There is no command line. We believe that our action panels are so powerful and intuitive that you can do anything you would be able to do on a command line, only better.

5. Future

Even though there is no pressing need, once the new generation is feature-complete and stable, we can start tripping away the unused Pascal code of the previous generation. This will be much easier than when we started, because it is clear what code is still in use and what code has become redundant. The cause for the earlier entanglement with other code, like pop-up dialogues initiated by computational procedures, will have been removed by the new interface, and therefore the presence of remote code in the DLL, like Ywin, will be obsolete.

7 http://www.3dconnexion.com/
As both Qt and Coin3D support multiple computer platforms, it is only the Prospero-compiled DLL that keeps us tied to MS Windows. With a reduced Pascal core we can consider switching to another compiler that does support multiple platforms, like gpc, as the number of incompatibilities that require porting will be much easier to handle. Then the road will be open to both 64bit binaries and platforms like Mac and Linux, should the customer desire.

But our biggest wish is to extend the fresh Fairway with new exciting features, like spatial deformation mentioned earlier, and other ideas that we like to reveal at a later opportunity.

Fig. 5: Multi-monitor setup with detached windows for maximal work area. A 6-degrees-of-freedom navigation device in the foreground

6. Conclusion

One might suspect that by keeping the computational core and replacing “just” the user interface one would produce an application that resembles much its predecessor in how it functions and is operated, only with a shiny appearance and probably with some visual effects. Instead, the change has enabled us to redesign the way we work with the program, providing a completely new user experience. Much of the empowerment is due to the third-party frameworks that we are using. It has been, and still is, a fun project to work on, albeit not without some hard moments. These are the lessons that we have learned:

1. Be strict in separating algorithms and interfaces. This counts for interfaces to devices (printers, plotters, sensors), to program applications (dedicated: fairAPI; generic: agent-based), and to humans (GUI). No pop-ups, ideally not even strings. Just codes, if you have to.
2. Implementation takes twice as long as you planned; even though you knew that and planned for double the time you thought the job would take.
3. Favor third-party libraries that provide access to its source code above the ones that do not.
4. Anticipate changes in technology and design for flexibility. We are not good at predicting the
future and decisions are likely to be suboptimal. Doing the best you can is still better than programming for the present only.

5. Develop in iterations, covering a broad set of systems progressively implementing details, in order to recognize problem areas early.


7. Value the interaction between the user and the developer. You can't design a system like this the way they teach Programming Methodology in university: Analysis → Design → Implementation → Validation. It is more a process of investigating what is wanted and investigating what is possible and then have the two grow closer in rounds of iteration. What is wanted is affected by the possibilities and new possibilities are discovered in response to what is wanted. In the end you get more than you thought you wanted and thought was possible.

References


Automated Strength Analysis for Propeller Blades

Sebastian Daniel Bade, Gemanischer Lloyd, Hamburg/Germany, sebastian-daniel.bade@gl-group.com
Andreas Junglewitz, Gemanischer Lloyd, Hamburg/Germany, andreas.junglewitz@gl-group.com

Abstract

In this paper the modular tool chain is illustrated that enables the Germanischer Lloyd (GL) to cope with heterogeneous design representations handed in for approval by propeller manufacturers. It is applied to perform an automated calculation of the design as precondition for a quick plan appraisal. Various professional engineering tools of the presented approach are interlinked. The Friendship Framework is used to create, examine and mend a parametric geometry representation of the designs. A lifting surface code is applied to determine the hydrodynamic loads for the propeller and ANSYS is used for the calculation of blade stresses. The work-flow and the coupling of the individual software tools are controlled by a set of scripts while the interactive user interface for this controller is again supplied by the Friendship Framework. The pros and cons of this modular approach will be discussed.

1. Introduction

The ships screw propeller is one of the most essential component related to ships safety. Therefore the design of a propeller has to undergo a plan appraisal by a classification society. Within this plan appraisal a check of the propeller blade strength is performed. This check is based on simple formulas as well as on FEA calculations. The simple formula is derived from beam theory. It can only be applied to old fashioned blade designs with low skew back. The skew back characterizes the backward sweep of the blade centreline from the direction of rotation (figure 1). The contour of the blade is not radially symmetrical about blade center axis. This geometry causes a twisting of the loaded blade and the resulting stress is not covered by the simple formulas. Propeller designs with a large skew back require a stress calculation taking the real 3D blade geometry and the hydrodynamic loads into account.

The flow field behind the ship’s hull is inhomogeneous. The propeller operates in the wake behind the ship that varies in direction and speed. The propeller blades are rotating around the centre line of the propeller and experience a variation of the onset flow due to the wake distribution in the area of the propeller disk. The inhomogeneous inflow causes a variation of the load – in the magnitude as well as in radial and chord wise distribution. To evaluate the design of a propeller it is necessary to identify the circumferential positions of the blade in the wake field that cause extreme stresses. The wake field behind the ship is normally measured in a towing tank of a ship model basin and scaled to the real ship. Such a wake field has to be applied in the calculation of the 3D hydrodynamic loading of the propeller. Currently the Germanischer Lloyd (GL) applies a very reliable vortex lattice method to calculate the hydrodynamic blade loads in the various wake field positions. This method requires information about the blade outer contour, the distribution of thickness and other data derived from the geometry of the design. From the calculation results the most critical positions of the blade in the wake field as well as the associated load distributions have to be selected based on pre-defined criteria such as bending moment and spindle moment acting on the blade.

Additional criteria for the plan appraisal of the design are the loads created when the propeller is acting in ice. These loads are modelled by applying surface loads on certain regions of the propeller blade as specified in the rules.

While a very detailed geometric representation of the propeller is of minor importance when calculating the hydrodynamic loads more facts are needed for the calculation of the strength. Such details include the hub or the transition between hub and blade. The stresses are calculated using finite element analyses (FEA).
The FEA is used more frequently nowadays due to three reasons. At first the number of unconventional propellers, e.g. with a skew angle in excess of 25°, has grown considerably. Secondly, newly introduced ice rules require a screw propeller blade FEA and FEA has become simple to be applied on a conventional Windows or Linux desktop computer. Nevertheless, the geometry treatment is a critical point in the preparation of a FEA calculation. The entire ships screw propeller is a solid object represented by a very complex 3D geometry. Starting from the set of global and local parameters used for the hydrodynamic calculation a 3D surface representation of the propeller has to be created including all the details that are required for the detailed stress analysis. The essential task is the creation of a closed surface representation that can be seamlessly processed in the typical FEA software tools. The components of this integration include the generation of a mesh for the FE-model and the mapping of hydrodynamic and ice loads as well as boundary conditions into the mesh.

The challenges in this process are the variety of geometry definitions used by the different designers, propeller makers and model basins. The community lacks a standard for the propeller geometry.

In this paper an automated work-flow to evaluate the propeller design at a classification society is introduced. For this approach a lean set of essential parameters characterizing the design from hydrodynamic as well as from strength point of view is used to feed the process.

2. The parametric definition of the propeller design

In the past the data defining a propeller design and the geometry details were created and communicated on paper in the form of drawings, offset tables and calculation sheets. Nowadays, nearly the complete propeller design process is software based and generally the digital information produced in the process is suitable to support the appraisal of the design by a classification society. The software systems used by the designer support the evaluation of the hydrodynamic performance of the design. It is also possible to examine the strength characteristics or the manufacturing process of the new design. One would expect that such information can be directly handed into the process of design appraisal at a classification society. Unfortunately there are several potential conflicts hindering a seamless integration of the data. A detailed CAD representation of the design may expose much of the manufacturer's know how and understandably the suppliers tend to protect their intellectual property. Another typical problem with CAD data is the transfer of data into other systems which yields corrupt or not entirely complete data models. A classification society or other service providers may not have the same CAD system available. Many times the imported CAD representation contains defects due to differences in the modelling approach.
From the standpoint of a classification society using a detailed CAD file of the ship propeller does not render a comprehensive representation of the design in the appraisal process. The classic rule based approval approach is based on some characteristic parameters that are difficult to collect form a 3D CAD model. The parameter values defining the propeller design together with the design drawings are considered to be the mandatory set of information used by the classification society. It can be checked and stamped. Such values can easily be cross checked with the real propeller. Additionally the scope of the design appraisal is entirely documented by the parameter representation and the drawings of the design.

In naval architecture it is common practice to eliminate the need to deal with the detailed definition of the 3D blade surface by using a collection of propeller parameters that ultimately represent the surface. These parameters include both global and detailed parameters. These parameters define the overall size and shape of the propeller. Other parameters render 2D blade section and how the “wing” shape actually moves in the fluid. Further values typically identify where the 2D blade sections are located in space, as well as considerations for clearance, strength and manufacture.

In the process for the automated strength analyses of propeller designs described here the geometry representing parameters are stored in a file-format called “PFF” (Propeller File Format), Schulze (1998). This format is currently used by makers to transfer their geometries to Germanischer Lloyd in order to speed up the approval process Brunk (1998). A characteristic section of this format is presented in Fig. 2.

The PFF format contains the information necessary to properly reproduce the hydrodynamic characteristics of the propeller. The elementary strength characteristics of the propeller design are rendered fairly well by a propeller geometry created using these parameters. Details of design elements such as the leading edge of the propeller tip or the anti singing edge are not subject of the appraisal as they usually don’t contribute significantly to the overall strength. These details are not part of the PFF file. The type of fillet used to smoothly connect the propeller blade to the hub can be extracted from the design drawings and common parametric definitions. These assumptions are sufficiently precise to allow an overall judgement of the global stresses experienced. The hub design is also characterized in the drawings handed in.

```
| * PropDiameter / HubDiameter / Scale / ExpArRatio / BladeMass / |
| 3600.00        980.00        1.00000   0.460000      0. |
| * MomentInert / ShaftPower / |
| 3410.00       1800.00       |
| 4            3            12           13             1 |
| * r/R / r / CordLength / |
| 0.272      490.000    584.000 |
| * Pitch / DistLeaEdge / MaxCamber / MaxThick / Rake |
| 3546.000   292.000         19.300       126.900    0.000 |
| * Station / Dist.SucS / Dist.PressS / |
| 0.02500000 22.60000000  -16.50000000 |
| 0.05000000 32.20000000  -21.70000000 |
| 0.10000000 45.50000000  -28.20000000 |
| 0.20000000 63.00000000  -36.00000000 |
| 0.30000000 74.00000000  -40.80000000 |
| 0.40000000 80.50000000  -43.40000000 |
| 0.50000000 82.70000000  -44.20000000 |
| 0.60000000 80.70000000  -43.00000000 |
| 0.70000000 73.20000000  -39.00000000 |
| 0.80000000 58.80000000  -31.70000000 |
| 0.90000000 35.00000000  -21.20000000 |
| 0.95000000 20.10000000  -13.50000000 |
| 1.00000000  5.00000000   -5.00000000 |
```

Fig.2: Sample section from PFF propeller parameters
In summary the parameters defined in the PFF render an adequate model representation of the propeller and contain the details necessary for a well-founded global strength analysis. Consequently this kind of geometry definition is used as solution for the problem of the large variety of different user defined formats and CAD models handed in for the appraisal process. Also the ASCII data can easily be checked against the officially stamped documents such as drawings and offset tables.

3. Modular automation work-flow

In this section the general work-flow for the automated strength analyses of ship propellers is presented. The focus is on the basic procedures involved as well as on the elements the process is compiled of. The central idea is the application of a modular tool chain that can be executed without further manual user interaction. This way it is possible to enhance the whole work-flow by replacing single modules. These modules can be developed and tested individually. The Friendship Framework is used interactively as tool to control the complete work flow.

The automation of the strength analysis for propeller blades is composed of several specialized programs and scripts. Two different systems are in charge of the general flow control. The Friendship Framework provides the user interface for the creation and manipulation of the propeller geometry and the PFF parameter file. It also acts as the user interface for the calculations. The execution of the individual calculation steps and the transformation of data are controlled by Python scripts. This set up has the advantage that the whole calculation work-flow is separated from user interaction and can be developed independently. Given a correct PFF file, a wake-field file and the parameters for the calculation processes the whole strength analysis of the propeller blade can be run without user interaction. On the other hand the user interaction can be developed without running the calculations.

![Diagram of modular automation work-flow](image-url)

**Fig. 3:** The modular system of the automated propeller appraisal work-flow

The overall activity consists of the preparation of the PFF file containing the parameters defining the propeller. A next step is the generation of a proper 3D surface geometry of the entire propeller for the
preparation of the tetrahedral finite elements mesh. In parallel the PFF file and a given wake field is used for the calculation of the hydrodynamic loads. From all resulting loads relevant cases are selected for further processing in the FEA. The tetrahedral mesh, the load cases and the boundary conditions are used to execute a FEA calculation. The calculation results are transformed into figures displaying the stresses in the propeller blade. The final step in the appraisal process of the propeller is the review of result by the classification engineer. The entire process is illustrated in Fig. 3.

3.1 Creation, checking and editing of PFF files

Starting point for the automated strength analysis of ship propellers is the PFF file. There are basically three use-cases to be considered: checking and editing of existing PFF files and creating new PFF files.

In many cases a proper PFF file representing the propeller can be provided by the propeller manufacturer submitting the design. Even though the PFF file is a human readable format the engineer at the classification society needs a tool to visually check and interactively manipulate the parameters. Another regular situation is that there is only a design drawing and parameter sheets handed in for appraisal. In such a case the starting point of the process is to load a general PFF-template and to interactively adapt the PFF parameters to the information available.

There are several ways to present the parameters of the design. A general view is a diagram of the 2D section used for the design, Fig. 4 (bottom). Such a diagram can be very useful to check the sectional data. Typing errors are easily detected on the figure as the curve representing the section has discontinuities.

Another useful illustration shows the curves of the radial distribution of pitch, chord length, rake, skew, maximum thickness and maximum camber, Fig. 4 (bottom left). These curves characterize the propeller blade. They are typically used by the designer and for the experienced engineer to create an idea of the hydrodynamic properties of the propeller, Harries and Käther (1997).
The PFF file is imported to the Friendship Framework and transferred into a parameter based representation of curves and surfaces. Those can be presented in different views. Several diagrams can be arranged in a user defined set up storable in the system configuration for later reuse. Generally it is possible to use the Friendship Framework to interactively manipulate the parameters of the propeller design by picking a point on one of the curves and change its value by pushing it to a new location. The changes to the parameters can be stored in the PFF format for further processing.

3.2. Creation of the propeller surface geometry

In many cases of the failure of an automated CAE process the problem is caused by a faulty geometry representation. From a general point of view the reason is the misinterpretation of the different aspects of 3D geometry. There is the computer generated image representing the illusion of a real worlds object to the viewer. On the other side there is the 3D object modelled in such a way that it reflects the object’s physical properties in a calculation process such as CFD or FEA. For each calculation process a different model of reality has to be created. Usually these models are incompatible to each other. For this reason each CAE calculation typically requires an individual model representation.

The task here is to create the complex 3D geometry of the propeller in such a way that it can be reliably transformed into a mesh representation applicable in the FEA. The central challenge is that this 3D geometry is composed of boundary surfaces without any gaps.

In the automation process described here the surface model of the propeller is constructed using Friendship Framework. Its capability to handle parametric geometry entities and its feature definition language facilitate the tools necessary to create a closed 3D surface boundary representation of the propeller. The creation of the suction side and the pressure side surfaces of the blade is a straightforward skinning of the sectional data from the PFF file, Fig. 5. These two surfaces depict the topological anchor objects for the entire propeller. In order to create a tetrahedral mesh for the FEA calculation, a closed 3D shape is necessary. In addition to the blade surfaces, a hub has to be modelled. This hub has to be smoothly connected to the blade using fillets so that flow and strength criteria are fulfilled. Additional fillets are needed at nose, tail and tip of the blade to close the shape.
The main challenge is to identify a topological set up for the network of surfaces that form the entire propeller including the blade, the hub and the fillets in such a way that the manifold designs coming in for appraisal can be handled similarly. The goal is to reuse this topological set up. In the initial step presented here this topological set up is created for the traditional fixed pitch propeller. The hub is represented by a twisted slice of a cone. The transition between blade and hub is the root fillet and in the future hub designs are developed to better synthesize other propeller design variants. For the current state of development the resulting topological closed shape is shown in Fig. 6.

3.3 Calculation of hydrodynamic loads

The hydrodynamic loads are calculated using the vortex lattice flow code VORTEX provided by the SVA, Potsdam. This code takes the PFF file to generate the geometric model of the propeller blade for the calculation. Additionally a wake field and the operational parameters such as propeller rotational speed, ship speed have to be appended. The VORTEX code calculates the loading of the propeller blade in a series of incremental angular positions. For each angular position of the blade the local wake field is applied. Each individual result is created containing the hydrodynamic force vectors at the positions of the calculation grid representing the propeller blade. In order to determine the critical load cases and their angular position the extreme values of bending moment at blade root positions and extreme values of the spindle moment around the blade axis are calculated. The identification of the relevant load cases is performed by a set of Python scripts integrated into the automated process. The result of this procedure is a set of load cases prepared for the FEA calculations.

Some propellers are intended for the operation in ice covered waters. They have to undergo additional strength analysis in the appraisal process. The load cases to be applied depend on the type of propeller and are defined in Rules for Construction of Sea going Vessels, GL (2011). Certain areas of the blade have to be loaded with a static surface pressure, which is determined taking the ice class and individual geometrical and performance data into account. Taking the geometric representation of the blade modelled using the Friendship Framework the ice load cases can easily be generated by a feature. Samples of these areas are shown in Fig. 7.

3.4. Creation of a mesh for FEA

Typical screw propellers are represented by a fairly complex 3D surface geometry. This is also true for the geometric representation discussed here for the automated strength analysis. To simplify the mesh generations the tetrahedral elements are chosen for the FEA calculations. The tetrahedral elements mesh can automatically be generated using the tools provided together with the FEA software system. Precondition for the automatic mesh generation is usually a closed boundary representation of the object.

The Friendship Framework is used to generate the 3D propeller geometry from its parameter based representation. By applying the topological set up sketched above, the set of 3D surfaces representing
the propeller is closed by default and forms a closed shape. The trouble here is that the Friendship Framework currently provides no interface for the export of closed shell boundary representation. To export the geometry to a FEA system, the IGES surface representation could be utilized but this transfer-format does not carry the topological net of surfaces properly. Usually receiving systems are not capable to reconstruct the closed shape of the propeller when transferred in the IGES format. Another option is to transfer the propeller as tessellated geometry. This technique is commonly used for complex geometries. The transferred data consists of a large number of triangles that represent the original shape. The typical format is called STL. This procedure is less error-prone at the price of a lower geometric accuracy. At the current stage of development the STL transfer is the method used in the automation procedure discussed in this paper. With the availability of a better method a new transfer module will be plugged into the modular process.

A public-domain tetrahedral mesh generation tool is used that reliably creates the mesh from a tessellated geometry representation. This tool is netgen, \textit{Schröderl (2003)}. This way the tetrahedral mesh is created independently from the target FEA system. An extensive control for the density distribution of the net and the size of cells is possible. Using the tessellated propeller geometry representation, a tetrahedral mesh is generated, Fig. 8, in netgen.

![Fig. 8: A propeller blade with tetrahedral mesh](image)

### 3.5 FEA calculation of blade root stresses

Prerequisites for the FEA calculation of the blade stresses is a tetrahedral finite element mesh, a file defining the loads and the boundary conditions. In the procedure presented here, the commercial FEA-software ANSYS is used for the calculation of the nodal displacements and the resulting stresses. The automation of the FEA workflow is done using APDL macros. APDL stands for ANSYS Parametric Design Language and is a commonly used scripting language for the automation of common and reoccurring tasks in the FEA calculation process.

As a first step the hydrodynamic loads have to be mapped onto the nodes of the tetrahedral FE mesh. This is performed by an algorithm that transforms the hydrodynamic loads acting at specific positions of the hydro-grid representing the blade into forces acting at the nodes representing a similar position in the tetrahedral mesh. This procedure can be applied for a global strength analysis. The same mapping procedure is used to apply the fixed boundary condition to the surface nodes of the hub segment. This procedure is reused to apply the forces resulting from ice loads to the blade nodes.
With the availability of a mesh, the loads and the boundary conditions a FEA can be started. It is controlled by an APDL script that sets all necessary parameters and triggers the calculation. All load cases are processed in batch mode. The resulting nodal displacements have to be transformed into a material stresses. The post processing of the stress plots is performed using the functionality provided by ANSYS. A sample result is depicted in Fig. 9.

Fig.9: Plot of calculated stresses

4. Conclusions

The problem of handling a wide variety of different screw propeller geometries, user defined file formats and different CAD output files is overcome by using a simple ASCII file format for the propeller blade geometry. The Friendship Framework as back bone for the batch process and a series of different exchangeable calculation modules realize the steps from geometry to stress distribution, which can be evaluated by the approval engineer.

The automated and faultless transformation of 3D boundary representation models could not reliably realised at this stage. Alternatively building up a 3D body from a number of unconnected surfaces does not solve the problem either. The created body is not a closed shape and can not be handed over to other tools without errors. However, it is of crucial interest to avoid manual geometry manipulation and transformation. For this reason a special solution has been developed and described.

The concept of loosely coupling a modular chain of specialized tool proved to be very efficient. It is easily possible to develop or improve an individual component and integrate it to the tool by adding a simple code line to the controlling Python script.

The future development of the work-flow will take advantage of the modularity. One of the first procedures to enhance will be the mesh generation. Additional modules will be elaborated in order to...
deal with design variants such as the type of blade foots used in controllable pitch propellers. Here the blades are bolted to the hub and the critical stress areas may occur at the bolt holes due to notch effects.

Currently a further development of the parameter definitions used for the propeller geometry definition is under discussion. The aim will be to satisfy more users and to enlarge the community of makers, institutions etc. being able to operate their systems with this kind of propeller definition.

References

GL (2011), Rules for Classification and Construction, I Ship Technology, 1 Seagoing Ships, Chapter 2 Machinery, Section 13, Germanischer Lloyd, Hamburg


HARRIES, S.; KÄTHER, B.L. (1997), Rechnererzeugte Propellergeometrien, Jahrbuch der Schiffbautechnischen Gesellschaft 91, Springer

Underwater Archaeology Surveys with Autonomous Robots

Andrea Caiti, ISME & DESE, Univ. Pisa, Pisa/Italy, caiti@dsea.unipi.it
Vincenzo Calabrò, ISME & DESE, Univ. Pisa, Pisa/Italy, v.calabro@dsea.unipi.it
Giuseppe Conte, ISME & DIIGA, Polytech. Univ. Marche, Ancona/Italy, gconte@univpm.it
Pierre Drap, ESIL/LSIS, Univ. Mediterranee, Marseille/France, pierre.drap@univmed.fr
David Scaradozzi, ISME & DIIGA, Polytech. Univ. Marche, Ancona/Italy, d.scaradozzi@univpm.it
Silvia M. Zanoli, ISME & DIIGA, Polytech. Univ. Marche, Ancona/Italy, s.zanoli@univpm.it

Abstract

Underwater robots, tethered (ROV – Remotely Operated Vehicles) or unsupervised (AUV – Autonomous Underwater Vehicles), have a great potential in underwater archaeology, in particular to survey and map sites beyond the diving depth. To be an effective tool for archaeological exploitation, however, underwater robots must perform their operation in effortless, plug & play, manner. Our research group has conducted several projects to lead to a complete automation of the mapping process, including production on-site of photo-mosaicked maps and virtual reality rendering in post-processing phase. An overall description of the techniques developed is given, and results from field experiments are shown.

1. Introduction

The increasing diffusion of unmanned underwater vehicles is leading to novel procedures for performing tasks previously carried out from surface platforms or with the aid of divers, as well as to the definition of new tasks that are just made possible because of the availability of the vehicles. It is interesting to note that, while tethered unmanned vehicles (ROV – Remotely Operated Vehicles) are part of the routine off-shore work since at least thirty years, and are commercially available in a great variety of sizes and dimensions, the more innovative Autonomous Underwater Vehicles (AUVs) have only recently reached commercial maturity, with the presence of diversified models, tailored for different applications and addressing specific problems. Informative scientific and technological surveys on the state-of-the-art of ROVs and AUVs are given by Kohnen (2008), Nicholson and Haley (2008). Focusing on more research-oriented applications and developments, the reader is referred to the two special issues of the Journal of Field Robotics appeared in November 2010 and January 2011 and to Marani et al. (2010) and references therein.

The main applicative fields of underwater robotics are the off-shore industry and military missions. Several applications lay at the border between civilian, scientific and military scope (“dual use” technology), as it may be the case for environmental monitoring, search and rescue missions, harbor surveillance and protection. Since these are the fields from which market reward is to be expected, design and production of ROVs/AUVs and ancillary systems has been mainly oriented toward the satisfaction of their applicative needs.

Marine archaeology represents a very peculiar application niche for underwater technology. Underwater sites of archaeological interests (referred as “relicts” in the following) need to be found, mapped and periodically inspected, to monitor their preservation state. Not always there is scientific interest to excavate a relict site: usually, it is important to preserve the site and to avoid destroying specific information carried out by the relict layout. When there is a superior interest to excavate the site, for instance to recover remnants buried within the seafloor, a detailed map of the site has to be collected and stored. Indeed, at any site, a surface map with precise geometrical relationship among all the artifacts present on the site needs to be prepared and used as a reference background at each new inspection of the site. Such procedure is required also in operations of so-called “prevention archaeology”: in several countries (Italy amongst these) it is required by law to carry out a seafloor surface archaeology investigation on any submerged area where any work requiring seafloor morphology changes is planned (such as harbour dredging, foundations lay-out, pier extensions, etc.).
Considering the efforts in manpower and equipment required by any such operation, the abundance of relict sites and the by now chronic lack of resources affecting the government agencies responsible for mapping, inspection and preservation, it is clear that low-cost automation of such operations with the aid of autonomous or semi-autonomous underwater systems can be of great benefit to the marine archaeologist. This has been the theme of focused workshops, as for instance Akal et al. (2004) and papers therein. However, from the published literature, most of the technological case-studies belong to the following cases: applications of technology specifically developed in other fields (as for instance military or off-shore industry tools employed in archaeological surveys); development of special purpose, very specific, one-piece only archaeologically-oriented tools (Ballard et al., 2000)).

It is our opinion that neither case is a viable way for a sustainable technological growth of marine archaeology activities; this is because in both cases the costs associated to the instrumentation, and/or the complexity of the instrument operation are far above the reachable target of responsible archaeology agencies. For instance, the work described by Ballard et al. (2000) requires a dedicated ship with trained and skilled engineering personnel to operate the equipment. While such a work is certainly relevant from the technological research view point, it is hardly the case that such modus operandi can be standardized and routinely used by archaeological groups with low budgets. What is needed by the archaeologist professional is a set of low-cost, easy-to-use, plug & play technological tools and procedures that can help in reducing the overall cost of site inspection and documentation. To be effective, such tools and procedures have to be designed from the start with the archaeological application in mind.

Our research group has identified the above problems since several years, Caiti et al. (2006), Drap et al. (2008), focusing on the specific aspect of semi-autonomous geo-referenced mapping of a site and production of 2-D and 3-D maps of the relict. In particular, an ROV integrated with a USBL acoustic positioning system has been developed, in order to gather geo-referenced optical and acoustic with autonomous navigation. Specific image processing software has been built to produce bi and tri-dimensional map from the still images acquired by the camera installed on board the vehicle. In this paper we give a general description of the developed system and present some previously unpublished data from a relict site in the Atlantic Ocean off the bay of Cadiz, Spain. Finally, future plans are described, in particular with respect toward migration of the system on AUVs cooperating in team.

2. Semi-autonomous survey and mapping of archaeological sites with a ROV-based system

In this section the basic components and the integration algorithms of ROV-based systems developed by the Authors for archaeological survey and mapping are illustrated. The system has been designed starting from COTS components with in-house development of all the electronics and of the system software. The fundamental guideline of the development has been that of prototyping a light, easy-to-use, system, responding to the archaeologists requirements.

The research has been carried out in the framework of several EU-funded projects (see the acknowledgment section for a reference to the project web-sites). Marine Archaeologists were actively participating in these projects, and the opportunity of having their presence in the design team has been of fundamental help throughout the process of development and test. The first step in the design of the system has hence consisted in the proper understanding of the archaeologists work and of their functional requirements. Roughly, the major activities that define the work in marine archaeological surveys include:

- pre-analysis of the area
- definition of a restricted search area
- mission design
- local survey
- documentation,
- excavation
- post-analysis and final documentation.
Our interest in the design of an automated procedure is concentrated on the phases which precede the excavation. The pre-analysis is meant for the specification of the search area. Generally, information about archaeological remnants is as vague about the objects to be identified as about their localization, so that a broad area must be covered in this initial phase. In many situations, this is done employing acoustic devices of various kinds (multi-beam and side-scan sonar) for scanning the sea bottom. The output of this activity is a preliminary map of the area, whose scale is large or very large in comparison to the objects of (possible) interest, which appear as acoustic anomalies. From the analysis of the preliminary map, archaeologists define a more restricted search area and possibly identify waypoints and targets that deserve special attention. The following phase is devoted to design the survey mission for optical data gathering, keeping into account the above information and the equipment’s features (ROV manueuvrability, payload capability, support vessel/boat characteristics and so on) as well as possible environmental constraints (depth, visibility, presence of current, tides and so on). In this phase, in particular, different solutions for structuring the area of interest are considered. Then, local survey takes place, according to the data gathering modalities and to the navigation specifications defined in the previous phase. It is at this level that, by providing on-line a sufficiently detailed documentation of the site, the archaeologists can modify the parameters of the exploratory mission and repeat the data acquisition accordingly to emerging mapping needs. Finally, a documentation of the marine site is produced from the post analysis of the data gathered during the (various) surveys, acoustical and optical, and a 3-D model is generated.

In order to comply with the functional requirements above described, the light-weight inspection ROV (Phantom S2) of the DIIGA – Polytechnic University of Marche - ISME Research Unit has been equipped with:

- high definition underwater photo camera with companion flash lights for optical data acquisition
- on-board Inertial Motion Unit (IMU)
- external Ultra Short Base Line (USBL) acoustic positioning system
- high frequency (675 kHz) scanning sonar for high resolution acoustic data acquired from a position co-located with the optical photo camera.

The ROV maintained also the video camera and lights which are part of its default equipment. The system and part of its component are depicted in Fig. 1. The use of an acoustic positioning system stems from the need of an external, vehicle independent, geo-referenced positioning system. Among the possible systems for acoustic positioning, USBL systems may not be among the most accurate (usually Long Base Line – LBL – systems at short range have better performance), but USBL systems are also much easier to transport and deploy, and cheaper to rent or buy. From the point of view of on-board sensors, there is also a wide range of possible choice for the IMU; also in this case, we selected a relatively cheap and low-performance unit.

The technical challenges posed by the integration of the acoustic positioning system with the rest of the equipment are twofold: one is the integration with the IMU data, in order to obtain consistent geo-referenced data for on-line autonomous guidance, navigation and control; the second is the storing of all the navigation data together with the payload data (photographs for photogrammetric reconstruction) with consistent time stamp at the correct rate. The main technical features of the used hardware are now reported. The instrumentation has not been selected based on the performance, but on the cost, giving privilege to relatively low-cost systems that could still guarantee, when appropriately integrated, an acceptable performance. In the case of archaeological work, such acceptable performance is order of the meter for absolute positioning error, and of the centimeter for relative positioning error.
Fig. 1: The original ROV system (top) and the modified system with the installed payloads and navigation systems in action over a relict site (middle) and seen from the seafloor (bottom).
The USBL system we have integrated has the following main nominal specifications:

- Operating frequency: 35 – 55KHz
- Telemetry: RS232
- Operating depth & range (max): 500 m
- Accuracy: 2.7 % of slant range

It has to be mentioned that, being our ROV umbilical cable of 120 m, the USBL has always been employed at a slant range for which the nominal instrument error is of the order of 1 m.

The IMU is equipped with three gyros and three accelerometers. From these data, the vehicle position and attitude (defined with the traditional SNAME convention) can be estimated by implementing an Inertial Navigation System (INS). The INS data are merged with the USBL data according to a multi-rate discrete-time Kalman Filter (KF) scheme, to account for the different sampling rate of the IMU (30 Hz) and of the USBL system (1 Hz). Note that at the surface the range and bearing measurements of the USBL system are converted in absolute positions with DGPS data, taking into account the displacement between the DGPS receiving antenna and the transducer position. An additional set of sensors is available on the ROV and it is recorded for post-processing purposes and sanity checks; this set includes depth meter (pressure gauge), altimeter (echo sounder), encoders on vehicle shafts. Through these combined system, each photograph is associated to the navigation and system data, in particular with the geo-referenced position of the camera and with its orientation with respect to the seabed, assumed to be flat. Integrated optical/navigation data are directly stored in JPEG/EXIF format.

A set of automated behaviors has been developed, including auto-depth, auto-heading and way-point navigation. These automated facilities can be combined together through a mission planner, in order to program in advance an autonomous mission for the vehicle. Note that, while the vehicle remains tethered to a surface platform, where its navigation and payload data may be monitored by human operators, the mission itself is conducted by the ROV without the pilot intervention in a completely autonomous fashion. Of course, the supervisor can preempt at any time the ROV autonomous command and control system, and force it back to the remote control mode. As usual in most ROV systems, commercial or not, there is the possibility of co-presence of autonomously controlled modes (e.g., auto-depth), and pilot steering for the remaining vehicle degrees of freedom.

One critical aspect in the association of data from different sensors is the synchronization of the system. All the systems on board the vehicles, including the camera, and the USBL system are synchronized with GPS time at mission start. GPS clock is always available to all the equipment, through the umbilical cable. However, the instruments are polled serially before data is written on file, causing a non-constant unknown time jitter. ROV navigation sensor data are sampled at 30 Hz. Acoustic position is sampled at 1 Hz. Photographs are taken at 0.3 Hz. ROV speed is kept controlled to allow for the necessary overlap between two consecutive photographs (60% forward overlap between two subsequent photographs over a linear strip – the ROV speed selected depends on the setting of the altitude above the relict and of the camera setting; usually, in most of our experiments, the ROV speed has been set to 0.3 m/s, with an altitude above the relict of 3m). With the above parameters, and considering the slow dynamic of the ROV vehicle with respect to the sampling rate, the error due to the time jitter can be considered negligible with respect to the other source of errors in the system.

3. Data processing

The set of data acquired by the vehicle consists in the sequence of still images acquired from above the relict, with the camera view oriented vertically from top to bottom, and the acoustic bathymetric information. These data serve several different purposes. After the first survey over the relict, along pre-programmed paths (“transects”) designed on the basis of the pre-area survey, the archaeologist in
the field needs a rapid assessment of the site in order to evaluate the quality of the data and further actions (redefinition/enlargement of the transects, change of the acquisition system parameters). In our systems, the rapid assessment is obtained through a 2-D map of the site obtained by quick mosaicking of the camera images. At the end of the survey, the optical data will be used to obtain a 3-D map of the site through photogrammetric processing and will be integrated with the bathymetric data, in order to produce a complete documentation of the site. Further off-line processing by the expert archaeologist may consist in the identification of specific artefacts (e.g., amphorae) within the relict, geometric modelling of the artefacts, model-matching, annotation, augmented/virtual reality rendering, display to the scientific community and/or to the public at large (see Fig. 2 for a block diagram of the complete data flow). In the following the basic information regarding the processing stage that has to be performed in the field, i.e., the 2-D quick mosaicking technique, is given.

Fig. 2: Data flow and processing stages: 2-D photomosaic map needed on-line in the field; 3-D photomosaic, merging with acoustic bathymetric data, geometric modeling, virtual/augmented reality, all performed off-line.

In the 2-D map generation (on-line processing), camera images are processed sequentially, in the same order they are acquired. In particular, a Scale Invariant Feature Transform (SIFT) procedure, Lowe (1999), is first applied to the temporal-image sequence and used to perform pairwise key point detection on subsequent images. In order to speed up the processing, position and attitude data from the ROV navigation system, together with the altitude from the sea bottom elaborated from the bathymetric sonar returns, are used to orient and scale the associated images. Furthermore, images are converted to grey scale images not taking into account saturation and hue information. The choice to apply a SIFT operator is motivated by the fact that, been the SIFT detector invariant to image translation, scaling, and rotation, and partially invariant to illumination changes and affine or 3D projections, is well suited to underwater images feature extraction. By comparing SIFT output key points of two subsequent images \( I_k \) and \( I_{k-1} \) it is then possible to create real-time mosaics based on SIFT: common key points are found, as shown in Fig. 6, and on each couple of matched keys and relative image points, \( p_{i_k} \) and \( p_{i_{k-1}} \), an optimum problem can be stated as:

\[
\begin{align*}
& \min_{\theta} \sum_{i} || f_{i} \|_{\theta} - f_{i_{k-1}} \|_{\theta} \\
& p_i = [\hat{X} \ Y \ 1] \in f_i
\end{align*}
\]
T is an image space-transformation matrix. Thus, a growing “snake” mosaic is solved after each new image is added, Fig. 3. The map is presented in real-time to the supervisor on the surface platform.

Fig. 3: 2-D real-time photomosaic map construction based on SIFT processing. Data from the Pianosa 2006 field trial, VENUS project

At a later stage (offline- first stage processing), all the data previously acquired are processed again, this time by considering their totality, in order to contrast error growth due to sequential processing. Low quality images grabbed from the video stream are replaced by high quality photos downloaded from the camera internal compact flash memory. In this off-line phase of processing, more accurate local maps can be generated by the application of mosaicking techniques which account for all overlap information, including overlap from images that are not consecutive in time. The position and attitude information extracted from the integrated USBL/IMU navigation data is used for establishing correlation between images by grouping those taken (not necessarily sequentially) from close positions and, presumably, depicting overlapping areas of the sea bottom. The result of this processing is shown in Fig. 4. Although not performed on-line, this first-stage processing takes a relatively short computational time and its product can be made available shortly after the whole set of data has been gathered.
4. An example from the field: the Cadiz 2008 experiment

The implemented techniques have been applied in a set of experimental activities since 2006. In the following, some data gathered during the Cadiz 2008 field trial are reported. The Cadiz 2008 activity took place within the framework of the project ARCHEOMED, an initiative within the Inter-Regional programme of the EU, led by the Tuscany Region. The field test was part of a set of demonstration and instructional activities addressed to young archaeologists. The whole set of activities was organised by the Superintendence of the Andalucia Region, and saw the participation of the Centre for
Underwater Archaeology of Catalunya (CASC), CNRS Marseille, and the Interuniv. Research Ctr. on Integrated Systems for the Marine Environment (ISME), with its research units of the Universities of Pisa and Ancona.

The ROV operations in the Cadiz 2008 trial have been all but easy, the major problem being the late arrival of the equipment in Cadiz: the ROV shipment was caught by a truck drivers strike that lasted for almost a week. The late arrival greatly reduced the available operation time, and consequently the gathered data set was far less than expected. In addition, within the remaining available days bad weather occurred, causing also an increase in turbidity at the bottom in the area of operations. This in turn forced the system to navigate closer to the bottom than usual (2 m altitude), in an area of complex bathymetric features. Nevertheless, some interesting optical sequences were acquired and processed.

The data were gathered in the approaches to the bay of Cadiz, roughly three miles East of the harbour, in the Atlantic Ocean. In this area, remnants of guns from the Napoleonic wars are present, and the objective of the field trial was to systematically map their presence. While the objective could not be fully accomplished, for the reasons mentioned above, some of the guns could be mapped. The ROV was operated from the Research Ship Thetis, managed by CASC. Fig. 5. Figs. 6 and 7 report the image sequences and the on-line reconstructed 2-D map related to two different guns in the area. The value added of the mapping resides also in the precise geo-referencing of the relict that has been made possible by the automated mapping system. Fig. 8 shows a gun recovered from the same area and currently restored by the archaeologist of the Superintendence of Andalucia.

5. Conclusions and future work

The paper has described the main components of an ROV-based system for mapping of underwater sites of archaeological interest. The components and performance of the system has been specifically tailored to the needs of marine archaeologists, resulting in a relatively low cost system, easily operable even by a small team, and without the need of expert engineers to manage the operations. The system has been tested in a series of different sea trials, at water depths ranging from 15m to 65m. An adapted version of the integrated navigation/data acquisition system has also been installed on a different vehicle and operated at 130m in the Marseille 2008 field test, in conclusion of the VENUS project.

We are currently involved in the project “Thesaurus”, in which the goal is to operate a team of AUVs in a completely autonomous way, for wide area exploration, site identification and subsequent fine scale mapping. The goal is ambitious, but, in our opinion, reachable with the technology progress in underwater robotics. The project started on March 1\textsuperscript{st} 2011, and it is expected to come to a conclusion in 30 months.
Fig. 6: Original images sequence and 2-D photo-mosaic reconstruction.
Data from Cadiz 2009 field trial.
Fig. 7: Original images sequence and 2-D photo-mosaic reconstruction. The mapped gun has a rope and buoy marker previously deployed by archaeologist divers. Data from Cadiz 2009 field trial.

Fig. 8: Recovered gun from the Cadiz 2009 field trial
Acknowledgment

The work described in this paper has been supported by the EU projects EPOCH – Excellence in Processing Open Cultural Heritage (2004-08), http://www.epoch-net.org; VENUS – Virtual Exploration of Underwater Sites (2006-09), http://www.venus-project.eu; ARCHEOMED – Patrimoine Culturel Maritime de la Méditerranée (2007-09).

References


Fleet Wide Operational Reporting - Performance and Environment

Jukka Ignatius, Onboard-Napa Ltd, Helsinki, Finland, jukka.ignatius@napa.fi

Abstract

Modern ships generate a multitude of reports for different purposes. This operational data is of crucial importance for the owner, the management company or the charterer. The sheer amount of data requires a data management strategy to store, filter and analyze the information. Economical transportation means environmentally friendly operations and vice versa. Less consumed fuel simply means lower costs but also lower emissions. This paper looks into the different ways of data collection onboard as well as on how such data can be used in optimization, monitoring and analyzed for other purposes.

1. Introduction

The managers in the fleet offices worldwide are faced with similar problems of trying to improve the operations and understand what is going on onboard the ships in the fleet. As a typical measure, more reporting is required from the ships. This is all well, but at some point the reporting work gets quite tedious onboard. And when reporting procedures have been set up, it is not too rare that the shipping company’s offices receive quite a lot of data in the form of logbooks, logbook abstracts, miscellaneous spreadsheets, etc. This data, however, is almost never analyzed as a whole, nor is feedback delivered to the crew operating the ship. Analyzing this data is difficult or even impossible due to the extensive amount and distributed nature of the data sources.

In this paper, we will look into a practical approach to overcome the above-mentioned problems. Most of these difficulties could be surpassed by using the correct tools for planning, monitoring and follow-up. Please see the following figure for a simplified process model of a single passage. The process is iterative, Follow-up will, or at least should, feed the future planning tasks for continuous improvement. Each step is discussed in the next chapters in detail.

![Process Model](image)

Fig. 1: Process described in this paper

2. Planning

A ship, the largest human build construction for transportation, offers a magnitude of possibilities to plan and optimize the operations without any physical modifications to the vessel itself. It would be unwise to leave these possibilities unexploited.

The growing attention to emissions has given another positive outcome for any optimization: Less
consumed fuel naturally means less exhaust and emissions. Providing adequate decision support tools for the operating crew helps them to constantly improve their work even further.

2.1. Route, Speed and Engine Mode

Nowadays the modern optimization algorithms can provide even the most experienced Masters with beneficial data and prediction of the upcoming voyage.

With up to date weather and sea current forecasts the upcoming voyage can be very efficiently optimized with optimization tools available on the market. The effect of the sea current is highly important and together with wave added resistance influences the resistance and thus energy consumption the most.

Predicting the vessel’s resistance in a given condition is not a simple and straightforward task. Vessel’s performance is constantly affected e.g. by hull fouling and thus the optimization system should be adaptive, based on the measurements from the operating vessel.

Most of the solutions aim for an optimum consumption but also the safety aspect has to be considered. Seakeeping functionality is important and for many types of vessels and operating areas mandatory.

2.2. Floating Position

Typically on a cargo vessel the load planning offers a possibility to optimize the floating position for the upcoming voyage. Optimum floating position is a function of speed and hence the input from the voyage planning process is required.

For floating position the safety aspect is even more crucial. Since for example trim can be partly optimized by only collecting measurements of the propulsion efficiency and dynamic trim the market has some solutions searching for the optimum trim with statistical approach. This has caused already couple of occasions where excessive trimming has caused the longitudinal strength to exceed the limit curves.

For this reason any optimization related to the loading condition and floating position should be done or at least checked with the loading computer, which constantly monitors the vessels stability and strength.

2.3. Effect on Consumption and Emissions

Combined savings from the voyage and floating position optimization can be up to 5 percent in consumed energy and fuel oil.

Since the emissions of SOx, NOx, CO2 and particle matters are in direct relation to the consumed fuel oil consumption the careful planning can be directly used to obtain a green vessel.

3. Monitoring

Monitoring the operations is a crucial part of efficient operations. Giving instantaneous feedback to the operating crew increases the overall awareness about vessel’s safety, performance and environmental impact. Monitoring standard and fleet wide key performance indicators enables benchmarking and for example, sister vessel comparison.

3.1. Crew Awareness

Studies have shown that visualizing for example fuel oil consumption in a car will improve the fuel
oil consumption even up to 15%. Same applies for any operating vessel. Typically the operating crew is not optimally aware of the instantaneous fuel oil consumption, possibilities for improvements or more crucially, about the emissions.

The information should be easily visible for the operating crew. Using digital dashboards with simple indicators such as traffic lights in the front of the bridge is a very good approach.

### 3.2. Key Performance Indicators

Having a standard set of Key Performance Indicators (KPIs) logically organized enables improvement on many separate areas of vessel performance.

It should be noted that these separate KPIs can be grouped to a general performance indicator by using suitable weight factors for each individual meter. Please see an example in the following Figure.

![Logbook example](image.png)

In this Figure some arbitrary KPIs are grouped with weighted average to achieve the Common KPI. In fact each presented KPI in the example should be a weighted result from sub KPIs, such as SOx, NOx and CO2 for the Emissions. Here we see that more significant KPIs (such as cargo/payload) will have higher impact on the Common KPI than the less important ones (such as AC efficiency).

Same KPIs should not only be monitored but also recorded and used in the follow-up which we will discuss next.

### 4. Follow-up

Typical problem in operational reporting and follow-up is the distributed nature of the various measurements and data. Using an operational reporting tool which combines all the relevant data sources to a common database is a good foundation for continuous improvement.

Vessel’s logbooks have a lot of important data for the business intelligence. If this cannot be automatically utilized in reporting it means duplicate or in some cases even triplicate work for the operating crew. Electronic logbook solutions can typically be used to combine the significant data with, for example, fuel flow meter readings. Data collection and follow-up will be much more efficient if the resource writing, calculating and approving the reported values sees the impact of his or her work. For this reason the operating crew should have open access to the reported and collected data.

To further minimize the workload all the data should be automatically transmitted to the centralized database.
4.1. Categorizing and summarizing data

To compare the fleet’s KPIs, the data should be categorized, for example, to group the same class of vessels and the same kind of voyages. In addition to this, the fleet wide comparison can be made by using the normalized consumption readings. Categorizing the reported data by power plant operating profiles would give detailed output for the user onboard trying to optimize the usage of the power plant.

Categories should be available in the reporting software for the user generating the reports. It should be possible to select multiple categories at the same time.

An example of using categories is illustrated in Figure 3 below. The subset of interest, the area highlighted in red, is obtained by selecting specific vessel, operating officer and voyage type.

![Fig. 3: Example of using categories to narrow the source data set](image)

A ready-made set of smart categories helps in creating custom searches to the database and generating the reports for the specified subset, Ignatius (2008).

4.2. Presenting the results

Digital dashboards are commonly used to view some KPIs. In this concept, a simple +/- gauge meter or a traffic light indicator among other simple devices (these are called dashboard devices) are used to show a certain (usually a KPI) measurement. Management should not be bothered with excessive reports or views. If everything is going well a simple green light is typically enough. More detailed information should naturally be available but it is usually used only if the dashboard indicators show either very good or poor performance, Tuefte (2006).

See the following figure for an example of a digital dashboard from a reporting portal called NAPA Office. In this example, you can see traffic lights representing the class fuel oil consumption compared to the forecast in three different time periods. Forecast in this case is based on the budget but it is adjusted as the year progresses. Fig. 4 also visualizes the fuel oil consumption in time series as a line graph.

If the user wants to see vessels the class name can simply be clicked to view separate traffic lights for each individual vessel. The reporting views form a tree like structure as described in Fig. 5. The tree structure unfolds more details as the user browses the tree.

Easy way to create this type of structure is using weighted KPIs as presented in Chapter 4.2 Key Performance Indicators.
Fig. 4: Example of a digital dashboard visualizing fuel oil consumption (highlighted detail zoomed)

Fig. 5: Simplified example of tree type reporting structure.

4.3. Distribution of the results

The IT infrastructure of today allows for a very simple and efficient distribution of the results within the user organization (intranet) and even to related organizations in need of the data (extranet). A web-based BI (Business Intelligence) portal is the perfect solution for many reasons. It can be built up to offer different users different information and furthermore the client software (web browser) is totally free of charge and available nowadays on basically any type of device. Typically the operator, the charterer and the operating crew onboard are interested in somewhat different data.

Providing feedback for the operating crew is important for several reasons. First of all, the motivation for inputting more data into the new application could be low if no absolute results would be available for the crew itself. Once the crew sees that the data is in fact used and the results can be viewed, the motivation is likely to rise. Secondly, a committed chief engineer might try to vary the profile of the power plant and review the reports to find the optimal performance. Comparison with sister ships
would help this evaluation process and in addition the know-how could be spread via a portal where the crew could discuss the performance of their vessels.

4.4. Feedback to Planning

The measurements, key performance indicators and analysis results should be used effectively when improving the operations. For example the decision supports tools can be adaptive and provide even more accurate simulations with the help of observed conditions.

Having any measurement data in a centralized reporting database enables baseline comparison and the use of statistical methods to find correlations and trends.

5. Conclusions

“You can’t manage what you can’t control, and you can’t control what you don’t measure” is the famous quote of DeMarco (1982) and it can be applied to many different disciplines.

Measurements should be done without increasing the workload of the operating crew and control should be achieved by adequate decision support tools.

A significant solution to this problem is to log as much of the needed data as possible through automatic logging. Only data that cannot be automatically recorded should be collected through means that do not increase the workload. The natural way to do this is to upgrade the traditional logbooks to smart electronic logbooks. The filling out of logbooks are legislated by the authorities, and changing the procedures to fill in electronic logbooks will actually decrease the workload onboard.

Setting up a standardized way to collect the operational data from the ships in the fleet will give the possibilities to analyze the data in a comprehensive way. To find key performance indicators and trends is of high importance in the competitive business of shipping today. By using modern IT tools and integrated data management solutions in ship operations it is possible to achieve a higher understanding of fleet operations.

6. Acknowledgements

I would like to express my deep appreciation to Onboard-Napa Ltd for making the writing of this paper possible. NAPA Power, NAPA Loading Computer, NAPA Electronic Logbook and Voyage Reporting teams deserve special thanks for providing me with adequate information about the corresponding problem areas.

References


SJÖFARTSDIREKTORATET (2007), Guideline on the NOx tax, Norwegian Maritime Directorate.

IGNATIUS, J. (2008), Fleet wide operational data in shipping companies' business intelligence, Master’s Thesis, Helsinki University of Technology.


Generating More Valid Designs during Design Exploration

Ties van Bruinessen, TU Delft, Delft/The Netherlands, tmvanbruinessen@tudelft.nl
Hans Hopman, TU Delft, Delft/The Netherlands, j.j.hopman@tudelft.nl
Thomas DeNucci, United States Coast Guard, Delft/The Netherlands, t.w.denucci@tudelft.nl
Bart van Oers, Defence Materiel Organization, The Hague/The Netherlands, bj.v.oers.01@mindef.nl

Abstract

Concept Exploration Models are a powerful tool in the conceptual design phase, triggering naval architects towards unconventional or better solutions. However, the quality of the design-set depends on the evaluation of each concept design and its compliance to customer requirements. To improve design-set quality new evaluation-modules are developed to calculate sea-keeping, weight and resistance and evaluate their compliance with requirements set by the US Coast Guard. To make this possible the MATLAB based CEM was coupled with the sea-keeping program Shipmo2000 from the Marin Research Institute. In a final step the designs were graphically filtered and evaluated.

1. Introduction

The development of Concept Exploration Models (CEM) has made significant process in last few years; these developments were not unnoticed and sparked the interest of several large parties such as the design offices of the Dutch Navy, the US Coast Guard and commercial parties such as Gusto. CEM can be used to generate a wide range of concept designs, exploring the full design space. The resulting design-set can be used to evaluate design decisions or to trigger designers towards unconventional designs with favourable properties, compared to other more conventional designs.

However, the quality of the design-set depends on the concepts and their compliance with the client requirements, Fig. 1. These requirements are usually generated early in the design process and can, for example include cargo capacity, speed or sea-keeping requirements.

Although research shows that CEM are useful tools in early design stages, Oers et al. (2010), Wagner et al. (2010), the software was not able to evaluate the requirements set by the client based on sea-keeping, speed and weight. By evaluating these requirements in the conceptual design phase, the validity and use of the methodology increases, providing designs suited to the desires of the client.

![Concept exploration phase, La Rocca et al. (2002)](image)

CEM research at the Delft University of Technology concentrates on the Configuration Optimization Routine (COR), capable of generating concept designs fast and with a high degree of accuracy, Oers et al. (2010). In this paper the COR is applied to the design of an Offshore Patrol Vessel (OPV) for the US Coast Guard. The Coast Guard provided input and requirements for the vessel, shaping the required evaluation models and generated concepts.

The COR and information about the OPV will be discussed in section two and three, section four...
shows the developed modules, in section five, six and seven a test case is presented and conclusions and recommendations are drawn. The final section will give an introduction to subsequent research.

2. Introduction in the Concept Exploration Model (CEM)

The concept exploration model used during this research is based on the 3-D Configuration Optimization Routine (COR), Oers et al. (2010). The program combines the NSGA-II Genetic Algorithm (GA) with a Space Allocation Routine (SAR), both discussed in subsequent sections. The SAR is the basis of the program, placing the different building blocks and calculating the performance indicators of each design. GA drives the program towards ever improving designs, using similar mechanisms as found in the original evolutionary theory described by Darwin.

![Configuration optimization routine](image)

2.1 Genetic Algorithm

The driving force behind the COR is the off-the-shelf NSGA II algorithm described by Deb (2002). The genetic algorithm uses a string of position information, supplemented by the constraint and objective functions. The genetic algorithm works towards improving designs using the conventional evolutionary theory described by Darwin, where principles of evaluation, selection, cross-over and mutation create improving generations towards the two objective functions. The NSGA-II is capable of handling multiple constraints and objectives, combined with a relatively fast process make the algorithm suited for the optimization of ship configurations.

2.2 Space Allocation Routine

The SAR, the basis of the COR, can be seen as a packing process where a wide variety of objects have to be fitted to a ship-shaped envelope. An extensive description of this methodology can be found in Oers et al. (2010).

![Three-dimensional packing](image)

During previous research, Oers et al. (2010), Wagner et al. (2010), the merits of a 3-D packing approach have been discussed extensively. However, the 3-D approach takes considerable time, even on power-packed 4 or 8 core processors.
During the design exploration phase a slightly modified version was used to generate the wide range of designs required to describe the entire design space on a conventional desktop computer. The modifications will be discussed in the following paragraph.

**2½-D Packing Approach**

The packing approach used during this research is the 2½-D Configuration Optimization Routine. The routine is in essence a side view approach, where objects are placed in X and Z direction, resulting in a filled profile. As ship designs are generally three-dimensional the 2½-D version checks the available width, evaluating if each system can be placed.

This approach combines the applicability of a three-dimensional approach with the calculation speed of a two-dimensional approach. According to DeNucci et al. (2009) the 2½-D approach limits the search and placement matrix to two dimensions (X and Z), reducing the computational effort with the size of the transverse matrix (Y direction): in this particular case approximately twenty-fold. By checking the available width the program can still produce accurate results.

During the design exploration phase the level of detail is secondary to the complete exploration of the design space. Although the level of detail is reduced, the possibility to generate a multitude of designs while still retaining validity increases its use during the design exploration phase.

3. **Offshore Patrol Vessel**

In the last few years the US Coast Guard started a vast improvement strategy which involved over 20 major projects, replacing aircraft, boats and larger vessels throughout their fleet. The vessel in this research is considered to be the replacement for the medium sized offshore patrol vessel. The vessel should have a mission spectrum similar to the old vessels, but updated and adapted to suit modern times.

The following paragraphs will give a short introduction about the vessel and its role, including the requirements set by the US Coast Guard.
3.1 Systems, envelope

The design was modelled based on the new National Security Cutter, recently designed by the US Coast Guard. Installed systems were adapted to suit the needs of the mission statement and the size of the new vessel. The model was structured as such that each design contains the following systems and satisfied the mentioned constraints.

<table>
<thead>
<tr>
<th>Hull</th>
<th>NSC Hull</th>
<th>Weapons</th>
</tr>
</thead>
<tbody>
<tr>
<td>Length</td>
<td>70-140 m</td>
<td>57 mm Bofors Canon</td>
</tr>
<tr>
<td>Width</td>
<td>10-20 m</td>
<td>Phalanx (CIWS)</td>
</tr>
<tr>
<td>Depth</td>
<td>7-15 m</td>
<td>6x .50 weapon systems</td>
</tr>
<tr>
<td>Draft</td>
<td>3-5 m</td>
<td>Sensors</td>
</tr>
<tr>
<td>Speed, max</td>
<td>24 kn</td>
<td>Thales I-Mast 400</td>
</tr>
<tr>
<td>Speed, patrol</td>
<td>12 kn</td>
<td>Vessel Systems</td>
</tr>
<tr>
<td>Boats</td>
<td>Long Range Interceptor</td>
<td>4 engine propulsion system</td>
</tr>
<tr>
<td>Aircraft</td>
<td>Short Range Prosecutor</td>
<td>Gearbox &amp; Exhaust scaled</td>
</tr>
<tr>
<td>Flight Systems</td>
<td>Flight deck, HH65 Dolphin</td>
<td>RAS-Installation</td>
</tr>
<tr>
<td>Flight Systems</td>
<td>Hangar, HH65 Dolphin</td>
<td>Requirement 1: Safe helicopter operations</td>
</tr>
<tr>
<td>Flight Systems</td>
<td></td>
<td>in sea-state 5</td>
</tr>
<tr>
<td>Flight Systems</td>
<td></td>
<td>Requirement 2: Propulsion system tooled to</td>
</tr>
<tr>
<td>Flight Systems</td>
<td></td>
<td>both patrol and maximum speed</td>
</tr>
</tbody>
</table>

3.2 Constraints

The OPV only had an abstract set of requirements at the beginning of this research, some of which were related to systems discussed in the previous paragraph, while others related to the operability of the vessel. The most complex constraint is the sea-keeping requirement, which states that helicopter operations should be possible in sea-state 5. The second constraint was a speed and range requirement, in which each design should be able to maintain a certain service speed and be able to sail a certain distance.

The current program was unable to evaluate these constraints during the generation of the design-set; furthermore one important aspect of the design, the weight, was estimated using a surface based approach with little scientific backing. To solve these problems new modules were developed to calculate weight, predict sea-keeping behaviour and estimate the resistance to size propulsion power: all with the limited data available in the COR.

4. Module Development

In order to generate and evaluate a wide range of feasible designs for the Offshore Patrol Vessel several problems had to be addressed. As mentioned in section 3.2 the US Coast Guard had very specific constraints, which the current version of the code was not able to evaluate in an early stage of the design.

To evaluate the constraints of the US Coast Guard three modules were developed:

1. Sea-keeping Analysis
   The sea-keeping analysis is used to evaluate if each design is capable of launching and recovering aircraft and boats in severe sea-states, the constraint is used as a proof of concept but can be extended to calculate accelerations or other sea-keeping requirements.

2. Parametric power plant sizing
   Customers usually require a predefined range and speed of each design. This causes problems
in the initial COR, which requires a predefined propulsion system. This methodology scales the propulsion power according to the expected resistance.

3. Improved weight calculation
   Initial weight calculation was based on the surface of decks, hull and bulkheads but yielded insufficient accurate results for the entire steel weight of the vessel. This methodology was replaced by a simple and quick calculation module based on the encased volume of the vessel.

4.1 Sea-keeping Analysis

As section 3.1 discussed, many OPV missions require a need to launch and recover aircraft and boats in severe sea conditions. Therefore, sea-keeping performance should be evaluated as early in the design process as possible. Several sea-keeping analyses can only be done at the end of the design process, using expensive towing-tank tests or CFD calculations. Linear strip theory is one of the few methods applicable in an early design stage, giving an indication of the behaviour of the vessel.

At the Delft University of Technology two linear strip theory programs are available, shipmo2000 provided by the MARIN and octopus-office provided by Amarcon. Octopus-office, although more extensive, was unable to communicate with Matlab without creating an extensive interface, requiring access to the source-code of the program. Matlab was able to generate the input files and run Shipmo2000, generating the required Response Amplitude Operators (RAO) for each vessel. The information was used to predict the motions of each vessel using the process shown in Fig. 6.

\[ S_{x}(\omega) = \left| Y_{x}(\omega) \right|^2 \cdot \beta_{x}(\omega) \]

Fig. 6: Linear Motion Theory, Pinkster (2006)

In the first part more information will be given on the linear strip theory. In a second part the process will be discussed, with several detailed points to the wave generation and constraint definition. In the final two parts the complications and conclusions will be discussed.

4.1.1 Linear Strip Theory

Shipmo2000, a software tool developed from the older Shipmo-MARIN, is a sea-keeping prediction program based on linear strip theory. Linear strip theory assumes that the hydrodynamic forces working on the total ship can be approached by integrating hydrodynamic forces on each transverse section over the length of the vessel. Strip theory is linear and cannot predict non-linear, extreme motions, velocities and accelerations which, in some cases, can be more limiting than average values. This can be problematic in situations with non-linear motions, but the method is generally accepted for Froude numbers lower than 0.3 (-) and as an optimization method in early design stages, NATO (2000), Journée (1992).
4.1.2 Process

The process of the linear prediction method is shown in Fig. 7, more information about Shipmo2000 and its use can be found in Daalen (2010).

![Diagram of sea-keeping evaluation process]

**Fig. 7: Sea-keeping evaluation process**

4.1.3 Level of Detail

Section 2.2 discusses the relation between the level of detail, the validity of the design and the possibility to generate a multitude of designs to explore the design space related to the COR. A similar consideration can be made for the sea-keeping analysis: Initially the COR produced a concept design in approximately 2 s on a conventional, single-core computer. After Matlab was able to successfully run Shipmo2000, the sea-keeping analysis slowed the COR down 15 times, taking over 30 s for each design.

During a study of the program it appeared that the calculation time was linearly dependent on the number of frames loaded into the program, Fig. 8, while at the same time influencing the resulting RAOs, Fig. 9. To reduce calculation time and improve design exploration while still retaining valid designs the number of frames was reduced to 30, subsequently reducing the calculation time to 7 s. Furthermore, the calculation was only done when all other feasibility checks (freeboard, stability, etc.) were deemed valid, to reduce calculation time further.

![Plot of calculation time vs number of frames]

**Fig. 8: Calculation time of a sea-keeping analysis of a single design**
4.1.4 Wave Generation and Constraint Definition

As mentioned in section 3.1 the US Coast Guard provided a wave system and related motion constraints in a basic manner: The vessel should be able to perform helicopter operations in mid sea-state 5, resulting in a significant wave height of 3.3 meters (Federal Business Opportunities, 2008). The limit for safe helicopter operations are withdrawn from documents provided by the Joint Chiefs of Staff and summarized in Table II.

### Table II: Helicopter Operation Constraints, N.N. (1997)

<table>
<thead>
<tr>
<th>Constraints</th>
<th>Limit</th>
</tr>
</thead>
<tbody>
<tr>
<td>Maximum RMS Roll</td>
<td>4°</td>
</tr>
<tr>
<td>Maximum RMS Pitch</td>
<td>2°</td>
</tr>
</tbody>
</table>

To generate the applicable wave-spectrum a generic Matlab module was used, capable of generating a wide range of commonly used wave-spectra, such as Bretschneider, Pierson-Moskowitz and JONSWAP. Furthermore, the program was highly adaptable in the level of detail and available input, *Fossen and Perez* (2008). Although other spectra were evaluated, the selected Pierson-Moskowitz spectrum required a minimum of input while still concurring with the requirements set by the Coast Guard.

4.1.5 Potential of the sea-keeping module

In this case, the sea-keeping module is only used to test a very specific constraint: to check if the design is able to perform helicopter operations in sea-state 5, successfully providing a simple constraint during a diverging process. This is by no means the only application: during the project the vertical and horizontal accelerations of the bridge were calculated, although the results were not used it shows the potential of the calculation module.

The position information of each system, supplemented by a full operational profile can be used to calculate the Sea-keeping Performance Index (SPI). Using this - or similar indices - as an objective will not only influence the basic parameters, but also the layout of the vessel in a very early stage of the design process. This means the layout of the vessel can be optimized towards sea-keeping performance when the design is still flexible, generating a new dimension in conceptual design. Further application of the sea-keeping module was outside the scope of this project, but should be tested and verified in subsequent research.
4.2 Parametric Propulsion System Scaling

To provide each design with sufficient speed and range to comply with the requirements of the client, the propulsion system has to be sized in relation to the hull and the installed systems. Although the propulsion system influences the weight and draft of the vessel as a recursive process, Fig. 10, the method provides a first estimation of the propulsion power and engine size based on the known systems, hull size and a resistance database.

Fig. 10: Recursive process

4.2.1 Module Description

The Propulsion system scaling module changes the size of the propulsion plant within a known layout, fulfilling the client’s requirement on speed and range. The module uses a database of resistance information, interpolating the resistance using the known length, width and height of the vessel and an estimated draft parameter, calculating the required power at both the patrol and the maximum speed. The database was generated using a Holtrop-Mennen resistance calculation on 54 different hull sizes (six lengths, three depths, three widths) on three different drafts. For each hull the resistance data from 0 to 40 knots was included. The database can be filled with any ship or hull specific information, providing a better estimation of the resistance.

Fig. 11: Propulsion and fuel tank scaling
4.2.2. Engine Selection

When the required engine power is defined in the module described in the previous paragraph the propulsion train is sized accordingly. When sizing the propulsion system, the configuration and type of engines is fixed, while the size and weight of the components is scaled using power/weight, power/volume and length/width/height ratios. The approach was selected over using a list of finite-example systems because of increased flexibility and decreased vulnerability of small changes having a large impact. This method provides each vessel in the design set with a coherent propulsion system, while still complying with the requirements set by the client.

4.3 Improved weight calculation

The sea-keeping and engine selection methodologies are highly influenced by the weight and Centre of Gravity (COG) of the vessel, but weight calculations in early stage design is very difficult. The weights of some systems are dependent on their deck surface, while others are more dependent of the volume of the object. Within the COR a distinction is made between system weight, which can be found in company documentation to fairly good accuracy and steel weight, which is complex because little is known about the structural properties of the vessel. To calculate steel weight three alternatives were evaluated: Westers, equivalent thickness and the volume based methods. The equivalent thickness method provided an incomplete steel-weight and lacked scientific basing, the Westers method, as a regression based approach lacked information concerning this type of vessels. Eventually the volume based method was selected because the method suited the level of detail and provided a simple, validated and robust method. A more extensive description of the volume based method is provided in the following section.

4.3.1 Volume Based Method

The volume method’s calculation simplicity and available scientific basis was decisive in selecting the method for further research. The volume-based method calculates the weight of vessel based on the total enclosed volume of the ship, taking a weight per m$^3$ into account. The method had one important flaw; the calculation did not provide a vertical and longitudinal centre of gravity, necessary for stability and sea-keeping calculations. In a subsequent step, the longitudinal and vertical centres of gravity were related to the volume centroids of the two main components: the hull and the superstructure. The information was calibrated using information from both Dutch Naval Fleet and Coast Guard vessels. Verification of the weight calculation was based on the available vessels and yielded no extreme and unexpected deviations to as-built weights.

5. Test Cases

The generation of a wide range of designs in a design set is only useful if designs are visualized for the designers, providing a window into the design set. To be able to post-process the generated designs a visualization module was developed consisting of a GUI with four information tabs. The first two tabs provide information about the entire design set, while the third and fourth can be used to visualize individual designs.

The most interesting information can be found in the second tab. The entire design-set is plotted on a histogram, counting the occurrence on each parameter-range. Fig. 12 shows an example of a histogram showing a design set with all designs between 2800 and 5000 t of weight, and its respective occurrence every 100 t.
The most valuable data is not in the peaks of occurrences, which shows local optima, but in a range of designs, and its influence on other parameters. As shown in Fig. 13, the selection of designs between 17-17.5 m of width correlates with a low range in GM. The visualisation was able to select ranges in one or two parameters, visualizing their influence throughout the design space.

In the second part of this section a test-case will be presented to show the merits of this approach and using extensive design-set evaluation in CEM.

5.1 Test case: Combined Diesel and Diesel versus Combined Diesel and Gas

To show the approach two different design spaces were visualized and evaluated, the first design-set was shaped using a conventional propulsion setup with four diesel engines, two large and two small (CODAD). The second design-set replaced the two large diesel engines with a gas-turbine (CODAG).

All designs in both design-sets had to comply with constraints set by the US Coast Guard. The two small diesel engines were used for loitering and slow steaming, while the larger engines are used for transit and full speed conditions.
The change in propulsion system influenced primarily the weight, fuel consumption and installed engine power. The gas-turbine has more fuel consumption; approximately 15% for the entire design space. On the other hand, the draft of the vessel was reduced with an average of 0.2 m, reducing the resistance. Both effects are shown in Fig. 14.

To visualize the design-set 8 designs were selected. These designs are extremes of the objective functions (Weight, Fuel Consumption and Motions) to give an idea of their implications.

Fig. 15: Several designs produced by the COR

The visualization method provides the designer with the means to evaluate a design set, and to compare the implications of design choices. Although not perfect, it can help trigger designers to explore other options.

6. Conclusions

Both the conclusions and recommendations will focus on the use of the COR during the concept exploration phase. No conclusions will be drawn on the design of the modelled OPV as it was used as a test-case for the new calculation modules and the software.

The primary objective of the research was to increase the quality of the design-sets used in Concept Exploration by increased evaluation.

This research has shown that the COR is capable of generating a wide range of designs of increased quality by adding or refining evaluation modules, tooled to the requirements of the client. It is still important to realize that the behaviour of the design set, constraints and objectives can be influenced by extensive knowledge of the program and the processes used. When applied correctly the COR can be used to visualize and quantify design aspects, exploring and comparing design choices.

At this stage, the program is run with a minimum of design rationale, solely constraining each system that it is capable of performing its mission. The lack of rationale results in a wide range of possible solutions which are generally unusable as a template for further research but can trigger designers to explore other, more exotic options.
This paper concentrated on extensive design exploration with limited resources, in some cases reducing calculation time at the cost of detail. When focussing on extensive concept exploration the consideration should be made between validity, level of detail and calculation time. When a choice is made to extend evaluation modules or work in higher detail the calculation time or resources will increase considerably.

7. Recommendations

Recommendations are given on different areas, in general, the COR is a powerful tool which is capable of triggering designers to new and innovative solutions. At this time the software is not user-friendly: it is easy to make small mistakes which have a profound influence on the eventual design. When developing the COR for further use the code has to be combined with a user-interface and re-written for easier usage.

When using the code, it becomes clear that the objective functions and constraints have a huge impact on the results of the COR, if incautious the design space can be tooled towards certain ‘expected’ values, defining and exploiting the definition of the objective functions. The definition of the objective functions, while maintaining diversity can be a very interesting new research, but was well beyond the scope of this research.

When evaluating the specific modules developed in this research the sea-keeping analysis has the most potential. The sea-keeping analysis is, at this stage, only used to calculate roll and pitch motions in a single heading, one sea-state condition. The method is sufficiently flexible to create more extensive motion analysis and consider the influence of motions on the overall layout of the vessel.

The final recommendation concerns the weight calculation module. In this paper the calculation is based on the three available vessels, but as weight is one of the main drivers of production costs the weight calculation can be subject of more extensive research projects.

8. Subsequent Research

As mentioned in the introduction the Concept Exploration Models sparked the interest of several companies. It is seen as a possibility to generate more designs in an early stage, to make sure that the design space is fully evaluated before a concept is selected. Although the Configuration Optimization Routine (COR) used in this research is still a rough version, two main problems could be identified.

(1) The COR is currently based on computable knowledge such as stability and sea-keeping, supplemented by simple, hard constraints required by systems to perform their individual function. More intricate knowledge of system relations, their influence and common sense is essential in generating valid designs which can be used as templates for further design. This knowledge is currently undefined in the software package.

Current research at the TU Delft, *DeNucci (2009)*, aims to address this problem by capturing and using intricate knowledge (rationale) about systems and their relations.

(2) The implementation of this software can cause a second problem. The Ship Design industry is used to a solution-oriented approach, adapting a single solution to suit the needs of a client. The development of new software capable of extensive concept exploration and other advanced calculation modules can have a profound effect on the Product Development in the ship design industry.

Historically, several examples can be found where the implementation of advanced software or design approaches failed because it did not grasp the relation between technology, people and process.
8.1 Socio-Technological Model

The last problem, the implementation of software packages in a commercial environment sparked my own interest. According to preliminary research, product development is mainly dependent on three main aspects: Technology, People and Process.

![Socio-Technological Model](image)

Fig. 16: Socio-Technological Model

Improving product development in a company is not just implementing innovative technology but a complex process of aligning technology with their approach and the people involved. In brief, there is no silver bullet in product development.

This is particularly interesting in the current situation, where software developments in other industries flow towards the Ship Design industry, traditionally a conventional industry, influences not only the technology, but also the process and the people involved.

8.2 Project

The project started in early January in cooperation with Ulstein Sea of Solutions, a pro-active design office involved in ship design for the offshore drilling, construction and production markets.

At this stage the research focuses on the product development approach in the Dutch ship design industry, supplemented by other, unconventional, product development approaches in industries such as car, chemical-process or airplane design. Subsequent steps will be in the identification of key parameters in the ship design industry and eventually generating a new approach in combining technology, people and processes in an effective Product Development System.

References


DeNUCCI, T.; HOPMAN, H.; VAN OERS, B. (2009), Prototype development of a ship rationale capture tool, 16th Int. Conf. of Ship and Shipping Research, Messina, pp.29-38


OERS, B.v.; STAPERSMA, D.; HOPMAN,H. (2010), *A 3D packing approach for the early stage configuration design of ships*, COMPIT, Gubbio


An Integrated Approach for Simulation in the Early Ship Design of a Tanker

Stefan Harries, FRIENDSHIP SYSTEMS, Potsdam/Germany, harries@friendship-systems.com
Fabian Tillig, FRIENDSHIP SYSTEMS, Potsdam/Germany, tillig@friendship-systems.com
Marc Wilken, Germanischer Lloyd, Hamburg/Germany, marc.wilken@gl-group.com
George Zaraphonitis, National Tech. Univ. of Athens, Athens/Greece, zar@deslab.ntua.gr

Abstract

An integrated approach is presented that combines the simulation of key measures of merit in the early phase of ship design. The approach was applied to an Aframax tanker for which payload, steel weight, strength, oil outflow, stability and hydrodynamics were considered simultaneously. Required freight rates, Energy Efficiency Design Index and maximum speed for given engine output were determined so as to rank design variants. Formal exploration and exploitation strategies were utilized to investigate the design space and, subsequently, advance competing design proposals into certain directions such as maximum energy efficiency, attainable speed and environmental protection in case of accidents. The paper focuses on integration and optimization, utilizing the tanker as an elaborated design example to illustrate the holistic view.

1. Introduction

Ship design is often considered a sequential process that is classically pictured as a design spiral, Fig.1. Even though this represents an idealization, the traditional work flow is to study one issue at a time and to advance a design step by step, undertaking modifications and establishing refinements iteratively. Particularly when looking at a complex system with many relationships and dependencies, it is beyond any single individual's capacity to bear in mind all options and consequences. However, an integrated approach can be taken, Fig.1, that brings together key aspects of a design task at the same time. A synthesis model of Computer Aided Engineering (CAE) allows investigating the design space to a greater extent, leading to new insights and promising new options.

Looking at an Aframax tanker for Caribbean trade, a CAE environment was established to examine key measures of merit for a considerable number of variants simultaneously: Payload, steel weight, strength, oil outflow, stability and hydrodynamics were computed by means of sophisticated simulation codes. Required freight rates (RFR), Energy Efficiency Design Index (EEDI) and maximum attainable speed for given engine output were determined so as to judge and rank variants.

---

**Fig.1:** Traditional design spiral (left) vs. integrated approach (right)
2. Overview

Shipping’s major ecological impact stems from energy consumption and associated green-house gas emissions in standard fleet operations. The introduction of the EEDI as put forward by the MEPC (2009) raises both awareness and efforts for higher energy efficiency while high bunker prices continue to excite economic pressure on the operators. A recent study for large oil tankers showed that potential loss of cargo is dominated by grounding and collision along with fire and explosions, MEPC (2008). Enlarged double hull width and double bottom height consequently lead to better environmental protection as elaborated by Papanikolaou et al. (2010).

An analysis using Lloyd’s Register Fairplay WSE Database revealed that one fifth of the existing Aframax tonnage would be older than 15 years by 2012. Even though current tanker capacity appears to outweigh anticipated demand of oil transport, the fleet’s ageing is likely to trigger replacements.

It is therefore safe to assume that new tanker designs will be sought in the near future. However, it is not obvious what will be the main driving forces:

- Safer shipping by containing oil outflow in case of an accident,
- Greener operations by reducing emissions per ton-mile of cargo,
- Smarter business by increasing returns (higher cargo capacity and lower fuel consumption).

A reasonable combination is likely to be favored over an extreme, depending on the specific situation and preference of the stake holders. The more high-quality design data are available the easier it is to understand opposing influences, come to a sound judgment and choose the best compromise.

![Fig.2: General arrangement along with layout of tanks and selected free variables](image-url)
2.1. Design task

So as to have a focused design task Aframax tanker trades in the Caribbean Sea between St. Eustacius (transshipment), Aruba and Maracaibo (source) and the US Gulf region (sink) were selected for elaboration. This not only allowed to create and prove an integrated CAE approach but also to propose interesting novel designs for a ship type of imminent commercial interest. Restrictions of the prevailing shipping lanes, the main US port facilities and the US Emission Control Area (ECA) established important constraints, most notably limits on maximum length, beam and draft and an additional demand for tanks to carry marine gas oil (MGO). Requests from ship operators active in the trade were taken into account. A prominent call was to attain relatively high speeds. Furthermore, major structural modifications that would lead to deviating from recognized Aframax design principles - like cargo tanks without hopper plates - were to be avoided. A conventional 6x2 layout for the tanks was used, Fig. 2. The challenge was to identify designs that would not deviate too much from conventional practice but still yield significant improvements.

2.2. Design approach

The process was set up in the FRIENDSHIP-Framework (FFW), combining POSEIDON, NAPA and SHIPFLOW simulations. The following key measures were computed:

- Cargo tank capacity in full load and design load conditions,
- Steel weight of the cargo tank area,
- Maximum ship speed at design, ballast and scantling drafts,
- Probability of oil spill in case of accidents measured by IMO's oil outflow index (OOI).

A general flow chart is presented in Fig.3. For each variant a hull form is generated within FFW along with optimal tank shapes. The ship structure in the cargo tank area is then determined with POSEIDON in accordance to the prescriptive part of the Common Structural Rules (CSR) for Double Hull Oil Tankers. The hydrodynamic performance is computed via a response surface model (RSM) built from a priori flow simulations using SHIPFLOW with potential flow (XPAN) and viscous (CHAPMAN) analyses. This is followed by a batch mode execution of NAPA to get intact stability and trim characteristics plus the probability of oil outflow on the basis of the current tank shapes and hull form. The process is complemented by several additional features within the FRIENDSHIP-Framework to gather and combine the various outputs from all external simulations.
From cargo tank capacity, steel weight and ship speed two combined performance measures for ecology and economics were derived:

- Operational impact measured by the energy efficiency design index (EEDI), combining engine power, deadweight and ship speed according to IMO,
- Financial attractiveness measured in terms of required freight rate (RFR), combining the annual cost of transport via capital, fuel and other operating costs with the number of roundtrips times cargo mass per year.

Free variables of the overall investigations were parameters that controlled the hull form (outer shell), the tank layout and geometry as well as the inner structure, Table I. Starboard (S) and port (P) parameters were assumed to be the same. As a use case tankers optimized in terms of RFR (cost of transport) were created by variation of the hull form and selected structural parameters, Fig. 2.

<table>
<thead>
<tr>
<th>Free variable</th>
<th>Lower bound</th>
<th>Upper bound</th>
<th>Primary influence</th>
</tr>
</thead>
<tbody>
<tr>
<td>Length over all (LOA)</td>
<td>242 m</td>
<td>250 m</td>
<td>Hull form</td>
</tr>
<tr>
<td>Beam</td>
<td>42 m</td>
<td>44 m</td>
<td>Hull form</td>
</tr>
<tr>
<td>Shift of longitudinal center of buoyancy</td>
<td>-0.008 LPP</td>
<td>0.008 LPP</td>
<td>Hull form</td>
</tr>
<tr>
<td>Block coefficient</td>
<td>0.8</td>
<td>0.885</td>
<td>Hull form</td>
</tr>
<tr>
<td>Depth</td>
<td>20.5 m</td>
<td>23 m</td>
<td>Tank geometry</td>
</tr>
<tr>
<td>Inner bottom height of cargo oil tanks 2 to 6 (S+P)</td>
<td>2.0 m</td>
<td>2.7 m</td>
<td>Tank geometry</td>
</tr>
<tr>
<td>Lifting of inner bottom of cargo oil tank 1 (S+P)</td>
<td>0 m</td>
<td>1.5 m</td>
<td>Tank geometry</td>
</tr>
<tr>
<td>Side shell width</td>
<td>2.0 m</td>
<td>2.7 m</td>
<td>Tank geometry</td>
</tr>
<tr>
<td>Angle of hopper plate</td>
<td>30°</td>
<td>60°</td>
<td>Tank geometry</td>
</tr>
<tr>
<td>Width of hopper plate</td>
<td>4.8 m</td>
<td>5.8 m</td>
<td>Tank geometry</td>
</tr>
<tr>
<td>Shift of the intermediate bulkheads (frame spacing a)</td>
<td>-1 a</td>
<td>+1 a</td>
<td>Inner structure</td>
</tr>
<tr>
<td>Number of frames per tank</td>
<td>7</td>
<td>8</td>
<td>Inner structure</td>
</tr>
</tbody>
</table>

Having established the most favorable main particulars, cargo holds and scantlings, in a global optimization, the ship’s aftbody was subsequently fine-tuned with regard to wake quality and total resistance. In addition, systematic changes were undertaken to study the dependencies of selected measures of merit on specific parameters, e.g. the change of oil outflow probability by further stepping the bottom of the foremost tanks.

3. Parametric models

3.1. Hull form

A fully parametric hull model was developed within FFW for typical tanker hull forms, Fig. 4. The model is divided into forebody, parallel midbody and aftbody. While the forebody and the aftbody are created using meta-surfaces, the parallel midbody is a simple ruled surface for connection.

Basic curves for points, tangents and integral values are employed to define the shape of the hull surfaces. The basic curves depend on global variables, e.g. length between perpendicular (LPP), and local variables which influence only small regions. The shapes of the basic curves are controlled by specifying the tangents at their start and end positions, respectively, as well as specific areas between the curve and an axis of reference. In special cases, for example the waterlines in the aftbody, additional points in the middle are utilized along with associated tangent information.

The forebody is realized using one single meta-surface with rotating sections, with the center of rotation at the intersection of the aft end of the forebody, the midship plane and the flat of bottom. In
the aftbody region several surface patches are combined, using sections (x constant) as input to the meta-surfaces except for the aft bulb which features a surface built on waterlines (z constant) to ensure tangent continuity at the transition to the adjacent surface.

Fig.4: Fully parametric hull model for a tanker

For hydrodynamic analyses, see section 4.2, the length, beam, longitudinal position of the center of buoyancy (XCB) and displacement volume were changed systematically. While length and beam of the hull form are global parameters of the fully parametric model, the variations of XCB and displacement were realized by means of a Generalized Lackenby for partially parametric modifications, Abt and Harries (2007). Local parameters defining the shape of the aftbody's basic curves were changed during the hydrodynamic fine tuning. In this phase 12 local parameters were varied, for instance the fullness of the diagonal starting in the forward clearance point, the forward clearance of the propeller and the fullness of the aft bulb curve in the midship plane.

An existing geometry following previous studies by Papanikolaou et al. (2010) was taken as a good starting point for the design task. The parametric model was adjusted to closely resemble the existing hull form. Realizing a new variant then simply meant changing the selected set of parameters.

3.2. Tank arrangement

The cargo tanks were generated within the FFW using feature technology, e.g. Brenner (2009). The tanks are computed such that maximum cargo volume is realized while ensuring a minimum distance to the hull form, e.g. 2 m. The feature takes the hull form, the minimum distance of the inner structure to the hull (outer shell) and the longitudinal position of the engine room's bulkhead as inputs. The collision bulkhead's position is computed according to IMO rules.

Fig.5: 6x2 tanks generated within the FRIENDSHIP-Framework (FFW)
During the global optimization the side shell width at deck height, the double bottom height at amidships, the angle and width of the hopper plate and the step in the double bottom towards the foremost tank were changed. The bulkhead positions were moved discretely according to the frame positions. The total number of frames was controlled by specifying the number of frames per tank. The first tanks (COT1) and the last tanks (COT6) were flexible in length by allowing shifts of the bulkhead positions by one frame distance forward or aft, Fig. 2. The tanks associated with a specific design variant were represented as an assembly of planar surfaces within the FFW, Fig. 5, and transferred to NAPA by means of the edge points for the bulkheads and hopper plates.

3.3. Structural model

For strength assessments a computational model containing all CSR relevant structural information was needed. The model had to include information about the main particulars of the vessel, plate distribution and stiffener arrangement of primary and secondary members, tank arrangement and load definitions.

For the design process at hand it was decided that only those elements should be addressed that would influence the key measures significantly. Therefore, there was no need to make all strength relevant data readily accessible for the actual design task which tangibly reduced data management. Nonetheless, the structural information had to be prescribed for all strength assessments. This was done by providing the principle steel design externally as a POSEIDON template database that could be transformed into a complete structural design by insertion of the current set of free variables. This template database specified the steel structure of the cargo tank area of an Aframax tanker with 6x2 layout and a plate arrangement and stiffener distribution complying with a conventional design, Fig. 6:

- Vertically stiffened flat transverse bulkheads with transverse girders,
- Longitudinally stiffened main deck, hopper plate, inner hull, inner bottom, stringer decks, longitudinal girders,
- Longitudinal bulkhead stiffened with transverse girders,
- Regularly positioned web and floor plates,
- Main deck supporting transverse girders.

Fig.6: Hull structure modeled within POSEIDON (main deck removed to show inner structure)
Using a Python interface to POSEIDON’s database, the template model could be modified according to the current design instance. An ASCII file was provided by the FFW which included an adaptation of the hull form in POSEIDON’s specific offset format, the actual tank compartmentation and the free variables for the inner structure like the number of frames per cargo tank.

4. Analyses and simulations

4.1. Structure and strength

For the design task the Common Structural Rules for Double Hull Oil Tankers had to be applied with their different levels of assessment. CSR start with prescriptive rules based on beam theory which are followed by Finite Element Analyses (FEA) of primary and secondary members and then finish with detailed FEA for fatigue assessment of structural details in a hot spot approach.

Here, only the prescriptive part of the CSR was applied to determine the strength of the structure. In this sense the proposed integrated approach yields a "pre-dimensioned" tanker design that needs to be approved - and slightly adapted - in a subsequent step to comply fully with the CSR. The reason behind this is that model generation for FEA is a rather sophisticated undertaking in its own right and that corresponding simulations need considerable resources. It was therefore decided to utilize the prescriptive part to rank variants according to their overall properties within the optimization process.

Each design variant was measured in terms of the steel mass necessary to fulfill the strength requirements. The steel mass computation was performed by POSEIDON's automatic plate sizing capability at given cross sections of the vessel. Characteristic frame cross sections like the main frame or transverse bulkheads, Fig. 7, were chosen to obtain the steel mass of the total cargo region.

4.2. Hydrodynamics

Since the Computational Fluid Dynamics (CFD) simulations are the most resource intensive of all analyses within the design task, response surface models (RSM) were utilized to capture resistance and propulsion characteristics for different speeds and drafts. In other words: Rather than to include a very time-consuming full CFD simulation for each variant during the overall optimization the hydrodynamics was pre-computed and then replaced by suitable meta-models.

Four free variables were chosen, namely length over all (LOA), maximum beam, a relative change in the position of the longitudinal center of buoyancy (Delta XCB) and the displacement volume. As summarized in Table II these variables were allowed to vary within meaningful bounds that stemmed from general constraints (like relevant harbor facilities in the Gulf of Mexico), pure hydrodynamic considerations and estimates for expected total displacement.
Hydrodynamic performance was considered at design draft (13.7 m on even keel at rest), scantling draft (14.8 m on even keel) and ballast draft (6 m at FP and 8 m at AP) in parallel. The fully parametric hull model, Fig. 4, was utilized to vary the free variables globally.

<table>
<thead>
<tr>
<th>Free variable</th>
<th>Lower bound</th>
<th>Lower bound</th>
</tr>
</thead>
<tbody>
<tr>
<td>Length over all</td>
<td>242 m</td>
<td>250 m</td>
</tr>
<tr>
<td>Beam</td>
<td>42 m</td>
<td>44 m</td>
</tr>
<tr>
<td>Delta XCB</td>
<td>-0.90%</td>
<td>0.90%</td>
</tr>
<tr>
<td>Displacement volume</td>
<td>126 075 m³</td>
<td>136 325 m³</td>
</tr>
</tbody>
</table>

Both potential flow and viscous computations were performed using the zonal approach offered within the flow solver SHIPFLOW. A sequence of computations was undertaken: A potential flow computation without free surface for the entire hull (XPAN), a subsequent thin boundary layer computation for the forebody (XBOUND) and, finally, a RANSE computation for the aftbody (CHAPMAN). The propeller was modeled as a force actuator disk, idealizing an active propeller for all computations. All viscous computations were performed at full-scale Reynolds number with the model free to sink and trim. For each valid variant, the viscous flow computations provided the frictional and viscous pressure resistance as well as the wake field in the propeller plane. Additional potential flow computations including nonlinear boundary conditions at the free surface were carried out to obtain the wave pattern resistance, Fig. 13.

In all computations - be it for building the response surfaces or for the final aftbody refinement - the same panel meshes and volume grids were employed. For the potential flow analysis, a body mesh with 1150 panels and a free surface mesh with 7175 panels were used. The volume mesh for viscous simulations featured 1.7 million cells with a longitudinal stretch toward smaller cells in the skeg region. An impression of panels and grids can be gotten from Fig. 8.

In order to achieve convergence, 3000 iterations were done for the RANSE solutions of globally changed variants and also for the baseline of the succeeding fine-tuning. One of these computations including potential and viscous flow simulations took about 8 h on a quad core 4x3.0 GHz AMD workstation. Subsequent computations for only locally changed variants, as created during the hydrodynamic optimization, were restarted from the baseline solution with some additional 800 iterations. The restarted computations then only took about 2.5 h each.

For the response surfaces, a total of 486 variants were investigated via an ensemble investigation, i.e. a systematic coverage of the design space. Thereof 270 variants fulfilled the inequality constraint on maximum block coefficient to be less than or equal to 0.88. For these 270 variants the CFD analyses were launched, using the FFW for process control. Six SHIPFLOW computations were needed for each variant to cover two speeds at three drafts (design, scantling and ballast). The speeds were
chosen individually for each draft such that the lower speed would lie below the expected attainable speed while the higher speed would be a bit above the anticipated maximum.

Three response surfaces were finally built, one for every loading condition, assuming quadratic speed-power relationships. The attainable speeds were determined for fixed power installed of 13 560 kW. This value corresponds to a MAN 6S60MC-C at around 100 rpm as a representative engine for Aframax tankers. An engine output of 85% MCR and a sea margin of 10% were assumed.

Power delivered was computed from effective power and an individual estimate of propulsive efficiency. This was done using a wake quality index, called the SVA criterion as discussed in Fahrbach (2004), multiplied with the ideal propeller efficiency as proposed by Tillig (2010). Thus, the power delivered was dependent on the total resistance and the wake quality. Therewith the attained speed on all loading conditions could be determined for all variants created during the global optimization. (The approach's validity was checked and confirmed by means of separate numerical resistance and propulsion tests in SHIPFLOW to establish wake fractions and thrust deduction factors. A Wageningen optimal propeller could then be selected which showed that the obtained speed estimates were conservative.)

The response surfaces were produced employing a Kriging approach with anisotropic variograms, Tillig (2010) and Harries (2010). The Kriging algorithm ensures that sample points are interpolated while oscillations of the RSM are avoided. Interpolation values are computed using a weighted sum of all samples on the basis of the variograms. Here a variogram for each variable direction was produced and the total variogram value of the interpolation point was obtained as a weighted sum of all directional variograms. Thus the influences of the different variables on the function value of interest were captured. The RSM was realized as a Scilab application for variogram creation and as a Python script for the function value estimation - which was later called from the FRIENDSHIP-Framework during the optimization.

Utilizing the three response surfaces it was possible to estimate the attainable speeds at ballast, design and scantling draft directly for a specified power installed, instead of performing an iterative CFD based search. Each RSM analysis thus took about one minute per variant instead of one to two days of full CFD simulation.

4.3. Stability and accidental oil outflow

Compliance with the regulatory requirements for stability and oil outflow was determined within NAPA on the basis of actual tank shapes and hull forms as provided by the FRIENDSHIP-Framework. The hull form is transferred to NAPA using a standard iges format representation. A set of parameters is taken as input to recreate the exact geometry of the inner hull and watertight subdivision. Suitable NAPA macros were developed, facilitating the calculation of the mean oil outflow index as well as the assessment of intact and damage stability requirements and the regulatory and operational trim and draft constraints in the various loading conditions.

Resolution MEPC.117(52) was taken as the regulatory basis for the evaluation of design variants. Regulations 18, 19, 23, 27 and 28 set the requirements for the segregated ballast tanks capacity, the double hull arrangement, accidental oil outflow and transverse stability in intact and damaged condition. For example, for crude oil tankers of 20 000 tons DWT, Regulation 18 calls for sufficient capacity of segregated ballast tanks (SBT), so that the ship may operate safely on ballast voyages without recourse to cargo tanks for water ballast. The capacity of SBT shall be at least such that, in any ballast condition at any part of the voyage, including the conditions consisting of lightweight plus segregated ballast only, the ship's drafts and trim can meet the following three constraints: Molded draft amidships ≥ 2.0+0.02 L, trim by the stern ≤ 0.015 L and draft aft (Taft) always yields full immersion of the propeller(s). Additional requirements come in via Regulation 19 for ballast tanks (or spaces other than tanks carrying oil), effectively protecting the cargo space with various minimum dimensions.

419
The accidental oil outflow performance of oil tankers of 5 000 tons DWT and above, delivered on or after the 1st of January 2010, is to be evaluated according to Regulation 23, based on the so-called non-dimensional oil outflow parameter or, shorter, oil outflow index (OOI). The upper limit of the mean oil outflow depends on the total volume of cargo oil tanks of the ship. In particular, for ships with a total volume of cargo oil tanks at 98% filling less than 200 000 m³, as is the case for Aframax tankers, an OOI value not exceeding 0.015 is required. In other words, statistically no more than 1.5% of the total volume of the oil tanks shall be lost.

The oil outflow is calculated independently for side and bottom damages and then combined in non-dimensional form. The calculations of the mean outflows for side and bottom damage are based on a probabilistic approach, Table III, and takes probability distributions for side and bottom damage cases as input. Finally, Regulation 27 sets the intact stability criteria when at sea in the same form that is applicable to most types of ships. In addition a minimum meta-centric height (GM) of 0.15 m after correction for free surface effects is required at port to ensure minimum stability while loading or unloading. The maximum damage extent for side and bottom damage, along with the corresponding stability requirements in damaged condition are defined in Regulation 28. All these regulations were accounted for in a batch mode execution of NAPA, making them part of the simulations within the optimization.

<table>
<thead>
<tr>
<th>Table III: Quantities needed for computing the oil outflow index</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Quantity</strong></td>
</tr>
<tr>
<td>Oil outflow index (mean outflow)</td>
</tr>
<tr>
<td>Mean outflow for side damage in m³</td>
</tr>
<tr>
<td>Mean outflow for bottom damage in m³</td>
</tr>
<tr>
<td>Total volume of cargo oil tanks (incl. slop tanks and fuel tanks located within the cargo block length) at 98% filling in m³</td>
</tr>
<tr>
<td>Corresponding outflow, assumed equal to the volume in cargo tank i at 98% filling, unless proven that significant cargo volume will be retained</td>
</tr>
<tr>
<td>Probability of breaching a compartment in case of side damages (analogous procedure applies for the probability of breaching a compartment in case of bottom damage)</td>
</tr>
<tr>
<td>Probabilities of damage extending into the longitudinal, vertical and transverse limits of the tank, respectively</td>
</tr>
<tr>
<td>Probabilities of damage lying entirely forward, aft, above or below or entirely outboard of each tank</td>
</tr>
<tr>
<td>Mean outflow for bottom damage at zero tide condition and at 2.5 m tide condition</td>
</tr>
<tr>
<td>Factor accounting for the partial capturing of oil flowing out from a tank in the double bottom</td>
</tr>
</tbody>
</table>

5. Selected results

5.1. Exploration

During the course of the design work approximately 2500 variants were generated and assessed. To start with a Design-of-Experiment (DoE) for the exploration of the global design space was performed, yielding a database with all relevant simulation outputs and the key measures of merit, namely RFR, OOI and EEDI. A conventional Aframax tanker served as a reference (baseline) for comparison and normalization, Table IV. For identifying the tankers with attractive economic
performance for instance, the design variants were ranked according to RFR, Fig. 9. Naturally, any other preference of the decision makers can be considered and the two best designs for OOI and EEDI, respectively, are marked in Fig. 9, too. The diagram shows that the cost of transport (normalized RFR) falls with rising deadweight (DWT) until a certain minimum is reached. Cost of transport could be reduced by about 4%. The performance of the heaviest tankers is slightly less attractive with regard to RFR but the tanker with lowest EEDI is found among them. The best tanker with lowest OOI turns out to be among the smaller designs with a slight penalty in RFR of ~2%.

Normalization was done with the baseline's data to gain a certain independence from current price levels and their volatility. The RFRs were determined via a roundtrip model for the Caribbean trade on the basis of contemporary cost levels. (Capital costs were based on a newbuilding price of 65 Million $, 25 years of lifetime and an interest rate of 8%. Fuel costs were computed with HFO at 500 $/t and MGO at 800 $/t for the transfer within the ECA. Other operating costs were approximated with 3 Million $/year and presumed to be independent of the variations.)

Fig.9: Designs established by means of the integrated CAE approach

Fig.10: Hull form of favored design

5.2. Refinements

Since quite a few designs produced during the DoE offered nearly the same RFR, see Fig.9, the variant with the best OOI among them was selected for further refinements. A local hydrodynamic optimization, utilizing a deterministic search strategy, was undertaken for the aftbody, focusing on the quality of the wake field as an objective. The aftbody was allowed to change such that the impact on the cargo tanks previously established in the global optimization was negligible. The fine-tuning of the hydrodynamics yielded a further increase in speeds V such that the tanker would be expected to attain 15.6 kn at design draft and 16.8 kn at ballast draft with a level of confidence of ±1.3% V. The main characteristics of this favored design are summarized in Table IV and compared to the reference design. The associated hull form is presented in Fig.10. The lines stem from the parametric model and were realized within the FFW without further interactive work, i.e. they are a direct outcome from the optimizations.
Table IV: Main particulars of reference and favored design

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Reference design</th>
<th>Favored design</th>
</tr>
</thead>
<tbody>
<tr>
<td>Length over all</td>
<td>250 m</td>
<td>250 m</td>
</tr>
<tr>
<td>Beam</td>
<td>44 m</td>
<td>44 m</td>
</tr>
<tr>
<td>Depth</td>
<td>21.0 m</td>
<td>21.5 m</td>
</tr>
<tr>
<td>Design draft</td>
<td>13.7 m</td>
<td>13.7 m</td>
</tr>
<tr>
<td>Block coefficient</td>
<td>0.83</td>
<td>0.85</td>
</tr>
<tr>
<td>Inner bottom height COT 2-6 (S+P)</td>
<td>2.50 m</td>
<td>2.10 m</td>
</tr>
<tr>
<td>Inner bottom height COT 1 (S+P)</td>
<td>2.50 m</td>
<td>2.75 m</td>
</tr>
<tr>
<td>Side shell width</td>
<td>2.50 m</td>
<td>2.65 m</td>
</tr>
<tr>
<td>Angle of hopper plate</td>
<td>50°</td>
<td>37°</td>
</tr>
<tr>
<td>Width of hopper plate</td>
<td>5.25 m</td>
<td>5.20 m</td>
</tr>
<tr>
<td>Frame spacing</td>
<td>3.780 m</td>
<td>4.400 m</td>
</tr>
<tr>
<td>Shift of bulkheads</td>
<td>0 m</td>
<td>0 m</td>
</tr>
<tr>
<td>DWT</td>
<td>111 436 t</td>
<td>114 923 t</td>
</tr>
<tr>
<td>Maximum cargo volume</td>
<td>124 230 m³</td>
<td>129 644 m³</td>
</tr>
<tr>
<td>OOI</td>
<td>0.0138</td>
<td>0.0142</td>
</tr>
<tr>
<td>Speed at design draft</td>
<td>15.1 kn</td>
<td>15.6 kn</td>
</tr>
<tr>
<td>Speed at ballast draft</td>
<td>15.9 kn</td>
<td>16.8 kn</td>
</tr>
<tr>
<td>EEDI</td>
<td>3.541 g CO₂ / (t nm)</td>
<td>3.281 g CO₂ / (t nm)</td>
</tr>
</tbody>
</table>

5.3. Sensitivities

In order to understand the robustness of the established design with regard to small modifications a separate DoE was performed. About 150 additional variants were generated whose free variables changed within ±1% of the corresponding parameters of the favored design. Fig.11 presents a selection of sensitivities, with changes in RFR displayed in the upper row and changes in OOI and EEDI in the middle and lower row, respectively. The favored design can be regarded as a (local) optimum for RFR while in its vicinity only few variants perform slightly better with regard to OOI and EEDI. In general, the sensitivity is quite small. This indicates that the favored design does not represent an extreme breed for just one purpose.

Fig.11: Sensitivity of best RFR design (marked by red bullets, band width of abscissas ±1%)
5.4. RFR-OOI study

The relationship between RFR and OOI was further investigated, again utilizing the integrated CAE approach. The tank geometry was systematically varied within the bounds summarized in Table V while freezing all other variables at the values of the best RFR design. Fig.12 opens a view on the compromise between economy (ordinate) and safety (abscissa). The smaller the accidental oil outflow the higher the cost of transport. This is not unexpected but the diagram quantifies how much an operator needs to pay for a safety margin beyond the regulatory limit set by MARPOL. Relaxing the normalized RFR from 0.961 to 0.966, i.e. taking just 3.4% gains instead of 3.9% in comparison to the reference tanker, leads to a further reduction of OOI from 0.015 to 0.012. In Fig.12 the design called best RFR is highlighted. It is evident that this design is a good solution for both economic performance and environmental safety. Fig.13 offers a synthesized impression of the resulting ship.

Table V: Free variables and their bounds for RFR-OOI study

<table>
<thead>
<tr>
<th>Free variable</th>
<th>Lower bound</th>
<th>Upper bound</th>
<th>Primary influence</th>
</tr>
</thead>
<tbody>
<tr>
<td>Inner bottom height of cargo oil tanks 2 to 6 (S+P)</td>
<td>2.1 m</td>
<td>3.0 m</td>
<td>Tank geometry</td>
</tr>
<tr>
<td>Lifting of inner bottom of cargo oil tank 1 (S+P)</td>
<td>0.2 m</td>
<td>2.0 m</td>
<td>Tank geometry</td>
</tr>
<tr>
<td>Side shell width</td>
<td>2.1 m</td>
<td>3.0 m</td>
<td>Tank geometry</td>
</tr>
<tr>
<td>Angle of hopper plate</td>
<td>30°</td>
<td>60°</td>
<td>Tank geometry</td>
</tr>
<tr>
<td>Width of hopper plate</td>
<td>4.0 m</td>
<td>6.0 m</td>
<td>Tank geometry</td>
</tr>
</tbody>
</table>

![Diagram of normalized RFR vs. Oil Outflow Index](image)

Fig.12: Economics vs. safety in Aframax tanker design

6. Conclusions

An integrated approach was developed that simultaneously covered all relevant aspects of early ship design: main dimensions, hull form, hydrodynamics and powering; structures, strength and weight estimates; safety including intact and damage stability; economics; and regulatory requirements. An example application was presented for an Aframax tanker with the aim of realizing better safety (lower OOI), efficiency (lower EEDI) and economics (lower RFR). Formal explorations and exploitations were combined to investigate the design space and, subsequently, advance competing design proposals into certain directions. About 2500 variants were realized, each instance having its individual hull form (outer shell), tank compartmentation and an inner steel structural system.
The integrated system brings together sophisticated software systems for analysis and simulation. Challenging issues, like CFD simulations, can be replaced by systematic numerical series and suitable meta-models (RSM). This not only speeds up the time needed for investigations by several orders of magnitude but it also reduces the complexity associated with CFD analyses and, hence, allows to already utilize them early in the process when gains are potentially the highest.

The presented example showed that once a (quasi-randomly created) database of variants is available it is quick and easy to search for the preferred combination of measures of merit. One may then choose a more conservative design, being a balanced all-rounder, or deliberately decide to favor a more extreme solution, featuring excellent performance in one measure of merit. Additional investigations can be done easily once the CAE environment is established, for instance to gain an appreciation of the relationship between costs and safety or to check the robustness of the favored design.

Setting up an integrated approach still requires quite some effort at this point in time. Nevertheless, the necessary software is available and the presented project proved feasibility. Major prerequisites are parametric models which allow automation. Significant design improvement can then be realized even for moderate deviations from established design practice.

**Acknowledgement**

The authors would like to warmly thank their colleagues directly involved in the project, most importantly Evangelos Boulougouris, Mattia Brenner, Kevin Coyne, Michael Haasis, Peter Horn, Apostolos Papanikolaou, Henner Eisen and Pierre Sames.
References


ABT, C.; HARRIES, S.; WUNDERLICH, S.; ZEITZ, B. (2009), Flexible tool integration for simulation-driven design using XML, generic and COM interfaces, 8th COMPIT, Budapest

ABT, C.; HARRIES, S. (2007), A new approach to integration of CAD and CFD for naval architects, 6th COMPIT, Cortona

BRENNER, M.; ABT, C.; Harries, S. (2009), Feature modelling and simulation-driven design for faster processes and greener products, ICCAS, Shanghai


HARRIES, S. (2010), Investigating Multi-dimensional Design Spaces Using First Principle Methods, 7th Int. Conf. High-Performance Marine Vehicles (HIPER), Melbourne


MEPC (2003), Resolution MEPC.110(49) - Revised Interim Guidelines for the Approval of Alternative Methods of Design and Construction of Oil Tankers Under Regulation 13F(5) of Annex I of MARPOL 73/78, MEPC, International Maritime Organization, London


Design Sketching –
The Next Advance in Computer Aided Preliminary Ship Design?

Richard Pawling, UCL DRC, London/UK, r_pawling@meng.ucl.ac.uk
David Andrews, UCL DRC, London/UK, d_andrews@meng.ucl.ac.uk

Abstract

Sketching can be seen as a technique employed by both designers of Physically Large and Complex (PL&C) systems, typified by the most novel and complex ships, and architects of major buildings and urban structures have employed. However, the HCI concepts implemented in advanced CAD tools for preliminary design have until now inhibited the use of an exploratory sketching approach. This paper considers whether the historic approach by designers to use sketching for design exploration and communication of ideas can still be met by CAD based design. Proposals are presented which could lead to a more responsive and innovative approach to the preliminary design of marine and other PL&C systems.

1. Introduction

Several decades ago it was stated by Gallin (1973) that “ship design without the computer was no longer imaginable”. Future progress in the practice of marine design methodology will in large measure arise from developments in computer technology but also from what practitioners adopt. Nowacki (2009) in a wide ranging review of marine design methodology review started with identifying the goals in computer technology, which had been “dreamt” of back in 1968. He saw a crucial issue to be that of man-machine interaction and to be two fold: assisting in problem formulation (which has been identified in initial ship design as that of requirement elucidation, Andrews (2010), and strategic decisions (that familiar aim in design of better decision making). By 2009 Nowacki saw that many of the goals envisaged for CASD seemed to have been achieved but importantly, in highlighting man-machine interaction, he called for “Human intervention should concentrate on creative problem formulation and critical review of results.” It is the former of which is seen to be the focus of this position paper. Pertinent to this also are Nowacki’s comments on future trends, particularly on “Design as learning” with the injunction to “Apply simulation and visualisation to illustrate causes of design effects”. This then needs to be married with the desire to sustain creativity and innovation, both of which are inherent in sketching, given the likelihood that advances in simulation and visualisation in design are likely to significantly expand with greater use of virtual reality.

2. The Nature of Sketching

A lot has been written over the years with regard to the nature of sketching in design and it is considered useful to group these comments into three broad areas of sketching theory, engineering design and architectural views on sketching in the architectural design process.

2.1 Sketching theory

If we take a “sketch design” to mean any rough, early stage graphical description of the design, then this has (historically) usually meant an image produced with a pencil and paper. However, in recent years, this restriction to a specific physical output by the designer has been broadened to suggest that the act of sketching is itself a creative act beyond just recording the already conceived image of the new form, Goldschmidt (1991), Arnheim (1993). Thus what might be called “Thinking sketches” are seen to be part of a process of internalisation, distinct from the sketches for communication utilised later in developing the design. This has been described as:

“the oscillation of arguments which brings about gradual transformation of images ending when the designer judges that sufficient coherence has been achieved”, Goldschmidt, (1991)
More recent work by Goldschmidt and Smolov (2006) identified the importance of visual stimuli on the creative process in design when viewing the design process as problem solving. Thus given different types of visual stimulus available to the designer they concluded that, for ill-structured problems, the designer’s environment strongly affects the nature of the design. This emphasis on the designer’s environment echoes Darke’s (1979) survey of architectural practice where the architects seek a “primary generator” as the key to commencing an architectural design. Then again Van der Lugt (2005) proposed three main types of sketches, the first being the “thinking sketch” as part of the mental process of re-interpreting ideas and generating new ones.

In a further approach to identifying what is the inherent nature of the sketch both Buxton (2006) and Gross (2006) summarised the main properties of sketches, from the point of view of their use in the early stage design process. Table I lists the main features of sketches they identify.

Table I: The distinguishing properties of sketches, summarised from Buxton (2006) and Gross (2006)

<table>
<thead>
<tr>
<th>Fluid</th>
<th>The designer can easily move from sketches to more detailed schematics</th>
</tr>
</thead>
<tbody>
<tr>
<td>Quick</td>
<td>Quick to make</td>
</tr>
<tr>
<td>Timely</td>
<td>Can be provided when need</td>
</tr>
<tr>
<td>Inexpensive</td>
<td>Cost must not inhibit concept exploration</td>
</tr>
<tr>
<td>Disposable</td>
<td>Investment in the sketch is the development concept represented, not the execution of the drawing</td>
</tr>
<tr>
<td>Plentiful</td>
<td>Meaning and utility of sketches is usually as part of a series of sketches</td>
</tr>
<tr>
<td>Ambiguity</td>
<td>Intentionally ambiguous, sketches gain value from being interpreted in different ways</td>
</tr>
<tr>
<td>Forgiving</td>
<td>Sketches can contain errors, or be under-specified</td>
</tr>
<tr>
<td>Constrained resolution</td>
<td>Sketches do not go beyond “good enough”</td>
</tr>
<tr>
<td>Appropriate degree of refinement</td>
<td>The sketch does not suggest a greater degree of design refinement than actually exists at that point in the design</td>
</tr>
<tr>
<td>Suggest and explore rather than confirm</td>
<td>Sketches provide a catalyst for further development</td>
</tr>
<tr>
<td>Functional</td>
<td>Sketches contain enough information to allow an evaluation of the design</td>
</tr>
<tr>
<td>Clear vocabulary</td>
<td>Certain conventions are used to distinguish a sketch from other renderings</td>
</tr>
<tr>
<td>Distinct gesture</td>
<td>Open and free</td>
</tr>
</tbody>
</table>

Gross (2006) makes a key observation that sketching can be regarded as a “kind of thinking”, which is supported by graphical representations and certain types of drawing, rather than the process being defined by the methods adopted. Gross applies this concept – that sketching is the process, not the method – to the field of programming, concluding that some programming languages support the sketching approach by virtue of having some of the properties contained in Table I.

2.2 Engineering practice

In his seminal book emphasising intuition and the non-verbal in engineering practice, Ferguson (1993) having stated:
“designers use sketches to try out new ideas, to compare alternatives, and (this is important) to capture fleeting ideas on paper.”

also identified three types of sketch:-
- “thinking sketch .. to focus and guide non-verbal thinking”
- “prescriptive sketch .. by an engineer to direct a drafter in making a finished drawing “
  (‘It is the chief engineer’s method of design” (Ferguson quotes this as having been stated in 1941 and again reconfirmed in 1986, well into our era of CAD.))
- “talking sketch … spontaneously drawn during discussions with colleagues”
Of these the first may be seen as primarily creative, while the latter two are about different forms of communication in the collaborative process typical of not just preliminary engineering design but throughout engineering design.

Fallman (2003) from the stance of human-computer interaction sees sketching, even if it is “the archetypical design activity”, as primarily design thinking:-

“not simply an externalisation of ideas already in the designer’s mind, but on the contrary a way of shaping new ideas.” a “dialectic” in which “the sketch itself takes on..’middle ground’ between the designer’s vision.. and how it becomes realised into a coherent whole”

and thus

”sketching is a process, a kind of enquiry.. (not) simply a matter of externalisation” and

“sketching to work out a coherent whole .. externalise .. interpret”

2.3 Architectural practice and theory

Sketching or certainly drawing has traditionally been seen as the architect’s primary mode of design and in his recent book on understanding architecture through drawing, Edwards (2008) towards the end has a chapter on designing through drawing, which commences with a quote from Alberti that encapsulates the issues of design concept and the visual realisation:-

‘The idea as formed in the mind was imperfect and could only be given its consequent form through examination, exercise of judgement, and modification of the idea through drawing.’

Edwards sees five types of architectural drawing:-

• the initial diagram – “mere lines and fragmentary sketches”;
• the sketch design – “initial physical form often exploring variations”;
• technical and preparatory details – “analytical and technical”
• site and other context studies – “records and analysis of site or questions posed by the brief”;
• the definitive drawing – “a representation of the architectural concept” (like an artist impression of a worked up ship design).

Of these really the first two are perhaps akin to the thinking sketch and the others different versions of the designer communicating with others in the wider team or the client. He then goes on to question whether “hand sketches are really appropriate in an era of CAD:-

“are freehand drawing and concept sketching the means (of marrying) art and technology necessary in the 21st Century”..(since) ..“can drawings truly be a vehicle for shared problem solving and more importantly shared form evolution?.. Advocates of CAD argue the screen (gives) more effective design collaboration and (is) a more democratic process” and “(does) drawing still retain this central position in face of considerable development of ..such as CAD”

Edwards reinforces this scepticism by listing specific criticisms of architectural drawing in that it “restricts architectural speculation” so that he prefers CAD “as a means of operating in multiple modes at the design stage”, since drawing has:-

1. “mechanistic tendencies limit creativity”
2. “greater attention to modernist diagram than traditional sketch design”
3. ‘need to interact with other modes of spatial exploration’
4. “ different frames of mind require different types of drawing at different points in creative process with a tendency to exclude non-designers”

All this poses significant questions as to how drawing might continue to be used for “sketching” or whether CAD and/or advances in computer graphics, tied to the other applications existing or
prospective in IT, might be a better basis for designers of complex systems such as advanced ships and buildings? Although it ought to be noted further that Edwards does admit “All designers .. interpret and solve problems by graphic mediation” and “when later changes are required, the freehand sketch retains its prime position.. in testing options.” Although both of these seem to be relevant to the other categories than the primary creative or “thinking sketch”. Before leaving Edwards comments it is worth noting his comment that

“Architects draw in a way unique .. although engineers and industrial designers also draw to solve problems their graphic techniques are quite distinct.”

as he then provides two engineering examples of sketches compared to an architect’s set of sketches for an arts complex proposal, Fig. 1. These sketches can not only be compared with each other but also with the ship “sketches” in Section 4.

---

**Fig. 1:** Sketches by Engineers (Calatrava and Ove Arup) contrasted with those of an architect (Terry Farrell), *Edwards* (2008)

*Brawne* (2003) in his chapter “Thinking and drawing” sees “design and drawing inseparable. …the translation of a thought into some visually discernable artifact.” “drawing (means) both making marks on a 2-D sheet or screen and making exploratory 3-D models. They are investigatory tools ... an essential element of the design process.” He also sees sketches fulfilling) three crucial and distinct functions:

- “as part of the thinking process of design,
- as an indication to the clients and users of what the building will be like, and
- as a set of specific instructions to those constructing the building.”

So “they are tentative answers to the hypothesis that has formed in the mind.” and “(sketches) are the nearest to a craft activity that occurs in the design sequence.” This seems to come back to achieving
the realisation of the initial creative idea, however he concludes, later in the book, that fundamentally "The sketch is communication"

An earlier book on “Visualisation and depiction of space in architecture” by Porter (1997) highlights Conceptual Diagrams as being the means of “The moment an idea is transferred from a designer’s mind to an external form is a critical point.” So the architect “adopt(s) some form of abstraction which represents or reflects the pictures in the mind. The process of abstraction usually involves the use of embryonic ideograms, descriptive symbols or annotation – images and words which combine to chart the potential relationship between concept and reality. A constructive doodle ... more concerned with the essence of ideas”. He then identifies five types of diagrams with example sketches of each. Not all of these seem to be sketches in the manner discussed by the three disciplines considered here, but it is not inconceivable that any computer enhancement of the traditional hand sketch could broaden the scope to cover much of Porter’s five categories of drawing.

In concluding this review of the nature of design sketching across different fields, it can be seen that the process of sketching focuses on different fundamental concepts and some domain specific vocabulary can confuse the concept of design sketching when applied to the different disciplines. For example the gross geometric shapes used to define overall form in architectural sketching, linkages and pivots in the development of mechanical systems or, when we get to discuss ship design, the profile and summary of principal particulars provided when sketching the design of a yacht.

A key point to note regarding these characteristics is that they are not linked to any specific solution or method of sketching. Instead, they define capabilities and features of sketching as an approach, which may be implemented in different ways and that some of these have already been seen to have distinct advantages in exploiting the power of the computer. It also needs to be considered whether, in the field of design more heavily engineering focused than architecture or product design that constitutes ship design, the future needs for encouraging innovation can also be enabled.

3. Recent Research in Design Sketching

Over the last decade or so, across the range of design disciplines, there has been research focused on sketching and how computer developments might be exploited. Thus McGown and Green (1998) noted that the design research community had at that time moved away from developing automated design systems and moved instead to developing design support software, which they specifically saw as including sketching tools.

3.1 Design theory

There have also been several investigations into the taxonomy of sketching in product design, in order to facilitate computer recognition and recording of sketching activities: Suwa et al. (1998) developed an extensive scheme for describing designer activities during the sketching process; Kavakli et al. (1998) investigated a more specific concept of breaking a design sketch for a product into distinct parts for digital storage, proposing that this type of hierarchy is an underlying part of human conceptual design processes; and Pachè et al. (2001) described an investigation into the design process utilising freehand sketches. This latter study examined the use of sketches by several student and professional designers when addressing a representative design problem. Different types of interaction with the sketches, such as re-enforcing lines and simply looking at the sketches, were identified. Of relevance to computer aided design, they concluded that future sketching CAD tools must permit the designer to proceed in a range of ways, rather than constraining them, and that the tools must be able to support geometric, symbolic and textual definitions of the design and that finally these definitions may also be at different levels of abstraction.

In addition to the traditional fields where design sketching has been extensively used, such as architecture and product design, recent research has examined its use in a range of professions. Thus Marchese (2006) described the use of highly symbolised sketches by organic chemists when seeking...
to understand the mechanisms involved in organic chemical reactions. Warr and O’Neill (2006) utilized sketching as a means of group work on urban planning, using a large format interactive display screen. Sketching has also been examined as a means to better understanding the conceptual design process itself with investigations of the differences between novice and expert designers, Kavakli and Gero (2001) and detailed examination of sketching processes leading to models of mental iteration, Jin and Chusilp (2006).

3.2 CAD applications

Despite Gross (2006) view that it is hard to sketch with a CAD system as it specifically lacks what he calls “fluidity”, there have been proposals to link the sketching processes and CAD. These are intended to have the ability to move from an initial sketch to a “more committed schematic” drawing. However Alvarado and Davis (2000) do describe what they call a sketching tool, with limited kinematic simulation capabilities for design of specific mechanical devices, which allows linkages, joints etc to be roughly defined and simple simulations of the resulting motion performed, but this very specific application would seem to lack Gross’s second “f” of being “forgiving”. Another tool that looks at the transition from the 2-D sketch is Borg et al’s. (2001) outline of a developmental system using feature recognition to generate 3-D CAD models (and subsequent CNC milled prototypes) from the 2-D sketches.

More consistent with the general view of sketching being free and exploratory in the “thinking” mode are earlier CAD tools by van Dijk (1995) and Tovey (1997), which allow the designer to rapidly develop crude 3-D CAD models based on their 2-D sketches. They appeared to do so by using 3-D modelling packages to display the 2-D sketches on-screen but overlaid in the 3D workspace. A related field where the use of sketching has been considered is in the development of new Human-Computer Interfaces (HCI). Fallman (2003) considers that sketching activities have taken place in HCI design, but these have erroneously been referred to as prototyping and that there has been a tendency in HCI research to focus on the attributes of the sketch / prototype itself, rather than what it represents, as an early sketch of the design. He went on to remark that the use of sketching in HCI development was complicated by the involvement of additional non-visual aspects, such as interactivity and sounds, not well represented in pen-and-paper sketches. A more recent example in this area is that of Gross and Do (2006), who describe recent work using rapidly produced prototypes, to overcome the limitations of traditional sketching. They see the processes of hacking and tinkering with software code and the production of partially-functional prototypes as being directly analogous to sketching in architecture and engineering design and akin to Ferguson’s “talking” sketch in engineering design, even if extended beyond the purely visual imagery associated generally with the term “sketching”.

3.3 Architectural theory

As early as 1997 Porter gives examples of non immersive and immersive use of virtual reality systems being used by architects as substitutes for the traditional hand sketching exploratory task when looking at the impact of a new building feature in a sensitive urban environment. Edwards (2008) draws attention to leading architects such as Zaha Hadid and Frank Gehry combing on a single sheet not just drawings but paintings, models and CAD representations on a single sheet to break out of “the straight jacket of orthogonal thought process of plan, section and elevation” thus achieving the “fluidity” of sketching, while drawing on the breadth of visual media becoming readily accessible. The ever radical architect Peter Cook (2008) concludes his recent book with the chapter “Beyond drawing- beyond reality” where he examples the use of photomontage and computer generated drawings to achieve the spontaneity, traditionally seen as the preserve of hand sketching.

4. Sketching in Preliminary Ship Design

Sketching has been used in used in the early stages of ship design, both as an aid to the development of the design itself and as a communications medium. Historically, the term “sketch design” has been
used in naval architecture to refer to an early stage description of the design, such as the formal outlines of naval designs submitted to the Board of the Admiralty for approval, *Brown (1983)*. These descriptions utilised a broadly specified format, including calculated estimates of weight, space, estimates of resistance and stability and an outline general arrangement. These aspects could be regarded as equivalent to the concepts of vocabulary and appropriate refinement listed in Table I. Figs. 2 and 3 provide examples of the general arrangements used in these sketch designs.

![Fig. 2: A sketch drawing of a design study for an armoured cruiser, 1907, *Brown (1999)*](image1)

![Fig. 3: “Spring Style” sketch of DDX1 study in 1978, *Keane et al. (2009)*](image2)

In addition to being used for communication of the design, these sketches would be (broadly) to scale and thus were used for hand calculations of armour weights, gun training arcs etc. In this, these sketch designs move beyond the “talking sketch”, in that they are not purely for communicating the current state of the design to others. In a more modern early stage ship design process, such as the UCL Postgraduate ship design exercise, this formal submission is retained in the form of design interviews and the Single Sheet Characteristics, which have a specified structure, *UCL (2001)*. It is notable that these sketches have a common grammar, both in the visual style and content, and in the nature of the supporting numerical data, that has changed little over time. Fig. 4 shows an example of the continued use of sketching in concept ship design. It shows three initial sketches for a modular OPV developed in a study carried out by the UCL DRC for the UK MoD Naval Design Partnering (NDP), *Pawling and Andrews (2010)*. The three sketches each focus on specific aspects of the configuration which acted as the “design generator”; the profile for the monohull and stern arrangements for a broad
transom monohull and trimaran. Fig. 4 also shows the completed designs, developed using the SURFCON tool, which have both similarities and differences from the initial sketch designs.

Fig. 4: Initial sketch designs and final configurations for UCL DRC modular OPV studies

The early stages of the design of yachts and pleasure craft also make extensive use of sketching, Woods (2006), Ivanov, (2006). Sketches are used both at the level of overall arrangement and external appearance and in the detailed layout of accommodation spaces. In these applications, sketching is primarily aimed at developing the layout and visual style and detailing of the vessel, with the ambiguity of sketching assisting the generation of new and creative ideas, Woods (2006). However, Woods also applies the sketching analogy to the hull-form coefficients and dimensions used to describe a hull-form design at the earliest stages of larger yacht design, showing that these numerical values demonstrate the ambiguity required of early sketches and are used in a similar process of creative discovery.

A key feature to note regarding the use of sketches and sketching in preliminary ship design is that it is not purely focussed on the visual and architectural aspects of the design but also includes numerical properties, such as armour thickness or powering calculations. These numerical values, however, are not the precise values that would result from more detailed analysis, but instead like the sketches contain inherent uncertainty and ambiguity.

5. Advances in Computer Graphics Relevant to Preliminary Ship Design

5.1 The Historical Perspective

There have been several reviews of the developments in preliminary ship design software tools that have specifically discussed user interfaces, Calkins (1988), Jensen et al. (1997), Tan and Bligh (1998). Thus the development of naval architectural software interfaces can be seen to be closely
linked to the development of computer tools in general and, while this general development path has not been applied universally, it is still convenient to summarise it as having occurred in three stages:

1. Text only interfaces, which presented the designer with information in a relatively abstract manner, such as the tanker preliminary sizing tool described by Nowacki et al. (1970);
2. Text interfaces with limited support for non-interactive graphical representation, such as the UK Ministry of Defence developed GODDESS tool described by Yuille (1978) and Holmes (1980);
3. Graphical user interfaces featuring interactive, integrated graphical representations of the ship design, such as PARAMARINE. This tool is described in more detail below, as it provides the environment for the research presented.

However some currently mainstream ship design tools, such as the US Navy tool ASSET, Heidenreich (2002), and concept requirement oriented systems, such as the UK Dstl Submarine Concept Aid, Biddell (1998), do not feature interactive graphical interfaces in their current configurations, preferring text-based dialogue boxes, with limited graphical output, within an overall windows interface.

The importance of the user interface in computer aided design tools has long been appreciated:

"Is it clear that what is needed, if the computer is to be of greater use in the creative process, is a more intimate and continuous interchange between man and machine. This interchange must be of such nature that all forms of thought that are congenial to man, whether verbal, symbolic, numerical, or even graphical, are also understood by the machine and are acted upon by the machine in ways that are appropriate to man's purpose," Mann and Coons (1965)

5.2 The Design Building Block Approach

Consistent with this aim the UCL Design Research Centre (DRC) has expounded and developed a configurationally-centred approach to preliminary ship design, which adopts a flexible configurational model of the ship combined with naval architectural numerical analysis tools to ensure technical balance, while enabling innovative exploration during the formative design evolution. This is designated the Design Building Block approach, Andrews and Dicks (1997). The DRC has instigated an alliance with Graphics Research Corporation Limited (GRC) to incorporate the Design Building Block approach through the SURFCON facility being incorporated within GRC’s PARAMARINE Preliminary Ship Design System, Andrews and Pawling (2003). PARAMARINE is an object-based naval architectural design package utilising the commercial ParaSolid modeller as its core, Munoz and Forest (2002). The screenshot, Fig. 5, shows the interactive graphical display of the design configuration (the “graphical pane” on the right, with a hierarchical navigation pane on the left and examples of numerical data and analysis (a resistance estimate in this case).

![Fig. 5: Screenshot of PARAMARINE showing interactive numerical, tabular and graphical information in the Design Building Block objects](image)
PARAMARINE-SURFCON is not just a graphical layout tool, it also contains objects for the assessment of the performance of the design across a range of ship design capabilities, including resistance and propulsion, stability, manoeuvring and radar cross section signatures, in order that each design study is both numerically balanced and achieves the desired levels of ship performance. The interactive graphical interface enhances the use of these numerical analysis tools by placing the results in the context of the current ship configuration – for example, the results of a stability curve (GZ) calculation can be visualised to directly investigate the effect of geometric shape on the GZ curve, a particularly important issue for certain multi-hulled vessels.

Thus the Design Building Block approach to the early stages of ship design seeks to encourage a more holistic approach to the development of the ship design solution. Instead of a set of numerical steps or a mechanistic approach, where each aspect of the performance of the design is examined separately and sequentially, with any limited graphics being an outcome of the numeric balance, the integrated nature of the SURFCON implementation in PARMARINE allows the physical aspects of the design to be continuously appreciated by the designer from the commencement of the design. In this sense it can be seen to be consistent with, at least, the philosophy of the “thinking sketch”. Whether it is sufficiently “fast” and “fluid” enough to wholly encourage creative and innovative sketching comparable with hand sketching is more debatable.

5.3 Applications of the Design Building Block approach

The UCL DRC has applied the DBB approach to a wide range of ship concept design studies and more detailed investigations into ship design methods. These have been summarised in several papers (Andrews and Pawling 2006), (Andrews and Pawling, 2009). Examples range from a series of seven mothership concepts developed for the UK MoD, each talking a week to develop (Andrews and Pawling 2004), to a trimaran Littoral Combat Ship (LCS) developed for the US Navy Office of Naval Research, which took three months to develop to a much higher level of detail (Andrews and Pawling 2008). Fig. 6 shows the development of a design for a Joint Support Ship (JSS) undertaken by the UCL Design Research Centre, as part of a bid team responding to a Canadian National Defence Department’s (DND) requirement for feasibility studies into a JSS programme, Andrews and Pawling (2007). This figure shows the four initial concepts at the Super Building Block design stage that were developed through two refined concepts to a single configuration over two weeks.

Fig. 6: Progression of the UCL JSS design from 4 initial designs to a single developed configuration

6. How Design Sketching Might be More Fully Realised in CAPSD

The concept of the “thinking” sketch, with the characteristics, outlined in Table 1, and the creative process of development and understanding it represents, has similarities to the earliest stages of ship design, particularly the Concept Exploration stage, Andrews (1993). It could be argued from the
summary of the properties of sketches and the process of sketching presented in the section on the nature of design sketching that the use of the Design Building Block approach in preliminary ship design is akin to the process of sketching. Since the DBB model, particularly at the earliest stages of design definition, Andrews and Pawling (2008), is used by the designer to explore options and suggest new ones this then is conceptually similar to all three types of sketches:-

- The thinking sketch: This is the sparsely populated, highly flexible model used in the Major Feature and Super Building Block design stages.
- The talking sketch: This is best represented by the descriptions of various DBB design studies summarised in Andrews and Pawling (2006), which show a range of potential concept design options intended to be discussed in design reviews.
- The storing sketch: This functionality was provided by a combination of the inherent ability of the PARAMARINE-SURFCON tool to store the design “as is”, without requiring a particular structure or level of detail, along with a textual and graphical design journal kept by the designer.

The SURFCON representation of a ship design, particularly in the early stages, can be seen to be equivalent to a sketch design, in that it has many of the properties of sketches listed in Table I. Although this could be true of other preliminary ship design tools, the Design Building Block approach, via its current PARAMARINE implementation, seems an appropriate method of ship design sketching for several reasons, Table I:

- Fluid: The DRC concept studies have shown that the DBB approach can be used to develop a series of “sketch designs” quickly and for use in group design reviews. The configurational model and graphical interface are seen to enhance the designers understanding of the design and ability to explore emerging features of the design.
- Forgiving: Although the analysis tools within PARAMARINE each require a certain level of definition to return results, the overall level of refinement in the design is not explicitly constrained, so approximations and estimates can be used to rapidly develop designs. Similarly, the design tool does not add more detail beyond that defined by the designer.
- Functional: As shown in Fig. 5, the PARAMARINE tool uses common representations of design data, such as resistance and GZ curves. Fig. 5 also shows recognizable features such as decks and bulkheads that allow evaluation of the design.

However, PARAMARINE and other software tools are still limited when considered from the perspective of a sketching approach, particularly in the modelling effort required of the designer. In some cases the designer must define conceptually simple design features (e.g. a mast, a bulkhead) in an abstract manner. This increases the time invested in creating the model, thus reducing its “disposability”, in that the designer will be more likely to adopt a “minimum change” approach to the design, and is clearly less “fast” and “fluid” (to use Gross’ terms) than the traditional hand sketch. This could be addressed by enhancing the use of ship design software, such as SURFCON, in the early, sketching – like stages of ship design. This would thus reduce the effort the designer needs to expend on modelling for analysis purposes and permit him to focus on modelling for exploration. The latter is a key sketching attribute, where creativity and innovation can be more fully employed and understanding of the problem greatly enhanced.

7. Future Possibilities

7.1 Requirements for a Sketching Based Approach

It has been argued that the DBB approach to preliminary ship design is a significant step towards a complete design sketch philosophy in the earliest stages of ship design, in that it encourages a sketching-like approach of exploration, innovation and understanding in design. However, the full exploitation of highly flexible sketch representations to explore and more fully understand the design space is limited in preliminary ship design. In part, this is due to the significant technical complexity inherent in ship design in the need to not just balance weight and buoyancy but also propel the vessel-a demand not required of most other physically large and complex systems, Andrews (2011). Thus to
enable a process more akin to design sketching for preliminary ship design a flexible, interactive, visually rich CAD system with a flexible configurational model and, vitally, integrated technical analysis is seen to be required.

7.2 Sketching as an Interface with Automation

Historically, sketching in preliminary ship design has been designer – centred using manual methods. The current implementations of the Design Building Block approach similarly rely on the designer to define all configurational details of the ship design. Although numerical parametric analysis and “optimization” approaches have long been used in ship design, Vasudevan (2008) more recent work has examined the possibility of developing configurations using automatic or semi-automatic methods, McDonald (2010, van Oers et al. (2010, Nick (2008). These numerical approaches have the potential to generate large amounts of data which must be included in the preliminary design process to have most benefit on the resulting design. One possibility could be the use of both a sketching approach – one of designer led exploration and investigation – and a sketching interface – a flexible, graphical toolset for investigating numerically generated configurations or options. Some work has been undertaken in this area, such as Van Oers et al. (2008) and tools exist for the exploration of large and complex data sets, such as those developed by Fry (2000). Enhancements in this area may be best achieved by adapting approaches that have originated outside traditional naval architecture and engineering fields.

7.3 User Interfaces for Sketching in Preliminary Ship Design

As noted above, although the flexible, configurationally centred SURFCON implementation of the DBB approach is seen to assist in utilising a sketching approach in preliminary ship design, there are also features that could be enhanced. Future developments could reduce the “gulf of execution”, Pawling (2007), by reducing the degree of abstraction and duplication needed to represent ship features. A key indicator of this need is that, despite advances in user interfaces, a pen-and-paper sketch was still the starting point for a design study conducted in 2010, Fig. 7.

Fig. 7: Concepts for a future preliminary ship design tool
Such interface development for preliminary ship design would address more than new objects to model spatial features, it would also address the implementation of a naval architecturally relevant visual vocabulary, within the graphical user interface of the software (for example, using traditional symbols for amidships, greater representation of ship systems and ship features such as decks, bulkheads and superstructure blocks). Those needs could also be improved by utilising existing and developmental interface technologies, such as graphics tablets, 3-D pens, high resolution screens, 3D and VR.

User interfaces are another aspect where engineers and naval architects may benefit from looking outside their own fields. For example computer game developers are focused on the user experience and may have much to offer. Fig. 7 shows some concepts for future interfaces, including; enhanced modelling tools with a focus on configurational features such as bulkheads and decks; seamless movement between 2D and 3D modelling; configuration driven access to design libraries; and visualization based exploration of design options. There is considered to be a clear agenda for significant developments in spatially oriented tools for preliminary ship design, provided the statements in this position paper can be taken up by the ship design profession and the vendors of their tools.

References

ALVARADO, C.; DAVIS, R. (2000), Intelligent mechanical engineering design environment: From sketching to simulation, MIT AI Laboratory Annual Abstracts, MIT

ANDREWS, D.J. (1993), The management of warship design, Trans. RINA 135


ANDREWS, D. (2011), Is the design of physically large and complex systems a science or an art?, paper submitted.

ANDREWS, D.J.; DICKS, C. (1997), The building block design methodology applied to advanced naval ship design, IMDC, Newcastle

ANDREWS, D.J.; PAWLING, R.G. (2003), SURFCON - A 21st century ship design tool, IMDC, Athens


ANDREWS, D.J.; PAWLING, R. (2006), The application of computer aided graphics to preliminary ship design, IMDC, Ann Arbor


ANDREWS, D.J.; PAWLING, R.G. (2008), A case study in preliminary ship design, IJME 150/A3

ANDREWS, D.J.; PAWLING, R. (2009), The impact of simulation on preliminary ship design, IMDC, Trondheim


BIDDELL, M.P. (1998), Whole submarine concepts and research integration, Undersea Defence


BUXTON, B. (2006), What sketches (and prototypes) are and are not, CHI 2006 Sketching Workshop, Montreal


DARKE, J. (1979), The primary generator and the design process, Design Studies 1/1, pp.36-44.


FALLMAN, D. (2003), Design-oriented human-computer interaction, CHI 5/1, Fort Lauderdale


GALLIN, C. (1973), Which way computer aided ship design?, ICCAS, Tokyo

GOLDSCHMIDT, G. (1991), The dialectics of sketching, Creativity Research J.,4/2, pp.123-143


GROSS, M D. (2006), Representations: Sketching beyond pencil and paper, CHI 2006 Sketching Workshop, Montreal

GROSS, M.D.; DO, E.Y.L. (2006), Sketching human computer interactions, CHI Sketching Workshop, Montreal

HEIDENREICH, T. (2002), ASSET: Advanced Surface ship evaluation tool ship design synthesis programs, Lecture Presentation at MIT, NSWCCD Carderock

HOLMES, S.J. (1981), The application and development of computer systems for warship design, Trans. RINA 123


MUNOZ, J.; FORREST, C. (2002), Advantages of software integration from initial design through to production design, ICCAS, Malmo


NOWACKI, H (2009), Development in marine design methodology: Roots, results and future trends, IMDC, Trondheim


PORTER, T. (1997), The architect's eye: Visualisation and depiction of space in architecture, E&FN Spon


TAN, K.T.; BLIGH, T.P. (1998), A new approach to an integrated CAD method for surface ship design, Naval Engineers J. 110/1, pp.35-48

TOVEY, M. (1997), Styling and design intuition and analysis in industrial design, Design Studies 18/1, pp.5-31.

VAN DER LUGT, R, (2005), How sketching can affect the idea generation process in design group meetings, Design Studies 26/2, pp.101-122.


VAN OERS, B.; STAPERSMA, D.; HOPMAN, H. (2010), A 3D packing approach for the early stage configuration design of ships, COMPIT, Gubbio


YUILLE, I.M. (1978), The forward design system for computer aided ship design using a minicomputer, Trans. RINA 120
Development of Decomposition-Based Design Optimization Tools for Ship Design

Hamid Hefazi, California State University, Long Beach/USA, hefazi@csulb.edu
Josh Hamel, California State University, Long Beach/USA, jhamel@csulb.edu
Adeline Schmitz, California State University, Long Beach/USA, schmitz@csulb.edu

Abstract

The design of complex engineering systems, such as ship systems, involves making design decision with respect to numerous highly coupled subsystems. Traditionally this level of complexity has led to designers using sequential and spiral design processes to ensure agreement between all included subsystem designs. This process can be influenced by designer subjectivities and/or can lead to suboptimal system designs. This paper describes the development and application of an alternative decomposition based design optimization method for ship design. This approach decomposes a traditional sequential synthesis design model (SDM) into set of coupled subsystem models which are then optimized using one of several coordination optimization methods available in the literature. Building upon previous works of the authors on synthesis level multi-disciplinary design and optimization (MDO) method for multi-hull ships, this paper explores the applicability of various decomposition based design optimization approaches and selects the collaborative optimization (CO) as the best approach. An existing SDM for trimarans is then modified and re-organized in order to make it appropriate for CO methodology. The CO methodology is formulated and applied to a notional trimaran vessel design that satisfies the Joint High Speed Vessel (JHSV) type mission requirement.

1. Introduction

The ship design process and associated tool development is an important current focus of the U.S. naval engineering community. Modern ship-design efforts could be substantially enhanced through the consideration of advancements in a number of fields including Operations Research, System Analysis, the Theory of Games and Meta Engineering. The efficient application of a “system” approach and appropriate methods in “system” analysis of the ship design process are critical and remain a fertile area for innovation. In this paper we are exploring the application of decomposition based design optimization to the ship design process.

Optimization in traditional engineering design is typically based on a sequential approach involving several phases: definition of the design optimization task, mathematical modeling of the task with selection of the input variables and constraints on the output parameters, development of the algorithm and/or task solver, and analysis of the solution. Traditionally, the analysis in each subsystem is
performed separately because connections between subsystems are too complex and interrelated to solve simultaneously. Therefore, the overall problem takes a sequential approach, where each discipline is optimized sequentially. The results of the previous discipline’s optimization become the basis for successive discipline optimization problems. As a consequence, very limited possibilities are left to the last optimization problem; the first discipline optimized dominates the final solution. This traditional “design spiral” is illustrated in Fig. 1.

In this approach, “convergence” is the simple balancing of the design in terms of major design constraints. It occurs when, for example, the estimated weight equals the displacement of the ship at the design draft, or when the area available equals the area required. However, this understanding of convergence in the design spiral process is only partially correct. In the ship design engineering process the term “convergence” actually means achieving the design goals with optimal values of measure of merits provided that any design solutions are feasible (from the point of design requirements and constraints). The convergence of the “design spiral” process cannot be mathematically proven, but is achieved in practice by artificial, intuitive interaction with the professional designer (naval architect) through his/her individual experience and historical data. Termination of iterations is based on personal experience of the designer who makes an arbitrary selection for the mentioned sub-succession, but not on the convergence upon formalized criterion. Furthermore, using this sequential method there is no means for assuring that any of the design solutions are globally optimal since it is highly likely that the design solution arrived at are dominated by designer preference, experiential knowledge and/or the results of previous iterations of the design spiral. As a result, strategies for ship design that move beyond the design spiral are needed and highly desirable.

Design optimization is a well-studied and widely used design tool that has applications in many engineering fields. However, when designing large scale, complex or multi-disciplinary system, such as in ship design problems, the application of many optimization schemes becomes extremely difficult. To meet this challenge the aerospace engineering community developed the concept of multi-disciplinary design optimization (MDO) in the 1980s and designers of complex systems wishing to employ design optimization techniques look to the MDO community for relevant solution methodologies. All MDO approaches function by providing designers with a means for managing the complexity associated with designing engineering systems that are comprised of numerous coupled subsystems. There are two main classes of MDO approaches. The first class of MDO approaches include those that seek to solve the optimization problem at the system level and include all subsystems as part of the overall system in a so-called “all-at-once” (AAO) fashion. The second class of approaches alternatively treats the overall design problem in a decomposed fashion where each subsystem is designed independently but within the context of the larger design problem. Both classes of problems have their strengths and weaknesses, but the decomposed approaches provides the most flexibility and offers an interesting opportunity for further improving the ship design process and moving it beyond the traditional ship design spiral.

In a decomposition-based design optimization strategy each subsystem design problem produces local solutions with respect to local decision variables, the number of which is less that the total number variables with respect to the entire system as a whole, thus resulting in a set of subsystem design problems which are easier to solve using optimization techniques. Furthermore, because each subsystem is decoupled from the overall system problem within decomposition-based design optimization approach, these strategies are well-suited to computationally expensive analysis tools such as Computational Fluid Dynamics (CFD) and Finite Element Analysis (FEA). Decomposition-based design optimization strategies then use the data generated by each subsystem design problem to coordinate the design optimization of overall ship system in a meaningful way. This decomposed strategy reflects the logical sequence and performance of the design process and offers the potential for a higher quality (and lower risk) ship designs in a shorter period of time. Within the context of ship design, subsystem design problems traditionally include design tasks such as hull form development, structural design, machinery plant development, propulsion selection and the design of other major ship systems.
Previous works by the authors on synthesis level MDO using an AAO approach have developed and demonstrated several innovative techniques for improving the design and evaluation of both commercial and military ships at the synthesis level, Hefazi et al. (2005,2008,2011). This paper extends the previous works to a decomposition-based design optimization strategy.

2. Decomposition Based Design Optimization

The complexity associated with a ship design problem is clearly present in the general ship synthesis design model (SDM) process flow chart depicted in Fig. 2. In Fig. 2 (and all subsequent figures) blue arrows will denote input parameters, green arrows will denote shared or coupling parameters and red arrows will denote output values. An SDM model of this type has been previously developed by the Computer Science Corporation (CSC) as part of an ongoing joint research program, but is currently configured as an open loop non-iterative design process. It works well with process integration and design optimization (PIDO) middle wares such as iSIGHT and Model Center since it is built upon standard database technology, but it is not well suited to decomposition or decomposition based design optimization strategies.

An alternative to the AAO approach is to instead decompose the overall system model, Fig. 2, into a set of independent subsystem models, which can then be optimized independently, Fig. 3, leading to an optimal system comprised of optimal subsystems. These subsystems are obviously coupled, so an additional system-level model is needed to manage the couplings between subsystems and to drive the system level design problem. In this approach, called decomposition-based design optimization, the values for all shared parameters and/or subsystem level couplings are determined at the system level, while independent subsystem parameters are determined by the lower level subsystems with a measure of design autonomy.
There are various strategies for decomposition-based optimization reported in the literature, including concurrent subspace optimization (CSSO), Sobieszczanski-Sobieski, (1989), Huang (2007), collaborative optimization (CO), Braun et al. (1997), Li and Azarm (2008), and analytic target cascading (ATC), Kim et al. (2003), all of which manage the system level decision-making process slightly differently with varying strengths and weaknesses. Selecting the appropriate decomposition based optimization approach is obviously problem dependant, but all of these approaches can use the same general organization as shown in Fig. 3. Each of these approaches will be outlined in the next several paragraphs, but for specific details please see the relevant references listed above.

The CSSO strategy is a well-known and widely used MDO approach credited originally to Sobieszczanski-Sobieski (1989). This approach involves decomposing a complex design problem into a set of subsystem design problems coordinated by a system-level coordination optimization problem (COP). The COP is connected to an optimizer that determines the values for all parameters shared by two or more subsystem in an effort to produce an optimal performing system based on the performance of each included subsystem. At the subsystem level each subsystem is optimized with respect to local design problems while all variables determined by the COP are held fixed. In this strategy each included subsystem optimization problem is solved for each candidate solution at the system level, resulting in a nested bi-level optimization problem. The CSSO strategy is well-suited to optimizing the design of multi-disciplinary systems that have shared parameters between subsystems but has no provisions for subsystems that are coupled (e.g. the output from one subsystem is an input to another subsystem. Fig. 4 depicts the basic CSSO formulation graphically, with Fig. 4(a) showing a general two subsystem multi-disciplinary model and Fig. 4(b) depicting the system decomposed via the CSSO strategy. Note that in the Fig. 1 there are no green arrows, meaning this strategy is not suited to coupling parameters between subsystems.

The CO formulation, originally developed by Braun et al. (1997) is shown in general terms in Fig. 5, with Fig. 5(a) again showing a general coupled system comprised of two subsystems, while Fig. 5(b) shows the same system decomposed via a CO formulation. When using CO, the system level optimizer determines the values for any design parameters that are shared by two or more subsystems (denoted by blue arrows) in addition to determining the values for any parameters that couple any two subsystems (green arrow), thus accounting for the shortcoming of CSSO that was highlighted in the
previous paragraph. As with CSSO the shared parameters are determined by the system level problem and then can be passed to the subsystems as fixed parameters. However, the coupling parameters are determined at the system level in the form of design targets (orange arrows) and then the subsystem optimizers force the local coupling values to converge to the system level determined targets, thus ensuring consistency between subsystems. The CO strategy is more flexible than the CSSO approach in that it can handle coupling parameter; however this increased flexibility comes at the expense of an increase in dimensionality in the form of the design targets which must be determined by the system level optimizer. The CO approach is also a bi-level nested optimization approach meaning that again each subsystem optimization problem must be solved for every candidate solution at the system level.

![Fig. 5: CO Formulation](image)

The ATC approach was developed by Kim et al. (2003) as means for completing MDO in a top down fashion. Both the CSSO and CO approaches (along with many other related MDO approaches) function by determining the values for the design variables for both the system and subsystem level problems in an effort to achieve optimal system performance and optimal coordination between subsystems within the final design solution. The ATC approach conversely is a top-down strategy that seeks to determine system and subsystem level design variables such that the resulting final design will vary as little as possible from designer specified design performance targets. In this approach the problem is decomposed into top-level performance based models and lower-level subsystem models. The latter define the subsystem designs that drive the top-level performance. The designer then specifies targets for system level performance and the top level optimization problem determines optimal target values for all included subsystem, which may be input parameter values, coupling parameter values or subsystem level performance matrices. These targets are then passed to the subsystem problems which are then optimized with respect to any local parameter in an effort to produce as little deviation as possible from the passed-down target values. Since the top level optimization problem is completed prior to passing down any target values to the subsystem level, the strategy does not feature a bi-level optimization problem, but there is the possibility that when the subsystem level problem(s) are solved given the top level target a feasible solution may not exist. For this reason iterations with respect to the specified top level target may be necessary to converge to a final design solution. In other words, it may be necessary to incrementally adjust the top level targets and successively resolve the entire design problem until all subsystem designs are feasible. This strategy has the unique trait of employing a top down approach and thus being target performance based, but developing a decomposed system model that works well with this strategy may require significant effort since many design models are based on the more common bottom up (or input-output) formulation. Fig. 6 depicts the ATC decomposition formulation graphically for comparison purposes, again showing general coupled system in Fig. 6(a) and the system decomposed using the ATC approach in Fig. 6(b).

![Fig. 6: ATC Formulation](image)
For this research effort the collaborative optimization (CO) strategy will be used to optimize the CSC developed SDM in a decomposed fashion. This approach was chosen due to the high level of coupling that is present in the CSC SDM, and due to the bottom up design approach employed by the SDM. The specific details pertaining to the formulation of the CO algorithm and how it will be applied to the SDM problem are presented in Section 4. It should be noted that there are various different formulations for each of these strategies (CSSO, CO and ATC) reported in the literature and that the preceding paragraphs simply outlined the basic concepts of these widely used approaches. Also, many decomposition-based design optimization strategies have convergence issues, especially when gradient based optimization solvers are used. These convergence issues can largely be avoided through the use of meta-heuristic optimization solvers (such as genetic algorithms) and as a result for this research genetic algorithms (GA) and multi-objective genetic algorithms (MOGA), Deb (2001), will be used to solve all optimization problems.

3. Simplified Decomposed SDM

This section describes the synthesis design model (SDM) for a notional trimaran vessel design that satisfies Joint High Speed Vessel (JHSV) type mission requirements as well as its decomposition, in order to apply the CO approach to optimize the design of the trimaran. The JHSV basic requirements are set out in the publicly available NAVSEA solicitation N0002407R2219. First a brief explanation of the existing SDM as it was used in previous research efforts, Hefazi et al. (2005, 2008, 2011) is presented. Next the main design variables, coupling variables and output parameters used for the model in a decomposed fashion, suitable for CO decomposition-based design optimization, are described.

3.1. SDM model

The CSC trimaran SDM is a set of calculations and data tables in a spreadsheet environment that supports the synthesis design of a trimaran through the selection of a set of ship design parameters, such as hull geometry values and desired operating conditions. The SDM provides the designer with feasibility checks and measures of merits based on the selected input parameters in order to achieve a balanced final design for the trimaran with respect to a) ship weight vs. buoyancy, b) required vs. available areas/volumes and c) fluid resistance vs. available propulsion power. The SDM also supports the design of auxiliary machinery and electrical loads within the final ship design. The flow of information through the SDM is shown schematically in Fig. 7 and is documented in detail in Hefazi et al. (2011).

---

**Fig. 7: Synthesis design model (SDM)**
The SDM supports the selection of numerous ship parameters, but for the purpose of this research as set of relevant design variables were selected and are provided in Table 1. As previously mentioned, the SDM spreadsheet calculates several different measures of merit for the selected design variables including the difference between the calculated displacement and Navy standard weight breakdown (SWBS), the wetted surface resistance at boost speed, the dead weight to displacement ratio, the maximum speed boost, and the overall beam. The objective functions for the SDM optimization selected for this research are the dead weight to displacement ratio and the wetted surface resistance at boost speed, which are both to be maximized.

A flow chart of the most important intermediate variables in SDM is shown in Fig. 8: Detailed description of SDM model parameters and subsystems Fig. 8. The design starts by selecting values for all chosen design variables. Note that any other parameters of the ship design are held fixed in the SDM (e.g. transit speed, ship manning, etc.). The SDM calculation can be divided into four main modules, or subsystems, including: Hull, PWR Req, Engines and Weight. In the Hull worksheet of the SDM ship volumes (Vch, Vsh and Volume), displacements (Dch, Dsh and Disp), slenderness Ratios (SRch, SRsh and SR), wetted surfaces (WSch,WSsh and WS) for the center, side hulls and the overall ship are calculated. Also the overall beam (Beam) and a modified slenderness ratio for transit (SR-H2) are calculated. These values calculated in the Hull worksheet are then fed into the required power worksheets (PWR Req) which calculate the boost and transit power required for boost conditions (PWRreqB1 and PWRReqT1), the boost and transit power required for transit conditions (PWRreqB2 and PWRReqT2), and the required boost and transit resistance (RB, RT). These required power values are then fed to the engine selection worksheet which chooses the appropriate boost and transit engines from a list of predetermined engine choices. Once these engines are chosen, the available power for boost and transit conditions (PWRaB and PWRaT) are calculated along with the Propulsion Weight (PropWeight). PWRaB and PWRaT can then be used to calculate the actual boost and transit speeds. These values along with Beam, Volume and Disp from the Hull worksheet are fed into the weight calculation worksheets, which calculate the lightship weight and the ship’s deadweight (Lightship and Deadweight). The lightship weight is the sum of the structure, propulsion, electric plant, command and control, auxiliary machinery, outfit, and weapons weights. The deadweight is the Navy defined ship work Breakdown structure (SWBS) deadweight. The sum of the lightship weight and the deadweight must fall within 300 m tons of the displacement calculated in the Hull worksheet. Finally, once all associated subsystem calculations have been performed, the design objectives can be calculated to evaluate the performance of the candidate design.

<table>
<thead>
<tr>
<th>Variable</th>
<th>Lower Bound</th>
<th>Upper Bound</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Bch</td>
<td>7.5</td>
<td>12.0</td>
<td>Center Hull Beam on Waterline</td>
</tr>
<tr>
<td>Bsh</td>
<td>3.0</td>
<td>6.0</td>
<td>Side Hull Beam on Waterline</td>
</tr>
<tr>
<td>Cch</td>
<td>0.5</td>
<td>0.625</td>
<td>Center Hull Block Coefficient</td>
</tr>
<tr>
<td>Csh</td>
<td>0.35</td>
<td>0.55</td>
<td>Side Hull Block Coefficient</td>
</tr>
<tr>
<td>Ccm</td>
<td>0.675</td>
<td>0.8</td>
<td>Center Hull Maximum Section Coefficient</td>
</tr>
<tr>
<td>Csm</td>
<td>0.7</td>
<td>0.8</td>
<td>Side Hull Maximum Section Coefficient</td>
</tr>
<tr>
<td>dch</td>
<td>9.0</td>
<td>12.0</td>
<td>Center Hull Depth</td>
</tr>
<tr>
<td>Lch</td>
<td>100.0</td>
<td>150.0</td>
<td>Center Hull Length on Waterline</td>
</tr>
<tr>
<td>Lsh</td>
<td>40.0</td>
<td>65.0</td>
<td>Side Hull Length on Waterline</td>
</tr>
<tr>
<td>Tch</td>
<td>3.5</td>
<td>6.0</td>
<td>Center Hull Draft</td>
</tr>
<tr>
<td>Tsh</td>
<td>7.5</td>
<td>10.0</td>
<td>Side Hull Draft</td>
</tr>
<tr>
<td>alpha</td>
<td>0.5</td>
<td>1.0</td>
<td>Separation</td>
</tr>
<tr>
<td>beta</td>
<td>0.15</td>
<td>0.35</td>
<td>Stagger</td>
</tr>
</tbody>
</table>
In order to use a decomposition-based design optimization strategy, the above described SDM must be modified. The same four calculation modules described in the previous section have been chosen to represent the SDM in a decomposed fashion. However, since the SDM model within a spreadsheet environment does not lend itself to decomposition, a necessary first step is to develop decomposed subsystem models for each of the relevant groups of calculations that were performed sequentially in the original SDM spreadsheet.

To accomplish this, each of the four relevant groups of calculation within the SDM spreadsheet was replicated as individual functions within the Matlab programming environment (i.e., Hull, PWR req, Engines and Weight). As a result, for this research, the fully coupled SDM spreadsheet is replaced by four distinct and decomposed Matlab functions which can then serve as the subsystem design models within a decomposition-based design optimization strategy. It is important to note that the use of Matlab to define all subsystem models for the CO formulation (or any MDO formulation for that matter) is by no means necessary. Subsystem analysis model can be developed using any computational tools that are available, relevant or useful to a specific set of designers or engineers involved a greater ship design effort, including C++, FORTRAN, shell scripts or Excel, just to name a few. Furthermore, subsystem analysis model can even be CFA or FEA models, but in those cases it may be necessary to approximate those computationally expensive codes through the use of less expensive surrogates such as neural networks, Schmitz and Hefazi (2007), and kriging models, Martin and Simpson (2005), for example. The only requirement is that the computational environment managing the coordination of these various models be able to provide all included subsystem models with inputs and then read the associated outputs from those models; which is a task easily accomplished using a middleware such as iSight.

During this process it was necessary to determine which outputs from one subsystem serve as inputs to other subsystem. These coupling are critical within the decomposition-based design optimization framework. As outlined in Section 2, all decomposition-based design optimization strategies (including CO) require the use of a system level model to coordinate the design of the decoupled subsystems. This new model, called the “SDM Coordinator,” organizes and controls the information used by the subsystems and must be developed in addition to the four subsystems design models (Hull, PWR Req, and Weight, and Engines). The resulting ensemble of models which are needed to define the SDM in a decomposed fashion are shown in Fig. 9. Note that in Fig. 9 it is necessary to define the parameters that couple the four subsystem models in Fig. 8 at the system level (or by the SDM Coordinator).
There are two different types of variables at the system level. The first are those that are shared by two or more subsystem, which will be denoted as $x_{sh}$ (with the “sh” short for shared) from this point forward. An example of shared parameters are the waterline length of the center (ich) and side (lish) hulls, which are input parameters to more than one of the subsystem models. The second set of parameters are those that couple two or more subsystem and thus must be determined through the use of target values set by the SDM Coordinator (see Section 2), denoted as $t$ (short for target). In the SDM model, some of the outputs of the subsystems are inputs to other subsystems. An example of this type of coupling can be seen in the center hull wetted surface ($WSch$) parameter, which is calculated in the hull subsystem but then serves as an input to the PWR Req worksheet.

The SDM coordinator in this decomposed formulation is tasked with minimizing the SDM objective functions by varying the shared ($x_{sh}$) and coupling ($t$) variables subject to system level constraints. Each subsystem model is then provided with each set of candidate shared and coupling parameter values to be evaluated within the system level optimization. The subsystem model then seeks to design that subsystem, subject to any subsystem level constraints, such that its design deviates the least from the values provided by the SDM. The specifics of this process and it formulation are described in the following section.

Fig. 9: Decomposed SDM

4. **SDM Collaborative Optimization Formulation**

The collaborative optimization approach described in Section 2 is formulated through the use of two or more coupled optimization problems in a bi-level arrangement. Eq.(1) provides the formulation of the system level optimization problem, which determines the values for the shared ($x_{sh}$) and coupling ($t$) parameters in order to optimize the performance of the resulting designs with respect to the system level objectives ($I$) subject to the system level constraints ($J$) and an additional constraint for each
included subsystem that forces each subsystem design to deviate from the system level target within a specified tolerance ($\varepsilon$).

\[
\min_{x_{sh}, t} f_{sys,i}(x_{sh}, t) \quad i = 1, \ldots, I \quad \text{subject to} \quad g_j \leq 0 \quad j = 1, \ldots, J \\
\quad d_k \leq \varepsilon \quad k = 1, \ldots, K
\]  

Eq.(2) provides the formulation for each of the $K$ included subsystem optimization problems. At the subsystem level there is a single objective for each subsystem, which seeks to minimize the deviation of the subsystem level design with respect to the targets provided by the system level problem, included both the coupling parameters that are inputs to the subsystem ($x_c$) and those that are output of that subsystem ($y_c$). At the subsystem level the shared parameters ($x_{sh}$) provided by the system level problem are held fixed. The decision variables at the subsystem level are all local variables ($x_{local}$) which are not used in any other subsystem, and any of the coupling parameters that are inputs to that specific subsystem ($x_c$).

\[
\min_{x_{local}, y_{local}} d_k = \frac{1}{2} \left( \left\| x_{c,k} - t_{x,k} \right\|_2 + \left\| y_{c,k} - t_{f,k} \right\|_2 \right) \quad \text{subject to} \quad g_j \leq 0 \quad j = 1, \ldots, J
\]

The function of Eq.(2) is to provide feedback on the selected target values. Eq.(2) allows each subsystem to select input values ($x_{local}$ and $x_c$) in order to produce a set of input and corresponding outputs that deviate from the supplied target as little as possible (minimum $d_k$). As a result, if the target values chosen by Eq.(1) are sub-optimal (or infeasible) the $d_k$ values for one or more of the included subsystem will be too high, indicating the chosen targets ($t$) will lead to a poorly coordinated ensemble of subsystem designs. Eq.(2) is nested within Eq.(1) as $d_k$ is the objective of Eq.(2) but also appears as a constraint (or set of constraints) in Eq.(1). This means that Eq.(2) must be solved for each of the $K$ subsystem for each set of decision variable evaluated within the optimization problem solved in Eq.(1). For the SDM design problem described above, there are four included subsystems, so $K = 4$.

The shared parameters ($x_{sh}$) and the target parameters ($t$) are shown in Fig. 9. The coupling parameters are depicted in Fig. 9 and are further detailed in Table II. Not shown are the local design variables, as for the SDM model only the Hull subsystem has local design variables and those can be found in the Hull subsystem block in Fig. 9. To solve for an optimal design of the SDM using the CO formulation, Eqs (1) and (2) must be solved (using MOGA for Eq.(1) and GA for Eq.(2)) within a computational environment where the decoupled subsystem models and SDM Coordinator model described above are used to evaluate the values for all $f_i$, $g_j$ and $d_k$ values necessary to solve Eqs.(1) and (2). The individual optimization problems are assumed to be converged in accordance with the usual guidelines for employing genetic algorithms (e.g. a predetermined number of generations).

<table>
<thead>
<tr>
<th>Subsystem</th>
<th>$x_c$</th>
<th>$y_c$</th>
</tr>
</thead>
<tbody>
<tr>
<td>Hull</td>
<td>WSsh, WSch, WS, SR, SR_H2, Beam, Volume, Disp</td>
<td>PWRreqB1, PWRreqT1, PWRreqB2, PWRreqT2, RT, RB</td>
</tr>
<tr>
<td>PWR req</td>
<td>WSsh, WSch, WS, SR, SR_H2</td>
<td>PWRreqB1, PWRreqT1, PWRreqB2, PWRreqT2, RT, RB</td>
</tr>
<tr>
<td>Engine</td>
<td>PWRreqB1, PWRreqT1, PWRreqB2, PWRreqT2, RT, RB</td>
<td>PropWeight</td>
</tr>
<tr>
<td>Weight</td>
<td>Beam, Volume, Disp, PropWeight</td>
<td></td>
</tr>
</tbody>
</table>

5. Results

As of the time of publication, results from applying the decomposition-based design optimization strategy described in this paper to the SDM design problem are not yet available in final form. To date all subsystem models have been reproduced in Matlab using the original Excel based SDM as a guide.
Those models have been validated and they accurately reflect the operation of the Excel SDM. Furthermore, the CO algorithm has been developed within the Matlab environment and work is underway to produce design solutions for the SDM using the newly developed decomposed subsystem models described in Section 3 and the Matlab based CO algorithm described in Section 4. Unfortunately the process of moving all SDM subsystems from the Excel model to new Matlab based submodels has proved to be a lengthier process than expected and delayed the production of quality optimal design results in time for publication. Work is ongoing and results will be available in time for the COMPIT conference in May 2011 and will be presented at the conference.

Acknowledgment

This paper is based upon work supported by the United States Office of Naval Research, under Cooperative Agreement No. N00014-04-2-0003 with California State University, Long Beach Foundation, and the Center for Commercial Deployment of Transportation Technology (CCDoTT). We would like to thank the program manager Dr. Paul Rispin from ONR for this support. We would also like to thank Mr. Stan Wheatley and Mr. Steven Hinds from CCDoTT, and CCDoTT program coordinator Ms. Carrie Scoville for all their supports. Also the Excel based SDM was developed by CSC as part of a previous joint CCDoTT program. The authors thank Dr. Igor Mizine (CSC) for this contribution.

References


SCHMITZ A.; HEFAZI, H. (2007), Constructive neural network ensemble for regression tasks in high dimensional space, 6th Int. Conf. on Machine Learning and Applications (ICMLA), Cincinnati

The Simulation Toolkit Shipbuilding (STS) –
10 Years of Cooperative Development and Interbranch Applications

Dirk Steinhauer, Flensburger Schiffbau-Gesellschaft mbH & Co. KG,
Flensburg/Germany, steinhauer@fsg-ship.de

Abstract

At Flensburger Shipyard simulation has been established as the main tool to support the decisions in production facility planning as well as in production planning and control. Because the available simulation tools are not sufficient for the usage in shipbuilding Flensburger Shipyard started the development of the Simulation Toolkit Shipbuilding (STS) in the year 2000. The STS contains a large variety of simulation tools for material flow modelling, model management, execution strategies and output analysis not strongly related to shipbuilding any more. It is further developed and used within the international cooperation SimCoMar and in the interbranch cooperation SIMoFIT.

1. Introduction

Especially in shipbuilding as a one-off-a-kind production the minimization of resources and the reliable adherence to a tight schedule is challenging. The dynamic dependencies between the product with its internal diversity of elements, the production processes and the involved resources are just too complex. Therefore there is a special necessity in a one-off-a-kind production for tools to manage the dynamic relationships. Simulation technology has been successfully used and integrated at shipyards for these purposes.

Since about 1997 Flensburger Shipyard has been working intensively on simulation in production and logistics. In the beginning facility layouts were evaluated by simulation to support several investment projects. After the successful application in these projects simulation was integrated into the production planning and control, Steinhauer (2005). One of the most important aspects of using simulation in production planning is the increase of planning reliability and robustness as well as the minimization of risks. Because the major impacts on the production by the product, the building method or other constraints can be analyzed dynamically, possible bottlenecks are detected and avoided in advance. The plan can be verified and optimized by building the ship virtually in the computer before building it in the real shipyard.

The success of the first simulation projects and the potentials that became apparent resulted in a more general simulation strategy at Flensburger Shipyard by the end of the 1990s. Due to the experiences that had been gained with the simulation tools used at that time a change of the software and the modelling philosophy became necessary. The effort for building up and maintaining the simulation models had been too high.

Building simulation models of production processes can be done efficiently using predefined object libraries. But the available simulation tools are not sufficient for the usage in shipbuilding since the development of these tools was primarily initiated and driven by the automotive industries. For these or comparable industries libraries of simulation tools are available. The demands of the shipbuilding industries could not be met by those libraries. The main reason was their focus on line oriented production processes compared to the site production typically used in shipbuilding. Furthermore the work packages to be managed in shipbuilding are a lot more complex.

At Flensburger shipyard the development of a library of reusable simulation tools for shipbuilding procedures started in the year 2000. The Simulation Toolkit Shipbuilding (STS) is based on the simulation software Plant Simulation by Siemens PLM, formerly known as Simple++ or eM-Plant by AESOP respectively Tecnomatix. Plant Simulation as an object oriented simulation software provides the functionality for programming and administrating reusable simulation modules.
STS contains a large variety of simulation tools for material flow modelling, model management, execution strategies, data input and output analysis. In the meantime only few tools are still strongly related to shipbuilding production respectively to steel fabrication aspects. Most of the tools provide a more general functionality for fabrication, assembly and logistics usable in many industries having comparable production conditions to shipbuilding.

Today, STS is the backbone of the simulation work at Flensburger shipyard as well as in other companies, universities and research institutes not only related to shipbuilding but also to other industries like the building industry. Modelling expertise from different branches and application scenarios meets in STS and furthers the development of its tools.

2. Philosophy of STS

The prior intention of STS is to enable the user to build up simulation models of production procedures in shipbuilding or a comparable industry in a most efficient and effective way. The effort for the modelling process shall be minimized by providing resusable simulation tools to be combined to a simulation model for the special purpose. Not only the installation of the pure functionality, but also the model building process and the operation of the model shall be supported. Therefore a set of administrating tools is additionally included providing support in structuring and organizing the model. For several typical application cases model patterns are available to be inserted and adjusted to the special requirements.

Soon after starting to use the simulation in shipbuilding it became obvious that the individual adjustability of simulation models is of major importance. Not any shipyard strictly follows a standard routine in running its production flow. Special adaptions are very common due to the special ship types to be built, special restrictions of the shipyard or even special habits of the operating staff. Therefore an important aspect of STS philosophy is to provide interfaces for user defined rules and specific settings.

Another aspect of STS’ philosophy is the cooperative development. The simulation team at Flensburger shipyard would not be able to build up, maintain and develop such a software package with sufficient power on its own. Therefore cooperation communities where founded to develop the tools and to organize the application. Requirements for further development from application cases in different industries are coordinated and integrated to bring up general solutions for the simulation challenges.

The first cooperation founded in the field of simulation was the Simulation Cooperation in the Maritime Industries (SimCoMar, www.simcomar.com) bringing together shipyards, universities and a research institute. SimCoMar has become the maritime forum for the joint development of new or existing simulation tools, for mutual support in implementation of simulation technology and for initiating research and development in the field of simulation.

In 2006 SIMoFIT (Simulation of Outfitting in Shipbuilding and Civil Engineering, www.simofit.com) was founded as an interbranch cooperation between shipbuilding and civil engineering Steinhauser (2007). Flensburger Shipyard, the Bauhaus-University Weimar, the Ruhr-University Bochum and SimPlan AG as a simulation consultancy join this cooperation. Outfitting processes in shipbuilding and building industry bear a high resemblance to each other. The same restrictions have to be considered such as dependencies between outfitting tasks, availability of resources and required work spaces as well as changing transport ways. In addition the planners have to answer the same questions: how to find a practicable schedule with sufficiently utilized equipment and employees satisfying principal guidelines. In the interbranch team of SIMoFIT methods for outfitting simulation are further developed and used in various fields. One of the major steps forward was the adaption of the constraint satisfaction method to the simulation. Complex dependencies in the production flow can now be modelled and considered in the simulation.
Another aspect of the STS' philosophy is the integration of well-proven modelling approaches for a multitude of different production scenarios. Not least, the less experienced simulation user is supported in the work of designing and building a simulation model.

3. Model building using STS

Using STS two basic modelling approaches can be chosen which are different but not mutually exclusive:

- The first approach is a more abstract definition of processes, their constraints and the required resources on any level of detail. For this definition a generic method was developed in order to minimize the effort for process and constraint definition. As part of this method process patterns can be defined including their typical process constraints. Global constraints can be generated from definitions on different levels of abstraction. This simulation approach enables the user e.g. to build up simulation models of flexible assembly processes for outfitting of ships or constructing of a building.

- Following the second approach the modelling is geared by the production facilities to be modelled. Tools for the single machines, production facilities or transport means are inserted and linked along the material flow by logistic functions.

In many applications both approaches are used and merged because a flexible definition of processes is as important as the consideration of the definite production environment e.g. in a production hall.

There is a strong communication on different levels between the tools of STS. Software design patterns like the observer pattern are implemented for managing the communication. Therefore information about the collaborating tools is available where it is required when building up the model and during the simulation run, e.g.

- the available qualifications of the personnel are listed when the required resources for running a machine are selected or

- the crane transport from one station to another is automatically calculated and performed based on the positions of the stations and their loading or discharging specifics.

When having inserted a simulation tool of STS or when building up a model there is a variety of possibilities to implement user defined rules or processes. STS tools e.g. provide interfaces to user definable controls which can be programmed in Plant Simulation’s own programming language SimTalk. The specific rules or restrictions e.g. for space allocation, resource selection or a special transport way can be implemented into the model in this way.

3.1 Update functionalities

Due to the experiences Flensburger’s simulation team gained with other simulation tools before using Plant Simulation the first functionality to be developed was an update procedure. Existing simulation models using STS tools can be updated to the latest version of STS not losing their parameterization. New functions of existing tools or new simulation tools are available in the simulation model afterwards.

STS is administrated at Flensburger Shipyard. New functions in existing tools or new tools developed by partners are integrated and published in Flensburg. New versions or servicepacks of STS can be downloaded from the STS homepage.
3.2 3D Animation

The animation of simulation models became of more interest when the processes to be modelled became more complex. The animation helps a lot in verifying the model and in communicating it with the staff on the shop floor. Besides of that 3D animation can be of help in addition to the technical results when investment decisions have to be made.

In order to minimise the effort for the user to create a 3D animation of the model a special approach has been developed and implemented into STS. The simulation tools of STS contain 3D information and functionality which can be used as default or user defined. After building a simulation model using STS, the 3D animation can be generated by using the simulation tool STS_ModelGenerator3D. By a mouse click a method is executed and the 3D model is generated fully automatically, Fig. 1.

Additional functionality to improve the 3D scene from a more visual then technical point of view is provided in STS as well:
- the tool STS_HallGenerator to create buildings in the 3D model
- interpolation function for easy definition of paths for camera flights
- methods for the creation of user defined geometry
- import functionality for 3D geometry from CAD

4. Simulation tools of STS

The STS tools are structured by their typical fields of application, Fig. 2. Tools of a general functionality and administrational tools are grouped as “Basics”. This group of tools contains amongst many others the administrational tools, the statistic functions, the personnel control, the space tool and the interface to the optimizing software ISSOP which can be integrated into the simulation model to do simulation based optimization.

All types of facilities and process functions related to steel fabrications form the group “Steel”. Likewise the groups “Transport”, “Logistics”, “Material” and “Interior fitting” contain related tools. Objects for the modelling of “Outfitting” procedures were being integrated based on the results of the Simba research project (simulation object library for ship’s outfitting procedures) that was funded by the German ministry for education and research, Steinhauer et al. (2005). Fig. 3 shows a selection of simulation tools of STS.
4.1 The administration tool (STS_Administration)

The general administration of the STS is done by using the tool STS_Administration which is the first to be inserted into the simulation model. The main functionality of this tool is to manage the loading respectively the update process of the model to integrate the latest version of STS without losing user settings. Additionally general settings can be selected like the language of the user interface for the complete model. Administration functions are provided, e.g. license tests or the management of methods programmed by the user.
4.2 STS_Space

The tool STS_Space manages the allocation of production areas by constructions or parts of different sizes and shapes (Nedess et al., 2007). The production area is modelled as a rectangular matrix with a flexible size of matrix fields. Using this approach it is possible to consider the allocation as accurate as needed. Within the space specific areas can be defined for different purposes e. g.

- blocked areas where nothing can be placed (crane posts, buildings, ways, et cetera)
- areas for special purposes (building sites, storage areas, et cetera).

The space tool provides automatic allocation of the space by a set of predefined rules which can be adjusted by the user. Additionally a graphical functionality enables the user to place certain constructions or parts manually.

4.3 STS_AssemblyControl / STS_ConstraintManager

The simulation tool STS_AssemblyControl was in the first phase delivered in close cooperation with Delft University of Technology to manage a variety of parallel assemblies considering individual rules with just one tool (Hertel et al., 2005). The basic assembly management is done by process patterns which can be defined by the user. These process patterns can optionally be structured by defining a certain sequence of assembly stages. The assembly stages consist of work steps associated to part types which optionally can be executed in parallel to a definable extend. By the process patterns assemblies can be standardized without losing the possibility to define individual strategies for special assembly procedures.

As there are possibilities provided in most STS tools the functions of the STS_AssemblyControl can be tailored to the specifics of every application by programming user-defined controls. By these controls the assembly process can be modified in many ways to fit the problem’s needs.

In order to consider the multitude of constraints ruling complex assembly scenarios the STS_AssemblyControl can be combined with the tool STS_ConstraintManager, (König et al., 2007). This tool assures the fulfillment of constraints to execute assembly work steps. There can be several types of constraints to be managed like

- predecessor and successor relations between work steps,
- simultaneous starts of work steps or
- limited amount of parallel work steps.

4.4 STS_Crane

One of the most important transport means in shipbuilding or in comparable industries is the crane. Several different types of cranes are used and are to be considered in the simulation. For this reason the tool STS_Crane was developed and integrated into STS. Using the STS_Crane four different crane types are covered:

- gantry crane
- overhead crane
- tower crane
- movable luffing and slewing crane

Crane moves between different locations can be managed in the simulation model easily by inserting a requirement for the part to be moved and its destination. Allocation of a sufficient gantry – if more than one is defined – and the crane motion itself is managed automatically. To consider specific transport requirements typical at shipyards there are additional function available as options:
- crane moves using more than one gantry
- turning parts during crane moves
- definition of specific transport paths
- coordination of multiple gantries on up to three levels on top of each other

4.5 STS_Statistics

The output data collected during a simulation run is standardized in the tool STS_Statistics. This utility is embedded in all of the simulation tools that provide material flow or resources. The collected data contains inter alia

- the utilisation of resources (time slice for each state of the resource),
- part statistics (start time, end time and duration for each part and process),
- chronological changes of selected parameters.

The collected data can be visualized by a selection of different types of diagrams, e.g. pie charts, Gantt charts or bar diagrams.

The statistics of the complete model can be collected in the simulation tool STS_ModelStatistics. Functions for the comparison of output data from different simulation tools can e.g. be used for bottleneck analysis. The simulation tool STS_ModelStatistics also provides a functionality to create standard reports from simulation runs and an interface to export the output data into external software systems for further analysis, distribution or storage.

5. Examples for application scenarios

Using the STS simulation tools, a variety of application scenarios can be covered. Some typical scenarios from shipbuilding are described below.

5.1 Part fabrication

The build up procedure of a simulation model of part fabrication processes follows the production flow and its required facilities. Plants like cutting machines or profile cutting robots are inserted into the model by drag and drop. The inserted tools are then adjusted to the requirements of the specific fabrication scenario. For example, the cutting machine for plates provides parameters like

- cutting speeds depending on the thickness of the plate,
- speeds for additional processes like marking, signing or grinding or
- times for setting up the complete machine or the particular tools.

Additional to the process related parameters the layout of the cutting machine can be configured to the requirements as well. The size of the machine, the number of carriages or the selection or positioning of the tools on the carriages can be defined for each particular machine.

The STS tools for cutting machines or cutting robots also provide an interface to the original NC code for the real machine. The movements of the plant and the executed processes can be taken from the original control data to get a more accurate result and to take the specifics of the fabrication program into consideration.

If the logistics in the simulation model are to be considered, the transport of the stock material or the parts can be executed by STS_Crane, a forklift controlled by STS_TransportControl or STS tools for conveyors. The buffering of the material could be modelled by the STS_Space if space allocation is to be analyzed.
5.2 Assembly of steel constructions

The assembly procedures of steel constructions dominate large parts of the production in shipbuilding. The constructions are usually assembled in a certain area of the shipyards using cranes. The components to be assembled are provided by heavy load vehicles, forklifts or trucks depending on their size and weight.

STS offers a set of tools to model assembly processes for steel constructions. The main aspects of the assembly can be modelled by the combination of the simulation tools STS_Space, STS_Crane, STS_AssemblyControl and STS_ConstraintManager as described in chapter 4 (Fig.4). The required personnel can be provided by the tool STS_PersonnelControl.

5.3 Space allocation

Space allocation is a typical challenge in production planning not only on shipyards. The steel constructions of a ship vary a lot in size and shape and they have to be placed in halls with restrictions in door sizes, floor configuration or site arrangements. Movements of constructions still have to be possible and not too much space is to be wasted. The dynamic aspect of constructions to enter and leave on different times increases the complexity a lot compared to a simple nesting problem.

A simulation model built using STS can support this production planning task by providing space allocation functions, automatic placing strategies and the analyzing functions for the utilisation of resources and the compliance with the schedule. The models can be built up by one or more instances of STS_Space, possibly combined by logistic functions if this is to be taken into consideration. The specific rules for space allocation in the regarded production area can be programmed in user controls or defined in specific rule definition tables. Predefined allocations from other systems can be imported and considered in the simulation as well.
5.4 Outfitting / Refurbishment

Outfitting processes are distinguished by interferences, disturbances, great interdependencies and different surrounding area requirements, Steinhauer (2010b). A multitude of requirements such as technological dependencies, resource and work space assignment have to be considered. In addition, the assignments of employees and equipment have to be regarded as well. Consideration of all the different restrictions and requirements result in a wide choice of practicable outfitting schedules. Detailed simulation of outfitting processes is challenging for two reasons: first of all the complex restrictions and requirements of the outfitting procedures have to be modelled and secondly the data about the outfitting parts and their dependencies are often not at all or not sufficiently available.

Building a simulation model of outfitting processes starts by the definition of the processes and constraints based on the product’s structure. For the process definition there is a generic method implemented in order to minimize the effort. Work steps for typical part types are defined including the required resources and process times or algorithms for the calculation of process times. Those work steps are combined to process patterns. These process patterns are supplemented by a set of default constraints between the work steps not only to define the typical sequences of work steps for one part but also to define general constraints between work steps. Based on the part lists for the assembly and a classification structure for parts the predefined process patterns can be applied to generate the complete list of processes automatically including their default constraints. These constraints are then complemented by the specific constraints for this outfitting compartment or the regarded system.

The simulation model executing the work steps is built of the simulation tools STS_AssemblyControl, STS_Space and STS_ConstraintManager combined within the simulation tool STS_Compartment. This STS_Compartment represents one part of the building or ship to be outfitted.

The simulation approach for outfitting was recently applied for building a simulation model of refurbishment processes on a passenger ferry, Fig. 5. Before the work steps for outfitting can be executed the disassembly of the existing cabin area has to be carried out. The process data and model for this application could be defined and built as described above for the outfitting processes. The simulation tool STS_AssemblyControl also provides functionality to run work steps for disassembly which have to be related to the outfitting work steps by additional constraints.

Fig.5: Simulation model of refurbishment processes in 3D animation
6. Integration of the simulation model into the data environment

Additional to a verified and validated model the input data is of major importance to get sufficient simulation results. Especially when using the simulation as a support for production planning and control the acquisition of the current data about the product, the production planning and the resource availability is as much necessary as challenging. At Flensburger shipyard the data is continuously collected from the different IT systems into the so-called Simulation Database by various interfaces. The STS tool STS_Data provides interfaces to select and import this data into each simulation model.

Because the availability and acquisition of input data is the major obstacle in using simulation at shipyards research was required in this field. In October 2009 the GeneSim (Generic Data and Model Management for Production Simulation in Shipbuilding) project started, funded by the German ministry of economy and technology (FKZ 03SX274), Steinhauer (2010a). The GeneSim consortium covers a variety of shipyards working on completely different types of ships like freight ships, yachts or submarines. This group of shipyards is reasonably completed by a company very experienced in simulation consultancy and a university for the scientific support. All partners in GeneSim use STS for their simulation modeling.

In the GeneSim project the data required for simulation of production processes shipbuilding are to be defined and structured in a generic way by the partners from the shipyards supported by a university and a research institute. The data requirements of the simulation were derived by the data structure within the STS. The aggregation of data from the generic database of the simulation data is done in a way that is compatible to the STS.

7. Add-on tools to the STS for special applications

Additional to the basic function for the simulation of production flows in shipbuilding or in comparable industries the STS provides functionalities for special applications in terms of production or logistics. Internal projects at Flensburger or external projects for customers often worked as the point of origin for this kind of tools.

7.1 Loading and discharging – STS_Shiplog

Flensburger Shipyard not only offers a pure ship but a solution for a transport problem. Therefore the overall performance is a very important aspect in the design of the ship at Flensburger. The minimization of the port time can be a significant contribution to this overall performance. Flensburger shipyard has been using simulation of loading and discharging processes for many years in order to evaluate the ships performance and to optimize its logistics, Soyka and Steinhauer (2008). The simulation tools for discharging and loading of ships are organized in the STS group STS_Shiplog and primarily linked to RoRo and RoPax vessels. The tools contain inter alia

- a loading control to manage the discharging and loading procedure,
- ramps including their logistical restrictions,
- a deck control to manage the logistics on a deck,
- areas for manoeuvring trucks of tug masters or
- storage areas on the deck.

Currently the tools for loading and discharging of ships are further developed in a work package of the European research project BESST (Breakthrough in European Ship and Shipbuilding Technologies) together with partners from shipyards and a research institute.

7.2 Logistics in ports – STS_Portlog

The ship-shore-interface gets more and more in the focus of the simulation because of its significant
impact on the overall logistic chain. An increasing number of shipowners and terminal operators request evaluations concerning this interface. Therefore Flensburger Shipyard in cooperation with Technical University of Berlin have been developing simulation tools for port logistics primarily focused on RoRo or ferry terminals Eckert, Fliege and Steinhauer (2008).

The tools for terminal logistics were successfully used in projects for customers already, Fig. 6. The overall cargo flow in RoRo terminals and the size of buffer areas were analyzed in these projects.

### 7.3 Installation of offshore wind parks – STS_Offshore

Due to design projects from the recent past, Flensburger’s simulation team has performed simulation studies for the installation of offshore wind parks, Fig.7. Different alternative procedures of the installation process can be modelled and evaluated. All the parameters of the installation process can be varied like the type of the installation vessel including its attributes or the strategies for supply or assembly. Many constraints and restrictions can be taken into consideration in the simulation model. In these special applications the weather influence is of major importance. By evaluating different alternative scenarios the over-all installation process can be optimized with respect to shortest installation times and highest robustness of the schedule.

One simulation tool having its source primarily in the offshore wind park projects is the tool STS_Weather. By using this tool weather parameters can be imported respectively generated and afterwards considered in the simulation. For this consideration different possibilities are provided:

- The weather parameters can be aggregated to so-called weather conditions which can be constraints for a certain production process. The jacking process of an offshore vessel can only be executed up to a certain wave height and wind speed. Work on the facing of a building can only be done up to a certain wind speed and not when it is raining.
- Certain weather parameters can have impact on certain process parameters. The speed of a ship depends on the wave height or the drying time for concrete depends on the air temperature.

As sources for the weather parameters real or calculated weather parameters for the relevant locations can be imported. Functions for stochastic weather calculations can be used as well.
8. Outlook

Currently the further development of STS is strongly related to the cooperation with the civil engineering. Tasks like central resource control, model generation from external data sources as well as flexible construction site logistics are being worked on at all partners of the SIMoFIT cooperation.

One continuous task in the STS development is to increase the performance of the simulation tools or their communication. Still large and complex models require a long time for a simulation run which limits the search space for model parameters to be varied in order to improve the simulation results for a certain task. Latest improvements in the basic software Plant Simulation will be tested and used in order to gain maximum simulation performance.

There is an increasing interest in the maritime industries as well as in comparable industries for the application of simulation for optimizing processes or for improvement of the production planning. In projects for other companies or in possible future interbranch cooperation, STS will improve its functionality and integrate new functions with respect to the new applications.

References


Conf. Computer Applications and Information Technology in the Maritime Industries (COMPIT), Hamburg


STEINHAUER, D (2010a), *GeneSim - Development of a generic data model for production simulation in shipbuilding*, 9th Int. Conf. on Computer Applications and Information Technology in the Maritime Industries (COMPIT), Gubbio


STEINHAUER, D; HÜBLER, M.; WAGNER, L (2005), *SIMBA: Entwicklung eines Simulationsbau- steinkastens für die Schiffsausrüstung*, Final Report, FSG, Flensburg

SOYKA, M.; STEINHAUER, D. (2008), *Simulation of logistic processes on board of vessels*, 13th ASIM Conf. on Simulation in Production and Logistics, Berlin, pp. 91-100
FSG.EcoPilot – An Onboard Tool for Fuel Efficient Speed Profile Selection

Anna Bruns, Flensburger Schiffbau-Gesellschaft mbH & Co. KG, bruns@fsg-ship.de
Katja Christiansen, Flensburger Schiffbau-Gesellschaft mbH & Co. KG, katja@fsg-ship.de
Dirk Rossow, Flensburger Schiffbau-Gesellschaft mbH & Co. KG, rossow@fsg-ship.de

Based on the numerical methods developed through the last 20 years in various research projects FSG has developed an interactive onboard tool for operator guidance. The FSG.EcoPilot assists the crew in fuel efficient ship operation. For a specified route the software tool recommends a speed profile considering water depth, weather forecast and time schedule. Several alternative routes can be defined to determine the optimum with respect to fuel consumption for propulsion considering the above parameters at the given time.

1. Introduction

In the recent past, fuel efficiency has been a major topic among both ship designers and ship operators. FSG has been increasingly confronted by higher requirements of both society and customers for economic and ecological vessels. FSG’s focus is not only on the tailor made design with an optimum of cargo capacity, safety and fuel efficiency but also on guidance to the operator for a cost-effective operation. This philosophy is still believed to be our greatest advantage in competition with other ship yards especially from Far-East. The knowledge gained during the ship design and construction was in the past primarily transferred through written documents. The requirements of the customers for a route specific guidance and on the same time altering operation areas cannot be met with the media of an unchanging document. Based on the numerical methods developed through the last 20 years in various research projects FSG has developed an interactive onboard operator guidance. This software tool FSG.EcoPilot assists the crew in fuel efficient ship operation. For a specified route the software tool recommends a speed profile considering water depth, weather forecast and time schedule. Several alternative routes can be defined to determine the optimum with respect to fuel consumption for propulsion considering the a.m. parameters at the given time.

1.1. Performance Manual – Communication of yesterday?

The performance manual is a document that FSG as standard delivers to the crew at the delivery of the vessel. This manual explains hydrodynamic aspects, gives advice for better performance and cautions the crew about the hydrodynamic limits of the vessel. Typical topics are:

- Speed-power; deep water performance, dynamic sinkage and trim, recommendation of the immersion of the bulbous bow, shallow water performance etc.
- Manoeuvring; Course keeping in strong winds, turning ability, crash stop performance, crabbing performance etc.
- Propulsion concept; shaft generator, main engine emergency shut down
- Seakeeping; dynamic intact stability in heavy seas, performance of roll damping systems, accelerations

The content of this manual is gathered throughout the design and construction phase including the sea trials. To condense the enormous amount of data into a consistent and useful printable picture requires a balanced consideration of the interest and need of the crew and our drive to inform. In this process naturally a lot of the available information remains at the yard. In addition reasonable suspicion is entertained that large portions of the manual is difficult to use in practice and consequently seldom read. The reason for that lies in the dissected way of presenting the hydrodynamic topics. In reality all hydrodynamic areas as listed above act simultaneously and interact. This interaction can be analyzed by the numerical programs that FSG has developed. To make these methods and our prepared data in more comprehensive manner accessible to our customer than it is possible today is the goal of FSG. The software FSG.EcoPilot is a first step to make FSG’s
sophisticated numerical methods easily available for the ship operator.

2. Program Structure

The task of the program is to assist the crew in choosing a fuel economic speed profile. In defining several alternatives a more fuel economic route may be found. The basis for the analysis is the calculation of the resistance depending on the relevant operational and environmental parameters. Knowing the total resistance a propulsion point is calculated taking the open water characteristics of the propeller, the combinator diagram and the load limit curve by the main engine into account.

The kernel of the program is an adoption of already developed method integrated in FSG’s ship design environment (RDE-System, formerly E4). In the following sections a short overview of applied methods and their theory is outlined. For more details reference is made to Söding (1993,1994), Krüger (2009), Billerbeck (2009).

2.1 Resistance

2.1.1 Still-Water Resistance

The still-water resistance is calculated using a potential flow method. The precision of the CFD-method is confirmed through model tests for each project and newbuilding. In these model tests generally the design and sea trial loading conditions are analyzed. The database in FSG.EcoPilot is extended using the CFD-method and the wide experience of the engineers to cover all legal and sensible floating conditions (in draught and trim). As the CFD-method is used and verified extensively in the range of design and ballast loading conditions the calculated still water resistance in these areas is fairly accurate. Though, in so called ‘off-design’ loading conditions this approach leads to uncertainties which are growing with the distance to the known model tests. The uncertainties can be controlled and minimized by a systematical series of model tests. An onboard data recorder is used to synchronize the calculations with the real operation. The recorded values are used after a certain time of operation to update the database of the software.

2.1.2. Shallow Water Resistance

The effect of shallow water on the resistance is taken into account using the method by Schlichting. The basic assumption is that the wave making resistance of the vessel in shallow water already is reached at a certain lower forward speed \( v_H \) compared to the deep water condition at a higher forward speed \( v_\infty \). The relation between the reference deep water speed and the reached shallow water speed is given by the following equation:

\[
v_H = v_\infty \cdot \sqrt{\tanh \left( \frac{gH}{v_\infty^2} \right)}
\]

\( g \) is the gravitational acceleration and \( H \) is the water depth.

The correction of the increased flow around the body at very shallow waters is given by a polynomial with the variable \( X \) which is related to the cross-sectional area of main section \( A_m \) and the water depth \( H \) according to the following equation, Lewis (1988):

\[
X = \frac{\sqrt{A_m}}{H}
\]

\[
\frac{v_{HW}}{v_H} = 0.056X^4 - 0.126X^3 - 0.05X^2 + 0.006X + 1
\]
2.1.3. Wind

The resistance due to wind $R_{\text{Wind}}$ is calculated following the equation:

$$R_{\text{Wind}} = \frac{1}{2} \rho_{\text{air}} C_{w} V_{w}^{2} \left( A_{T} \cos^{2}(\theta) + A_{L} \sin^{2}(\theta) \right)$$

$\rho_{\text{air}}$ is the density of air, $C_{w}$ is the wind coefficient, $V_{w}$ is the relative wind speed, $A_{T}$ and $A_{L}$ are the transverse and lateral projected areas respectively and $\theta$ is the relative wind direction. The wind coefficients are a result of model test in a wind tunnel of the current design or a comparable vessel.

2.1.4. Wave

The added resistance due to waves is calculated using strip theory, Söding (1993). The chosen wave spectrum is a JONSWAP spectrum as many of our customers operate in Short-Sea Shipping, close to the coast and in fetch restricted waters. The wave periods cover the range from $1/3$ to $3/2$ of the ship length.

2.2 Input Data

For the calculation of resistance and the fuel consumption not only the ship data is necessary but the environmental data is essential as well. The environmental parameter that influence the resistance the most are the water depth and the weather parameter such as the wind speed, the wave height etc. The water depth is given by the position of the vessel. In the program the depth information is linked to the position in the geographical map. The weather forecast information is included in a data file that regularly on a daily basis is send to the vessel.

2.2. Propulsion and Fuel Consumption

In order to estimate the self-propulsion point, all resistance components are summarized over the according speed through the water, taking all environmental influences into account. The self-propulsion point can thus be calculated with the openwater characteristics of the propeller. Since controllable pitch-propellers are regularly used for RoRo-vessels operating in Short-Sea Shipping, the characteristics of the propeller with different pitch settings are estimated with this method as well.

Fig.1: Estimated combinator mode and const. rev. mode (twin engine operation)

RANS (Reynolds averaged Navier-Stokes) methods are used to estimate the characteristics over the whole operating range, Staye (2011). When calculating the optimum fuel consumption, the specific fuel consumption (SFOC) of the main engine needs to be taken into account as well. An optimal
power demand on the shaft line may not automatically lead to the best fuel consumption, especially if the main engine is running at unfavourable revolutions. During the design of the vessel, the combinator diagram can thus be optimised for an efficient operation and giving the vessel always sufficient safety against the onset of face side cavitation in all operating conditions. During operation, the optimisation criterion is hence the fuel oil consumption per nautical mile rather than only optimising the power of the main engine. Unfortunately, detailed data of the specific fuel oil consumption of the main engine is not available from all engine manufacturers, Fig. 1.

3. Programmatic implementation

The methods for calculating the required information have been present at FSG for a long time. They are implemented and used within the ship design software of FSG (E4/RDE). Those methods are fairly technical in their usage and presentation because they are intended to be used during the technical design process of a ship.

An operator on the bridge of a vessel, however, needs a different representation of the data and an easy operability to gain the most benefit from the software. Therefore, the approach was to develop a tool that focuses on:

- reliable calculation results based on well-known methods and specific ship and environment data
- clear representation (e.g. display of routes and weather data on a map)
- ease of use (e.g. fast creation of alternate routes by using copy and graphical edit mechanisms)

The FSG.EcoPilot was developed with Microsoft Visual Studio C# 2010 using Microsoft .NET Framework Version 4. The user interface was built with Microsoft's Windows Presentation Foundation (WPF, a graphical subsystem for rendering user interfaces in Windows-based applications) and the Extensible Application Markup Language (XAML, used as user interface markup language). This allowed for easy integration of existing Windows functionality and external modules, and a highly flexible and adaptable user interface design.

The available depth data was coupled with the geographical coordinates of the map area to define the environment used for basic calculations. For more advanced calculations, FSG.EcoPilot includes the functionality to retrieve weather data files automatically or manually from a weather data file provider by E-mail.

The available weather data can be shown as an additional layer on top of the map to assist the operator in planning a route. Routes can easily be added and modified by using either graphical editing or numerical input.

4. User Interface

The innovative aspect of the program is not each single component, which have existed at FSG for some time but the more profound integration of these numerical methods into a user friendly software package. The core of the user interface is the geographical map, Fig. 2. On the right side control buttons of the map are positioned. It is possible to zoom in and out of the map as well as pan.

On the map the isobar contours can be displayed for all available data in the past and future. The intention is to easily show how far the weather data extend geographically and control whether the necessary weather file is available. Outside the area with weather information the calculation may still be carried out but without taking into account the influence of the weather. In this case the only environmental parameter that is included in the performance calculation is the water depth. The weather file provided by mail to the ship also contains wind, wave and current information.
The user needs to define a route for which the calculations are carried out. The route may be defined by picking waypoints on the map or by specification of Lat/Long of each waypoint. The sensible division of the route is the responsibility of the user. In addition the user needs to define the operating mode; constant revolution with optional the power requirements of a shaft generator (Power take-off, PTO) or the combinator mode, draught and trim, and optional the passage time available for the whole route. If the user omits the specification of the planned passage time, the program assumes that maximum power is available and the result is the minimum passage time.

In Fig. 2 is an example of a route from Lisbon to Oslo. In the table straight below the map are all defined routes described. A route consists of a number of segments; each segment is described in the table below the route-table. In this table is the optimal speed profile presented, for each section a recommended speed is shown.

Further results of the calculations are the percentage of fuel save which is based on the calculated fuel consumption in relation to an ‘average speed’. This reference average speed takes into account the environmental parameters. That is, if in one segment the numerical average speed cannot be reached, the maximum possible speed is assumed for this section and for the remaining segments a new average speed is calculated satisfying the specified planned passage time.

The total fuel consumption is a result as well, displayed in the rightmost column. This value is an estimate of the fuel consumption for the propulsion. The amount of power requirement for the PTO is not included in this value. The power for the PTO shifts the SFOC within the engine characteristics, Fig. 1, whilst the electrical power is produced by the auxiliary engines if the PTO is switched off. Attention must be paid to the use of the fuel consumption as this value is an estimate and should not be compared directly with fuel consumption on board due to uncertainties in the data and the restricted focus on the propulsion. Nevertheless this is a comprehensive value for the comparison with other routes where the fuel consumption is calculated under these terms.
5. Example

The defined route Lisbon to Oslo is taken as an example. Several alternative applications of the program based on this example route are shown in the following sections.

Fig. 3: Example route Lisbon - Oslo

5.1 Speed profile

The primary and first result of the program is the speed profile as displayed in Fig. 4 graphically and in Fig. 5 as table. Based on the waypoint specified by the user the optimum speed for each segment is calculated and displayed in a table. Internally the user-defined sections are divided additionally to include the effect of changing conditions (for example depth) which occurs across a segment. The speed recommendation for the user-defined section is an average of all internal subdivisions.

Fig. 4: Example route Lisbon - Oslo
In the recommended speed profile the average speed is kept in the segments 2-4 and 6-7. It is apparent that in the segments 1 and 8 the recommended speed is close to the maximum speed whereas in segment 5 and 9 the recommended speed is relatively low. This is for once due to the shallow water in the English Channel (segment 5) and the comparable deep waters in the Atlantic (segment 1, 8) combined with favourable weather conditions.

5.2 Comparison between two different routes

The original route from Lisbon to Oslo is copied and two waypoints (No. 2 & 3) are shifted resulting in a reduction of the travelled distance of 26 nm (-1.7%).

![Fig. 5: Comparison of two different routes](image)

The screen shot in Fig.4 shows the results:

1. For the new shorter route ‘Copy of Lisbon-Oslo’ the improvement by following the recommended speed profile is approx. 3% compared to the ‘average speed’-profile. Compared to the original route the potential of saving fuel by complying with the recommended speed profile is reduced. That means that the optimal speed profile of the second route is closer to the ‘average speed’-profile.

2. When comparing the total fuel consumption (for propulsion) the second route shows a better performance. This is additionally marked by the medal-symbol in the info-column to the right of the fuel-column.

The FSG.EcoPilot is not a route optimizing tool, where waypoints automatically are optimized to minimize the fuel consumption for a given passage between departure and arrival port.

5.3 Different operational modes (const. rev and combinator mode)

In the following example are three equal routes defined, which only differs in the operating model. The original route (route 1) is operated in the constant revolution mode, the second route using the combinator mode and in the third route the constant revolution mode is combined with a shaft generator consuming constantly 1500kW. The result shows that the fuel saving potential by following the recommended speed profile has its maximum in the third route. In this operational combination 4% fuel could be saved if the speed is chosen according to the recommended speed profile. Though, in direct comparison with the other alternatives the configuration operating in combinator mode results in the lowest overall fuel consumption (for propulsion).

Because the electrical load is not taken into account and neither is the fuel consumption of auxiliary engines the conclusion, that operating in combinator mode is generally favourable, is not possible on this basis. But in house calculations, Stoye (2011), have shown that this positive effect indeed has significance when the electrical load is small compared to the power requirements for propulsion and the vessel is operated in ‘off-design’ conditions.
6. Conclusion and discussion

The objective of a better conveyance of generated data by the yard to the customer at delivery of the vessel is reached for a certain portion of available information – i.e. speed-power, propulsion, fuel consumption and effects of depth and weather on fuel consumption. Information regarding manoeuvrability and seakeeping can still only be given in the form of written recommendations like FSG does via the performance manual supported by direct discussions and explanations. Additionally the program does not explain why the fuel consumption in some configurations is lower than in others. The explication for that can thus not be omitted in the ‘old school’ way of communication.

A different, quite difficult aspect is the legal responsibility for the operation of the vessel. The master is and should always be is the position to make the best choice for the welfare of the crew and the vessel. A software tool is in any way only a help for the decision-making of the crew. Increasing automatism on the bridge always comes in hand with a possible decreasing awareness of the operation of the vessel. Consequently a program as FSG.EcoPilot can only be regarded as a supplement to the usual way of informing about the technical background of the performance advantages and limitations of vessels ‘made by Flensburger’.

References

BILLERBECK, H. (2009), (Fuel) efficiency versus safety in ship design, Ship Efficiency Conf., Hamburg

KRÜGER, S. (2009), Strömungsmechanische Grundlagen zum Glattwasserwiderstand, Lecture Notes, Hamburg University of Technology, Hamburg

LEWIS, E.V. (1988), Principles of Naval Architecture, SNAME

SÖDING, H. (1993), Programm RSEE zur Berechnung des Zusatzwiderstands im Seegang, Institut für Schiffbau, Hamburg

SÖDING, H. (1994), Beschreibung des Programmes STRIP, Institut für Schiffbau, Hamburg

STOYE, T. (2011), propeller design and propulsion concepts for ship operation in off-design conditions, 2nd Int. Symp. Marine Propulsors (SMP), Hamburg
Ensuring Stability in Synergistic Computing: Combining Flow Simulations and Neural Nets to Predict Maneuvering

David E. Hess, Naval Surface Warfare Center, Maryland/USA, david.e.hess@naval.mil
William E. Faller, Applied Simulation Technologies, Florida/USA, faller@appliedsimulation.com

Abstract

A potential flow code for maneuvering predictions has been augmented with feedforward neural networks (NNs) to adaptively estimate the force and moment residuals that represent the differences between predictions and experiments. This is an example of Synergistic Computing, where a physics-based code (MCSIM) is combined with NNs to provide a superior hybrid computing tool. The feasibility of this approach was demonstrated by using the combined MCSIM+NN tool to produce significantly improved six degree-of-freedom maneuvering predictions for maneuvers of an unclassified submarine free-running model. This paper will present an overview of the Synergistic Computing approach and provide an analytical method to identify and remove a small number of NN inputs that cause instability. This work clearly demonstrates that it is possible to create synergistic computing tools with a significant increase in maneuvering simulation accuracy.

1. Introduction

A critical issue for submarines is safety as characterized by the submerged operating envelope (SOE). In particular, casualty conditions and recovery procedures can significantly limit the speed and depth profile of the vehicle. Accurately predicting casualty conditions using a maneuvering simulation code has proven to be a very challenging problem. Predicting other types of maneuvers, such as horizontal plane maneuvers and combined plane maneuvers can also be problematic. Similarly, critical issues for U.S. Navy combat ships are safety and operational capabilities in high sea states. This has been a driving force behind the current Naval push for the development of new time-domain simulation tools and advanced control strategies for surface ships. The capability to simulate these highly nonlinear conditions is recognized as a critical enabling technology for future platforms.

Submarine forces and moments, which direct the motion of the vehicle, result both from the controls, the propeller, and vortex dominated separated flow fields. Surface ships are acted upon by forces and moments resulting both from the controls on the vehicle as well as from the sea environment as described by the wave field and wind. The vehicles, in turn, serve as nonlinear transfer functions for the forces and moments, with the end result being the observed vehicle motions over time. The observed responses are nonlinear; the challenge is how to simulate these types of conditions. The choice of codes is often a trade-off between speed and accuracy. The opportunity exists to avoid this trade-off by synergistically coupling two codes to simulate nonlinear conditions with increased accuracy, but without increasing the computational time requirements.

Specifically, the US Navy’s Maneuvering and Control Simulation (MCSIM) is the primary predictive simulation tool for developing submarine hydrodynamic predictions, operational guidance, training submarine operators and responding directly to fleet needs. MCSIM is based on potential flow computations that are augmented by approximations of viscous effects in the streamwise and cross-flow directions. In the streamwise direction viscous effects are approximated by potential flow theory if the body is augmented by the displacement thickness of the boundary layer. The separated cross-flow vortex wake is approximated by thin vortex singularities in the form of point vortices. The creation and movement of these point vortex singularities within the potential flow framework comprises the evolution of the wake. Appendage force computations employ lifting line theory, and unsteady effects are captured by tracking the time history of the strength and propagation of the vorticity.

While this real-time code normally provides excellent results, there are instances where maneuvering
predictions do not match well with experiments at model scale, using free-running submarine models (FRM), or with full scale behavior. To address this, improvements in the experimental program with regard to model scaling behavior have been suggested, Shen and Hess (2010,2011). Also, on the modeling side, feedforward neural networks (NN) were recently coupled to MCSIM to provide an adaptive correction to the simulation data such that the sum of the original MCSIM prediction and the NN correction yields the measured vehicle behavior, Faller et al. (2010a). Specifically, NNs were used to estimate the force and moment residuals that, when added to the MCSIM output at each time step, correct for any differences between MCSIM and either FRM or full scale submarine maneuvers. A requirement to ensure success when combining multiple codes, which have competing operational requirements and where the mathematical solutions may differ significantly, is the input-output (I/O) interface between the codes. Since each code has different inherent sources of error, based on the physics and type of solution, the potential exists for these errors to propagate in a destructive fashion between the codes. This positive feedback will lead to significantly worse results rather than the intended improvement in predictive performance. Determining a stable interface without sacrificing the accuracy of the individual codes is critical. An initial graphical approach for determining a suitable interface was found and reported, Faller et al. (2010b). The feasibility of this approach was demonstrated by using the combined MCSIM+NN tool to produce significantly improved six degree-of-freedom (6-dof) maneuvering predictions for maneuvers of an unclassified free-running submarine model.

Ensuring stability in the hybrid computational tool requires removing a small number of the NN inputs that lead to instability in some of the maneuvers (identified graphically), retraining the network, inserting the network in the computational loop of the hybrid code and re-evaluating stability. These steps may have to be repeated, and the process must be accomplished without sacrificing solution quality. The previously mentioned graphical approach, accomplishes this, but is tedious and time-consuming. Is there a way to identify \textit{a priori} which inputs will need to be removed? If this could be done analytically, then the process could be automated. This paper will first present an overview of the Synergistic Computing approach. Then, an automated approach to identify particularly sensitive inputs of a feedforward neural network so that they may then be lesioned to create a stable hybrid computing tool will be described. The method relies upon the evaluation of the partial derivatives as a means to decompose the output of the network into contributions from each input. This automated technique is expected to simplify the identification of the proper I/O interface between the codes and facilitate the use of the Synergistic Computing method.

2. Synergistic Computing

An example of Synergistic Computing is the combination of MCSIM and a set of neural networks to more accurately predict the 6-dof maneuvering behavior of a free-running submarine model, Fig. 1.

![Fig. 1: Implementation of Synergistic Computing](image-url)
accelerations to be determined. Integrating provides the velocities, trajectory and attitude, and the vehicle is advanced to the next time step in the maneuver. The cycle then repeats, and this is the loop shown in the top of the figure.

Analysis of the MCSIM maneuvering predictions and comparison with experiment have indicated that the dominant, required corrections are in the roll, pitch and yaw moments, $K$, $M$, and $N$. This is not necessarily unexpected since the forces rely only on the total pressure difference, whereas the moments require the $x/L$ pressure distribution over the entire vehicle surface to be computed correctly. Therefore, the hybrid computing tool shown in Fig. 1 employs three feedforward neural networks designed to adaptively estimate the required residual moments. Since the moments are directly proportional to the angular accelerations via the moment of inertia matrix, each NN computes one output, which is a correction to the roll, pitch and yaw accelerations, $\Delta \dot{\theta}$, $\Delta \dot{\phi}$, and $\Delta \dot{\psi}$. Initial estimates of the acceleration correction time histories for each maneuver were obtained by comparing the MCSIM predictions to those calculated from the FRM experimental data as shown in Eq.(1).

$$\Delta \dot{\theta} = \dot{\theta}_{\text{FRM}} - \dot{\theta}_{\text{MCSIM}}, \quad \Delta \dot{\phi} = \dot{\phi}_{\text{FRM}} - \dot{\phi}_{\text{MCSIM}} \quad \text{and} \quad \Delta \dot{\psi} = \dot{\psi}_{\text{FRM}} - \dot{\psi}_{\text{MCSIM}}$$  \hspace{1cm} (1)

So, in the bottom of Fig. 1, the three NNs each provide a correction to the moments, which are then summed with those provided by MCSIM to produce total moments that agree with experiment. The feedforward neural networks are fully connected with an input layer, two hidden layers and an output layer, and they are trained using backpropagation. The architecture of one of the NNs is shown schematically in Fig. 2.

![Fig. 2: Typical Neural Network Architecture](image_url)

In order to implement Fig. 2, a set of maneuvers of the free-running submarine model were selected, and MCSIM was used to create predictions for the identical maneuvers. The time histories of the calculated moment residuals along with the maneuvering data and control time histories were stored for each maneuver. This data formed the initial estimates of the required output moment residuals used to develop the NNs. A large set of inputs, those shown in Fig. 2 as well as various products of these terms suggested by traditional hydrodynamic coefficient based codes, were drawn from this stored data and selected as possible input terms to the NNs. Further, several of the inputs such as stern-plane deflection, $\delta s$, rudder deflection, $\delta r$, and propeller rotation speed, $\text{rpm}$, were combined into expressions representing the controlling forces and moments, lift, drag, thrust, etc. The networks were then trained offline; that is, they were removed from the loop shown in Fig. 1 for training, and then the trained networks were reinserted into the loop for evaluation.

After the solution of the stability problems (to be discussed in the next section), Faller et al. (2010b) show simulation results comparing the predicted maneuvers using synergistic computing to those of the CFD code run without the NNs. They report that: “In 100 percent of the cases, the synergistic computing simulations are significantly more accurate.” The reader is directed to that paper to see the results.
3. The Stability Problem

When the NNs are inserted into the prediction loop, the following sequence of events occurs. The input quantities needed by the NNs are provided by MCSIM. The NNs produce the $\Delta \dot{p}, \Delta \dot{q},$ and $\Delta \dot{r}$ residuals corresponding to that set of inputs. However, the residuals are not precisely correct; no simulation is perfect. The residuals are added to produce the total force and moments at that time step and these are imperfect. Integrating provides the velocities, trajectory and attitude, and the vehicle is advanced to the next time step in the maneuver, but there are small errors in all of these quantities. MCSIM uses this imperfect information to provide the next set of predictions. These predictions contain errors from two sources: the error propagated into the inputs to MCSIM, and the fact that MCSIM is itself imperfect and provides some prediction error. Now, this error enters into the inputs of the networks. What happens?

There are two possibilities. The first is that the errors combine in a fashion that produces minimal error at the output. As this process continues around the computing loop, any errors produced at the outputs of the NNs remain uncorrelated with previous errors and do not grow with time. The other possibility is that these errors grow during successive iterations, thereby ruining the prediction for the maneuver; this positive feedback problem initially affected about 50% of the set of submarine maneuvers.

To learn more about the nature of this problem, time series of the inputs and the output for maneuvers that were poorly predicted were studied. An example of time series for a few of the inputs to one of the networks during a poorly predicted maneuver is shown in Fig. 3.

![Fig. 3: Selected inputs to the $y = \Delta \dot{p}$ NN for one maneuver; UL: $p$, UR: $q$, LL: $u$, LR: $v$; Bold: NN, Thin: NN+MCSIM](image)

In Fig. 3 the variables are roll rate $p$ at upper left, pitch rate $q$ at upper right, forward velocity $u$ at lower left and lateral velocity $v$ at lower right. Bold curves represent the histories of these four inputs for a particular maneuver when the NNs were separated from the prediction loop and the information was presented to the networks without any error (correct curves). The thin curves record the history of the same input when the trained network is inserted into the prediction loop and errors are present. Notice how the input terms at lower left and lower right display relatively benign behavior in the presence of error, whereas other input terms at top left and top right rapidly go unstable. The question as to why certain NN inputs are impacted more than others is a complex one, but it is safe to assume that predictions by MCSIM are more accurate for some variables than for others. Furthermore, it is also clear that some of the NN input terms are more sensitive than others. The output predictions for the $\Delta \dot{p}$ residual and the $\Delta \dot{q}$ residual for this same maneuver are given in Fig. 4 and are clearly poor.
The problem, then, is to discover a solution that permits the NNs to operate inside the prediction loop in the presence of unavoidable prediction errors and to produce residual predictions that do not engender positive feedback in the system. The solution requires that inputs that contribute to the amplification of errors (like the top two inputs shown in Fig. 3) be lesioned.

Lesioning refers to the systematic process of removing selected inputs from a NN and then retraining the NN to produce a new solution. This is performed because these inputs are of little or no value to the prediction of the solution or because they actually hinder the prediction of the solution. This is a standard procedure during NN development and is often reported in the literature.

The procedure for addressing the stability problems is to plot all of the input time series for maneuvers that were poorly predicted. Graphically, those input terms which are subject to, or cause, the positive feedback will deviate from the expected behavior (bold curve) before the time at which the output degrades. The procedure is slow and conservative by design. The inputs are studied over the entire set of maneuvers, and a few of the worst-case inputs responsible for amplifying error are removed for each of the NNs. Then, the NNs are removed from the prediction loop and retrained with the reduced set of inputs. Recall that the NNs each have initially a large set of inputs that provide more than enough information to pose the problem well and provide accurate solutions. The reduced set of inputs remains quite large and continues to provide ample information for the NNs to converge to accurate solutions. The fact that the error measures that quantify solution quality remain excellent proves that this statement is true. The lesioning of inputs and the retraining process produce a new version of the NNs. They are then reinserted into the prediction loop shown in Fig. 1 and the performance of the system is reevaluated. Following a number of iterations, the number of maneuvers that displayed positive feedback was reduced to zero; in other words, accurate predictions from the hybrid computer tool were obtained for all maneuvers and the system was working as planned.

The inputs that were removed for each of the $\Delta \dot{\varphi}, \Delta \dot{q},$ and $\Delta \dot{r}$ NNs are listed in Table I below. These are useful to identify as we will refer back to them in subsequent sections.

| Table I: Final set of inputs removed from each of the NNs
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>$\Delta \dot{\varphi}$ NN – Inputs reduced from 24 to 18</td>
<td>7</td>
<td>13</td>
<td>19</td>
<td>21</td>
<td>23</td>
</tr>
<tr>
<td>$\dot{p}$</td>
<td>$qr$</td>
<td>$up$</td>
<td>$\varphi$</td>
<td>$p'$</td>
<td>$\varphi^2$</td>
</tr>
<tr>
<td>$\Delta \dot{q}$ NN – Inputs reduced from 30 to 18</td>
<td>7</td>
<td>8</td>
<td>9</td>
<td>10</td>
<td>12</td>
</tr>
<tr>
<td>$q$</td>
<td>$pr$</td>
<td>$qr$</td>
<td>$uq$</td>
<td>$vq$</td>
<td>$q^2$</td>
</tr>
<tr>
<td>16</td>
<td>17</td>
<td>18</td>
<td>23</td>
<td>26</td>
<td>28</td>
</tr>
<tr>
<td>$(qr)^2$</td>
<td>$(uq)^2$</td>
<td>$(vq)^2$</td>
<td>$\theta$</td>
<td>$q'$</td>
<td>$\theta^2$</td>
</tr>
<tr>
<td>$\Delta \dot{r}$ NN – Inputs reduced from 19 to 17</td>
<td>14</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>$up$</td>
<td>$pq$</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
4. Analytic Approach

The lesioning method described in the previous section relies on the identification of those inputs that greatly deviate from the behavior they displayed during offline training as compared to when they are placed in the NN+MCSIM computing loop. Examples are shown as the top left and top right curves of Fig. 3. Clearly this method works, but is tedious to implement and requires a few iterations of training and evaluation within the computational loop. The question arises then, as to whether an automated method can be found to: probe the inputs of the trained network; evaluate their effect on the solution; remove the inputs likely to become unstable and retrain the network prior to placing the network within the computational loop.

Such a method would not have access to the thin curves in Figs 3 and 4, only the bold curves reflecting the behavior of the inputs during offline training. One would have to have some a priori knowledge, based on the code being merged with the NNs, of which inputs are likely to contain the highest prediction error. For example, after the fact, we can examine the terms that were removed in each of the NNs, shown in Table I, and come to the conclusion that \( p \) and \( q \) are likely to be the terms predicted by MCSIM that contain the most error. This knowledge, by itself, is not enough. These inputs enter into the NNs and propagate through the complex feedforward equations of multilayer networks. How do errors in these inputs affect the output predictions of \( \Delta \dot{p}, \Delta \dot{q}, \) and \( \Delta \dot{r} \) at each time step? Can we discover an approach that can identify the unique contribution that a specific input makes to the change in the output? If we could do that, then we could perturb those inputs that we suspect will contain the most error and can examine their effects on the output change and see if their contributions are destabilizing.

4.1. Opening Steps

To develop such an approach, consider two inputs, \( x_1 \) and \( x_2 \), and a dependent variable \( y(x_1,x_2) \), which may represent the inputs and output of a given neural network. Expand this output function into a Taylor series as follows:

\[
y(x_1,x_2) = y(a,b) + (x_1 - a) \left. \frac{\partial y}{\partial x_1} \right|_{(a,b)} + (x_2 - b) \left. \frac{\partial y}{\partial x_2} \right|_{(a,b)} + \frac{1}{2!} \left[ (x_1 - a) \left. \frac{\partial^2 y}{\partial x_1^2} \right|_{(a,b)} + 2(x_1 - a)(x_2 - b) \left. \frac{\partial^2 y}{\partial x_1 \partial x_2} \right|_{(a,b)} + (x_2 - b) \left. \frac{\partial^2 y}{\partial x_2^2} \right|_{(a,b)} \right] + \ldots.
\]  

(2)

If we let \( x_1 = a + \Delta x_1 \), \( x_2 = b + \Delta x_2 \) and \( \Delta y = y(x_1,x_2) - y(a,b) \), and if we truncate the infinite series such that we remove second and higher order derivatives, and if we understand that all derivatives are to be evaluated at the initial point \( (a,b) \), then we can simplify Eq.(2) to be:

\[
\Delta y = \Delta x_1 \left. \frac{\partial y}{\partial x_1} \right|_{(a,b)} + \Delta x_2 \left. \frac{\partial y}{\partial x_2} \right|_{(a,b)}.
\]  

(3)

To represent the more complex networks in use here, now consider \( N \) inputs \( x_1, x_2, \ldots, x_N \). Each of these inputs is a sampled function of time. For example, \( x_{ij}(t) = x_{i0} + j\Delta t \), where the first subscript identifies which input time history is being used and the second subscript identifies which value in the time history is being used. Then, we have \( \Delta x_i = x_{ij} - x_{i,j-1} \) and \( \Delta y = y_j - y_{j-1} \). To ease the notation, we will understand that each partial derivative is to be evaluated at \( (x_{i,j}, x_{i,j+1}, \ldots, x_{i,N}) \). Eq.(3) can then be written as:
where we interpret $\Delta y_i = \Delta x_i \frac{\partial y}{\partial x_i}$ as the contribution from input $i$ to the change in the output from time step $j-1$ to time step $j$. Now we have what we are looking for; namely, a method that can identify the unique contribution that a specific input makes to the change in the output. We can implement this method if we have a way to compute the partial derivatives of a feedforward neural network at each time step.

### 4.2. Partial Derivatives

*Hess et al. (2006)* derived the analytic equations for the partial derivatives. Reviewing here briefly, Fig. 5 shows a feedforward network with an input layer, two hidden layers and an output layer. Each layer is fully connected to the preceding layer, and nodes in the hidden and output layers use zero-to-one sigmoid nonlinearities. The right side of the figure shows how the input vector and the bias travel along weighted links from the input layer to the $j^{th}$ node in the first hidden layer. (Note that we have used index $j$ to refer to a time step and to nodes in hidden layer 1. This should cause no difficulty if understood in context.) Products of the inputs and the weights are summed with the bias to provide the input to the activation function, and the output from the nonlinearity is also shown. The process is similar for other nodes in each of the other layers; the minor changes in the notation are documented below.

#### General Architecture

![General Architecture](image)

**Input to Hidden Layer 1**

![Input to Hidden Layer 1](image)

Fig. 5: Architecture of a network with two hidden layers, and a view of the inputs to and output from a node in the first hidden layer.

The equations describing the transformation of the input vector into the output vector are given below. The notation used considers $i = 1, \ldots, N$ input nodes, $j = 1, \ldots, N1$ and $k = 1, \ldots, N2$ nodes in the two hidden layers and $l = 1, \ldots, No$ output nodes. (Note that there is only one output for each of the NNs used here.)

**Input layer to first hidden layer**

\[
v_{1j} = b_{1j} + \sum_{i=1}^{N} w_{1ij} x_i \quad (j = 1, \ldots, N1)
\]

\[
y_{1j} = \frac{1}{1+e^{-v_{1j}}}
\]

**First hidden layer to second hidden layer**

\[
v_{2k} = b_{2k} + \sum_{j=1}^{N1} w_{2kj} y_{1j} \quad (k = 1, \ldots, N2)
\]

\[
y_{2k} = \frac{1}{1+e^{-v_{2k}}}
\]
Second hidden layer to output layer

\[ v_i = b_i + \sum_{j=1}^{N_k} w_{ij} y_j \quad (i = 1, \ldots, N_o) \]

\[ y_i = \frac{1}{1 + e^{-v_i}} \] (7)

Substituting Eqs.(5) into Eqs.(6) and then Eqs.(6) into Eqs.(7) will yield a set of equations relating \( y_i \) to \( x_i \). Now, the set of partial derivatives that are required are \( \frac{\partial y_i}{\partial x_i} \) and represent the rate of change of the \( i^{th} \) node in the output vector with respect to the nodes in the input vector, as follows:

\[
\frac{\partial y_i}{\partial x_i} = \sum_{j=1}^{N_k} \frac{\partial y_i}{\partial y_j} \frac{\partial y_j}{\partial x_i} = \sum_{j=1}^{N_k} \left[ w_{ij} y_j (1 - y_j) \right] \sum_{l=1}^{N_l} \left[ w_{jl} y_l (1 - y_l) \right] . \] (8)

Examination of Eqs.(8) reveals that the derivatives depend upon the input vector at the current time step, as well as the fixed weights and biases; namely,

\[
\frac{\partial y_i}{\partial x_i} = f_i (x_i, w_{1i}, w_{2i}, b1_i, b2_i) . \] (9)

For a specified input vector and weight set, a computer subroutine implementing the feedforward equations, Eqs.(5)-(7) can be easily modified to also compute \( \frac{\partial y_i}{\partial x_i} \).

4.3. Computations

The time histories of the products \( \Delta y_i = \Delta x_i \frac{\partial y_i}{\partial x_i} \) can be computed for each of the inputs. For example, Fig. 6 shows the products for the same variables and same maneuver as given in Fig. 3.

![Fig. 6: Selected input contributions to the output \( y = \Delta \rho \) for one maneuver;](image-url)
Clearly, the products that represent the contributions of the variables \( p, q, u \) and \( v \) to the change in the output at each time step show the same tendencies as the original variables. Namely, the top two graphs show destabilizing contributions and the bottom two contributions remain mostly stable.

Before continuing further, we need to verify that the truncated series in Eq.(4) represents a good approximation, and that indeed higher order terms are negligible. In Fig. 7, \( \Delta y = y_j - y_{j-1} \) is plotted as the bold curve, and \( \sum_{i=1}^{\infty} \Delta x_i \partial y / \partial x_i \) is plotted as the thin curve. In this figure, the time axis was greatly expanded to show only 50 s of the time history in an attempt to show some differences between the two curves.

![Fig. 7: Agreement for truncated series approximation](image)

The curves are so close in agreement, that differences are not detectable. This agreement ensures that higher order terms are negligible and that all of the input contributions to the output are accounted for. This also serves as a check that the derivatives were computed correctly. Note that an additional independent check on the derivatives can also be performed using finite difference calculations as described in Hess et al. (2006). This check was performed and was also in close agreement.

5. Results and Discussion

Now that the \( \Delta y_j = \Delta x_i \partial y / \partial x_i \) products are available for inspection and analysis, we will proceed in a two-step approach. First, we will perform an analysis on the NN and NN+MCSIM data for the same maneuver for which selected inputs and outputs were shown in Figs. 3 and 4. The intent is to see if this analytical approach can identify the same inputs for removal as those that were chosen using the graphical method and listed in Table I for each NN. After this verification step, we will then work with the NN data only; that is, the data available when the NNs were trained offline and without the prior knowledge of how the NNs will perform when placed within the computational loop. The key here will be to perturb selected inputs (those most likely to contain the largest errors) of the trained NN and evaluate their destabilizing effect on the solution by means of the products.

The products, one for each input and examples of which are shown in Fig. 6, are time series. Therefore, we can compute the absolute value of the average and the standard deviation for each of these time series. The relative magnitudes can then be ranked on a column chart. This has been done for both the NN solution and the NN+MCSIM data for the same maneuver as used in the previous figures. The results are given in Fig. 8 for the \( \Delta \hat{p} \) NN and in Fig. 9 for the \( \Delta \hat{q} \) NN.
Ranking |Avg| of the products from largest to smallest gives a representation of those inputs, which give, on average, the largest contributions to the solution and are therefore most important; also shown are those which are least important to the solution. For example, for the \( \Delta p \) NN, the data in Fig. 8 show that inputs 20, 24, 4 and 5 are very important. However, notice what happens to the behavior of the NN when placed into the NN+MCSIM computation loop in the presence of additional error injected into the inputs. Now, instability in the contributions of certain inputs to the output is large enough, on average, that they become the most important inputs and drive the previously most important inputs further down the list. This is similar for the \( \Delta q \) NN shown in Fig. 9 as well. For ease of identification, those inputs that were removed by the graphical method are indicated with bold columns.

![Fig. 8: Ranking of products by |Avg| and Std. Dev. for the \( \Delta p \) NN](image)

**Bold:** Inputs removed by graphical method (see Table I)

Fig. 8: Ranking of products by |Avg| and Std. Dev. for the \( \Delta p \) NN

Instability also appears to cause large oscillations in the time series as shown in Figs. 3, 4 and 6. One might reasonably assume that those contributions to the solution that are dominated by instability might well have larger standard deviations. Clearly this is also seen in Figs. 8 and 9.
The $\text{Avg}$ and standard deviation are the first two moments of the underlying probability density functions for the products. As such, these statistics are obvious choices to use as discriminators to try to identify the same inputs for removal as those that were chosen using the graphical method. However, in this regard, they are somewhat lacking. While some of the inputs are correctly identified, others appear further down the list and would not be chosen without prior knowledge. Another potential possibility for a measure of oscillation due to instability would be to evaluate the kurtosis which quantifies the relative importance of data in the tails of the distribution. This was not performed; instead, a different approach was taken.

When one views the curves in Fig. 3, what is it that one sees that confirms the presence of instability, or not? The answer is that one detects dramatic differences in the shapes of the curves between the NN and the NN+MCSIM cases. On the other hand, other inputs, which remain stable in the presence of error, do not display these large differences. A quantitative measure which evaluates the similarity in the shape of two curves is the correlation coefficient, $R$. Given two time series $x(t)$ and $y(t)$, $R$ is defined as follows:

$$R = \frac{\sum (x - \bar{x})(y - \bar{y})}{\sqrt{\sum (x - \bar{x})^2 \sum (y - \bar{y})^2}}.$$  (10)

Therefore, for each $\Delta y = \Delta x \frac{\partial y}{\partial x}$ product, representing the relative contribution of a given input to the change in the output, one may compute $R$ to compare the shapes of the products for the NN data versus the NN+MCSIM data. The reasoning is that the additional error present in the NN+MCSIM loop will enter the inputs to the NN, propagate through the feedforward equations and emerge as contributions to the output. The destabilizing contributions will appear to be dramatically different in shape from those that were present in the NN when not in the computational loop. One may then rank the correlation coefficients for each product from smallest to largest and thereby quantify those that appear to be the most different. The results are given in Fig. 10 for both the $\Delta \dot{p}$ NN and the $\Delta \dot{q}$ NN.

As one can see from Fig. 10, this is the Eureka moment. Every one of the inputs removed by the previous graphical approach has been identified to have a low correlation coefficient for the products (contributions to the output). Indeed, returning to look at inputs 24 ($\alpha = \tan^{-1}(w/u)$, angle of attack of the hull to the velocity vector) and 29 ($\alpha^+$) for the $\Delta \dot{q}$ NN, one could argue that the error creeping into $w$ might recommend the removal of these two inputs as well. (However, since these variables did not contain the variables with the largest error, $p$ and $q$, and since the approach was designed to be conservative, removing as few inputs as possible, these inputs were retained. The fact that they did not, after the network was retrained, force the hybrid solution to be unstable in the presence of error, validated the decision.) We now have an analytical method which provides a strong discriminator to discover inputs that are likely to cause instability in the hybrid solution.

The next step is to work with the NN data only; that is, the data available when the NNs were trained offline and without the prior knowledge of how the NNs will perform when placed within the
computational loop. We will perturb two inputs, \( p \) and \( q \), of the trained NN (with all other inputs remaining the same) and then evaluate their destabilizing effect on the solution by performing correlations of the products before and after the perturbation.

What should the perturbation look like? We chose to input a saw-tooth perturbation beginning at time step 300, which is the point at which all of the controls had reached their final values for this maneuver. The before and after \( p \) and \( q \) time series are shown in Fig. 11. Although the sizes of the saw-tooth perturbations appear to be large, they were chosen so that the standard deviations of the perturbed signals, \( p: 1.163 \) and \( q: 0.942 \), were close to the standard deviations for these signals in the NN+MCSIM case, \( p: 1.120 \) and \( q: 0.908 \). Compare the graphs in Fig. 11 to the top graphs in Fig. 3.

The perturbed signals were input to the NNs and a correlation coefficient was computed for each \( \Delta y_i = \Delta x_j \frac{\partial y}{\partial x} \) product time series, comparing the behavior before and after the perturbation. Then, the correlation coefficients for each product were ranked from smallest to largest. The results are given in Fig. 12 for both the \( \Delta p \) NN and the \( \Delta q \) NN.

Injecting perturbed signals into \( p \) and \( q \) and leaving all other inputs the same, is not the same scenario as the NN+MCSIM case, where some of the other inputs besides \( p \) and \( q \) contained lower levels of error as well. Furthermore, the relative shapes of the altered \( p \) and \( q \) signals were purposely not made the same as the NN+MCSIM case, although the standard deviations were kept the same. Nevertheless, many of the products with the lowest correlation, indicating removal, are the same as those removed by the graphical method. Clearly, this analytical approach will identify many of the same inputs as those found by the graphical method, but this approach can be automated making the development of stable NNs for use in Synergistic Computing more straightforward and efficient.

A second case was performed using smaller saw-tooth variations in \( p \) and \( q \). For this test, the standard deviations of the perturbed signals, \( p: 0.557 \) and \( q: 0.389 \), were about 50% and 43%, respectively, of the standard deviations for the NN+MCSIM case, \( p: 1.120 \) and \( q: 0.908 \). The idea was to verify that the performance of the method did not rely entirely on the amplitudes of the perturbations. The perturbed signals for this case are given in Fig. 13 and the rankings of the correlation coefficients of the products are given in Fig. 14.
Comparing Figs. 12 and 14, one can see that, although there are some differences, most of the same inputs are identified.

5. Conclusions

This paper has given a brief overview of Synergistic Computing, which is the combination of CFD computations with NNs, to significantly improve the accuracy of maneuvering predictions from the hybrid computing tool. The overview given here described an application where the US Navy’s premier submarine 6-dof maneuvering prediction code, MCSIM, was combined with three feedforward neural networks, which were trained to compute residuals in the angular accelerations, $\Delta \dot{\phi}$, $\Delta \dot{\theta}$, and $\Delta \dot{\psi}$, representing corrections to yield better agreement between prediction and experiment. Although not shown here, Faller et al. (2010b) demonstrate that the NNs can accurately predict the moment residuals, and that this tool can provide accurate predictions over a varied array of difficult free-running submarine model maneuvers such as horizontal overshoots, vertical overshoots, turns and constant heading maneuvers. More importantly, the concept of using NNs to provide adaptive corrections to codes in a synergistic arrangement is feasible, and a new class of computing tools can be developed using this approach.

The primary intent of this paper has been to study the stability problem associated with the I/O interface between the codes. This is a complicated problem. To make any headway, a slow, but successful, graphical approach was initially used that also provided substantial insight into the problem. However, to be more efficient and to create an automated process for development of stable neural networks for use in the hybrid computing loop, an analytical approach was developed. The approach was based on computing the $\Delta y_i = \Delta x_i \frac{\partial y}{\partial x_i}$ products associated with the Taylor series expansion of the change in the output. These products are interpreted as the contributions from input $i$ to the output change from time step $j-I$ to time step $j$. In other words, we have a precise tool to study how destabilizing inputs enter into the NNs, propagate through the complex feedforward equations of multilayer networks and emerge to affect the output predictions at each time step.

The method was performed on the NN and NN+MCSIM data to see if it could identify the same inputs for removal as those that were chosen using the graphical method. Ranking correlation
coefficients of the products before and after insertion into the computational loop proved to be the needed discriminator to allow the method to be successful. Then, the method was applied to NN data alone, where selected inputs were perturbed to study their destabilizing effect on the solution. Different perturbation levels were tested, yet the majority of the inputs identified for removal remained the same.

Clearly, more work needs to be done. One aspect of the problem is how to characterize the nature of the errors that one anticipates will be coupled into the NNs in a given hybrid computing situation. Then, how does one mimic this situation effectively by injecting perturbations into appropriate inputs? Despite these lingering questions, the results of this initial work indicate that this method should be robust enough to reduce the time required for development and evaluation of the NNs used in a Synergistic Computing process.

Acknowledgments

This work is supported by the U.S. Office of Naval Research through Code 331. The program officer for the Synergistic Computing development is Dr. Ronald Joslin. In addition, program officers Dr. Ki-Han Kim and Dr. Patrick Purtell, also from Code 331, have also provided funding for neural network research. The automated feedforward neural network development code, Intelligent Calculation of Equations (ICE), was used to create the neural networks for the Synergistic Computing effort. This code was developed by Applied Simulation Technologies and is available for free by contacting the company via faller@appliedsimulation.com.

References


The Employment of 3d Collaborative Environment from the Very Beginning of the Conceptual Warship Design Stage

Carlos V. M. Santos, Brazilian Navy, Rio de Janeiro/Brazil, malheiros@cpn.mar.mil.br

Abstract

In this paper, a methodology of how to introduce popular low-cost CAD tools in ship design from the very beginning of the conceptual ship design phase is proposed. Considering the need of reducing the global design time and improving the overall quality of the job, reuse of design data is a must. From this consideration, it is proposed to link the FORAN system from the conceptual ship design to the accommodation layout via module FDESIGN. This methodology is applied to a small Ocean Patrol Vessel (OPV) and presented here.

1. Brief History of Warship Basic Design

The design of a naval warship is one the most complex tasks with which an engineer can be involved. A new warship design is a very complex combination of various engineering design activities. It is necessary to provide an integrated ship platform with its systems and moreover, to assure that the ship platform is perfectly integrated with the combat and weapons systems fulfilling their requirements. In order to provide a well-integrated and high quality warship design, it is mandatory to start the concern of good design management since the very beginning of the initial design stage, which means, the conceptual design. The following picture locates such design stage in the design lifeline.

![Fig. 1: Ship Design Lifeline](image)

The hard task of people coordination associated to engineering working under an ever-changing design environment leads design managers involve few people in the conceptual ship design stage. Sequential engineering is the most common practice at the conventional initial design stage. The design process is very dynamic and changes occur intensively being very hard to control and propagate them timely to the other design team members. Due to this fact, many design inconsistencies may be observed and some design cycles are necessary to refine the design eliminating such inconsistencies.

In general, the supporting design tools used at this design stage are not the same tools used during the detail design phase, mainly, due to the complexity of the usage; the high level training required and the little flexibility for accommodating big changes in a short period of time. Tools oriented to support the design in advanced stages are focused on production activities and they are not optimized to accommodate the ever-changing environment peculiar to the basic design stage.

Nowadays, the high competition of the global engineering market has increased more and more the demand for specialized people. To get and to keep specialized people has become a hard task. Such scenery aggravates the problem of investing on training people and to keep them in your company using sophisticated engineering tools.
In order to exemplify a problem that occurs when practicing sequential engineering instead of practicing concurrent engineering at the basic design stage, let us mention the interaction between the designer of the exhaust gas system and the naval architect in charge of the general arrangement, especially when they are discussing the funnel arrangement from the machinery room space.

Before starting the design of the exhaust gas system, the designer receives from the Naval Architect the current status of the general arrangement and at that moment he gets the information on the available space inside the chimney and its position in order to accommodate the exhaust gas ducts. So, the designer starts the calculation of ducts diameters. The preliminary ducts route is arranged but, in this meanwhile, the Naval Architect changes the position and the geometry of the chimney. In a sequential engineering environment such modification is not updated in a real time, which means the ducts designer will need to revise his work at the next cycle of the design. Such kind of problem happens frequently with several other systems interacting with the general arrangement. It is very complicated to keep people updated on the general arrangement. Due to this fact many naval architects choose practicing sequential engineering, i.e., only starting the systems design after having a stable general arrangement. Such practice is not efficient and time consuming.

### 2. Implementation of Concurrent Engineering Environment in Basic Design Stage

In order to optimize and to accelerate the design tasks during the initial design stage, one possible solution is to implement the concurrent engineering environment. As it was said before, customized tools optimized for working in a concurrent engineering environment are, in general, complex to use and they demand very specialized and well-trained technicians. Moreover, such tools are costly in comparison with traditional CAD tools.

At this section, it is proposed a methodology of design organization oriented to the concurrent engineering environment. Many different CAD tools are suitable for such methodology, but, at this paper, it is presented a methodology based upon AutoCAD powered by Autodesk.

The methodology consists of:

1. to divide the ship design documentation according to SWBS organization, separating ship design parts by folders and files adequately;
2. to link different parts into a single file by means of external references;
3. to control the access and changes of different parts by denying/granting the edition permissions of files.

#### 2.1. The SWBS design organization

The SWBS organization (Ship Work Breakdown Structure) was proposed and defined by NAVSEA (Naval Sea Systems Command) - US NAVY according to NAVSEA 0900-LP-039-9010, as used in the Naval Ships Technical Manual (NSTM). Under the SWBS organization the work of designing and building a ship is classified into ten groups based upon the ship system in which each part is located according to the engineering disciplines, i.e., structure, main machinery, electricity, weapons, auxiliary machinery, accommodation, etc. Table I shows the main SWBS subdivision.

<table>
<thead>
<tr>
<th>SWBS Number</th>
<th>SWBS Title</th>
<th>SWBS Number</th>
<th>SWBS Title</th>
</tr>
</thead>
<tbody>
<tr>
<td>000-099</td>
<td>General Guidance and Administration</td>
<td>500-599</td>
<td>Auxiliary Systems</td>
</tr>
<tr>
<td>100-199</td>
<td>Hull Structure</td>
<td>600-699</td>
<td>Outfit and Furnishings</td>
</tr>
<tr>
<td>200-299</td>
<td>Propulsion Plant</td>
<td>700-799</td>
<td>Armament</td>
</tr>
<tr>
<td>300-399</td>
<td>Electrical Plant</td>
<td>800-899</td>
<td>Integration/Engineering</td>
</tr>
<tr>
<td>400-499</td>
<td>Command and Surveillance</td>
<td>900-999</td>
<td>Ship Assembly and Support Services</td>
</tr>
</tbody>
</table>
Following such organization, creating folders according to the SWBS organization is proposed, Fig. 2. After creating the SWBS folders organization, we have to create files containing design parts following some criteria. The general arrangement, e.g., is composed by some plan views, each one related to a ship deck. Fig. 3 shows a typical general arrangement drawing. The criteria for splitting the general arrangement in multiple files consider that each deck is related to a separated group. Fig. 4 shows the initial General Arrangement main file and the files related to the deck levels separately. The files are placed in the appropriated folder according to SWBS organization. Each deck group is divided in logical elements, as follows:

- furnishings related to the deck;
- internal ship sub-division on the deck;
- deck contour with watertight bulkheads.
Figs. 5 and 6 illustrate the logical structure of a file containing a deck for the main deck composition.

Fig. 5: Reference files inside the main deck file

Fig. 6: Main deck drawing composition
2.2. Link between design parts and external references files

Once folders and files are created, the next step is the relationship among the existing files by means of some "key files". Such files are, in fact, drawing arrangements inter-connected that present all internal ship sub-division and organization. The following items list some key files.

- General arrangement
- Machinery rooms arrangement
- Steering gear room arrangement
- Bridge arrangement

In addition to these key files, other important files can be created and related. Such files are associated to ship systems integration. Below, it is presented some ship systems integration files.

- HVAC System schematic drawing
- Lighting system schematic drawing
- Fire-fighting system schematic drawing

Fig. 7 shows the integrated HVAC diagram on the ship main deck.

In a ship design, drawing arrangements and system schematic drawings are strongly related. It is very important to keep the coherence among them. The dimensioning of the HVAC system is strongly related to the deck subdivision as shown in the previous picture. The internal arrangement will define duct lengths, the amount of airflow terminals and their positions. Such parameters affect the system dimensioning and any change in the ship sub-division may result in a HVAC system reviewing. Many times when such arrangement files are not linked, it is very common to follow the design without updating important systems. Only at the end of certain design stages the design inconsistencies are perceived and treated. Many times the process of correction is iterative, what requires mutual adjustments between the general arrangement and the HVAC schematic drawing. As a result, we have waste of time and a low quality design.

Advantages of working with linked files in shared way are:

- The common geometric information is shared for different purposes inside the design environment; e.g.: deck contours and subdivision may be used for both for schematic system diagram and for area arrangement purposes, Fig. 8.
- All changes in the file are notified immediately and they can be updated without the need of reopening the file. A simple click on the command reload in the small pop-up window refreshes the visualization of the changes inside the drawing, Fig. 9;
- All design members have the updated information on current status of the general arrangement drawing. Multiple accesses are allowed to the same file in the edition mode since people work on different tasks or areas, Fig. 10;
The design integrity is assured by file edition protection, granting/denying edition permission according people responsibilities in the design context.

Fig. 8: HVAC diagram integrated with the general arrangement

Fig. 9: Popup window warning online changes

2.3. Changes and access control policy

Once separated files are created and the links among them are established, the next step is to implement the policy of control on the changes and access. Such policy is based upon Windows explorer security tools related to files and folders. The implementation of the policy of changes and control access consists on creating groups divided into engineering disciplines and associating people according their specialties.
After that, it is necessary to define the permissions related, folder-by-folder and file-by-file, Fig. 11. In addition to the designer groups defined according to engineering disciplines, it is important to define discipline managers with special privileges. In fact, a discipline manager is an engineer who coordinates the job of a group related to a specific discipline. The discipline manager is the engineer who verifies all documentation of his area and decides who is going to have edition privileges and which areas of the ship need to be frozen after certain stage of the work.
To freeze certain parts “files” of the design after revising and approving them is essential, because any non-authorized modification may compromise the coherency of the design. It is very important to filter who is authorized to edit each file and at this task the discipline manager has to take into account the individual competencies and previous experiences of each design member.

3. Transition between Basic and Detail Design Reusing Previous Information in FORAN

Up to now, we talked about the introduction of concurrent engineering environment in the basic design stage. In general, at very preliminary design stages we use to work in a 2D environment in order to assess the preliminary spaces allocation. Many ship designs involve high density of equipments onboard and, sometimes, complex hull geometry with very slope surfaces. Such features may become the task of positioning equipments and drawing the compartments layout a little bit complicated mainly if you are working in a 2D environment. In order to facilitate such task and to enhance the quality of the design it is mandatory to work in a 3D environment.
It is very important to keep the coherency between the 2D and 3D environments. How could this be done? The answer is to follow the same methodology of linking 2D files. The subdivision of the ship design into specific parts as in the previous section is very useful to promote the 3D link. Considering the general arrangement drawing previously presented, each view plant is related to a specific file. From a new file, named as 3D model, it is possible to insert deck plan views, one by one, by means of external references positioning them in the space, i.e., using the relative vertical positioning. Fig. 13 shows the resultant file.

Once we have the 3D layout of the deck plan views, the next step is positioning furnishings, outfitting and pieces of equipment inside the 3D layout. We have to keep in mind the goal of having all the design parts linked and integrated. Thus, each 2D symbolic representation for equipments, furnishings, etc, must be defined as smart blocks. Smart block is a group of geometric entities where the symbolic 2D representation is associated with one layer so called 2D representation and the 3D representation is associated with another layer so called 3D. The geometric relationship between the symbolic representation and the 3D one needs to be adjusted, that means the 2D plan projection of the 3D model needs to fit inside of the 2D symbolic representation using the same origin reference. Fig. 14 exemplifies this situation.

After establishing the link 2D-3D considering the general arrangement, it is possible to control the 3D layout from the 2D environment that means each movement in the 2D layout represents an associated movement in the 3D layout. Since we have the 3D layout available and after finishing the interactions among the engineering disciplines, an almost finished configuration is available. The next step is to introduce such information in a more robust CAD/CAM system aiming the detail design phase and production interactions. How could this be done? In order to try answering such a question we will exemplify such transition applying the FORAN system.

The FORAN system has the module FDESIGN that is an interface module dedicated to provide output drawings in different formats. So, in order to introduce the geometric information inside the FORAN environment the idea is to perform the inverse sense, i.e., to read an external file bringing relevant information and geometric references inwards the design database, Fig. 15.
4. Practical Results from the Design Methodology Applied to an Ocean Patrol Vessel Design

Among some practical results from the design methodology presented in this paper applied on the OPV design, we can highlight:

- The 2D General Arrangement as a single source of geometric information, (Ship internal sub-division), shared with system diagrams, Fig. 16;
- The 2D General Arrangement integrated with the 3D model, Figs. 14 and 17;
- The resulting 3D model as geometric reference to start the detail design into the FORAN system, Fig. 18 and 19
Fig. 17: 2d Resulting 3d model (intermediate)

Fig. 18: Integration of 2d arrangement in 3d model

Fig. 19: Resulting 3d model (intermediate)
5. Conclusion

The introduction of the methodology of producing the 3D General Arrangement Design from the conventional 2D layout inside a context of concurrent engineering has the purpose of reducing the probability of having to do the work more than once, reducing the access to outdated information or even wrong information. As a result of the employment of such methodology it is expected to improve the overall quality of the ship design since the very beginning of its initial stages. The implementation of the link between conventional CAD tools used in the preliminary design stages and the FORAN System represents a real advantage of the proposed methodology considering the design data reuse in the following design stages.

References

ALLONSO, F.; GONZALEZ C.; PINTO A.; ZURDO, F. (2008), Integrated management of ship compartments. COMPIT, Liege


MILLER, L.K. (2009), Competitive advantage in new products through simulation-based engineering, DataMetric Innovations.

NAVSEA (2008), Future Concepts and Surface Ship Design Tools Roadmap, Naval Sea Systems Command 05D.

NSRP (2008), Navy Product Data Initiative Integrated Product Data Environment (IPDE) Specification, NSRP ASE.


SULLIVAN, P. (2008), Ship design and analysis tool goals, NAVSEA Memorandum Ser 05D/047

TEPPER, G.; Jonge, T. De (2009). NUPAS-CADMATIC’s vision on CAD/CAM system development in today’s shipbuilding environment, COMPIT, Budapest
Abstract

This paper describes the necessary developments for a hull strength assessment during the lifecycle of a ship using 3D models for hull condition recording (HC model) and for simulating hull strength (FE model). Beside appropriate assessment methodologies of aging structures an effective data exchange between the HC model and FE model is needed. The difficulty here lies in the different abstraction levels of the 3D models. The first steps for transferring actual inspection results stored in the HC model to the FE model as well as the derivation of structural assessment components like buckling fields are presented.

1. Introduction

For new ship designs, the structural strength of the vessel is assessed frequently with the help of a finite element (FE) model in the design phase, for the purpose of design optimization or approval of design. The FE model is typically not used in later phases of the ship life cycle. For assessing the condition of a ship’s hull in operation phase, qualitative and quantitative methods are in use today. A visual coating inspection by an expert examining the structure and reporting the results would typically be a qualitative method, whereas the quantitative approach in most cases consists of periodical measurements of defined condition parameters such as the remaining steel thickness. Information is in general collected manually, e.g. by inspection contractors reading the measurement devices and reporting the results.

Whereas visual coating inspections can help in predicting future corrosion states, one has to rely on thickness measurements to re-assess the structural strength of a vessel. Performing inspections for hull condition monitoring and assessment as stipulated e.g. in IACS unified requirements and IMO regulations involves a huge amount of thickness measurement data to be collected, processed, analysed and maintained. Measurement campaigns can take several days and involve up to 30,000 measurement points. For an overview about current hull inspections for condition assessment see Paik et al. (2006).

Nevertheless, the standard thickness measurement procedure today involves planning and reporting of the measurements in the form of tables and associated 2D drawings and sketches. This makes evaluation of remaining strength with the help of an FE model difficult. For this (and other) reasons, the standard assessment methodology today involves the local comparison of measured thickness values against a minimal residual thickness values.

The use of a 3D model of the hull structure can obviously be beneficial for the thickness measurement procedure: planning of the measurements can be done without the need to gather drawings and resulting thickness measurement information can be directly related to the 3D model of the corresponding structural element. In this way, e.g. the software can find out if the thickness is still compliant with the minimal value approved by the classification society, Jaramillo et al. (2006). An example of such software is shown in Fig. 1.

If a 3D model for inspection purposes is available, it can furthermore be used to on the one hand report on visual inspection results, e.g. Cabos et al. (2010), or on the other hand to associate measurement values and other structural information with the FE model for strength assessment. The latter is the subject of this paper and –for various reasons described below– it is not common practice today.
The model which is used for inspection will be called Hull Condition Model (or HC Model, HCM) in the following. Note that the inspection and simulation processes have different requirements and therefore the 3D models have different properties and availability. Some examples are:

- HC models need to show the proper location of structural details in order to locate findings. FE models can however have a considerable level of abstraction
- Topology is central in an FE model, whereas not needed in the HC model
- Thickness readings are located at their actual location in the HC model and need to yield thickness distributions in the FE model

Because of these differences, if both models are available, they are typically built up independently. Alternatively they could be derived from one common CAD model, this is however not common practice. One reason is that CAD software support this has only become available recently, another reason is that the necessary CAD model is often only built up in a later design phase, Fig. 4.

2. Current assessment procedures

A number of different techniques for assessing ship structures in operation can be distinguished today. Apart from a purely visual assessment of coating condition and corrosion, we are focussing here on the assessment of strength based on thickness measurements. The scope of required thickness measurements is specified according to the respective survey regimes, i.e. Condition Assessment Scheme (CAS), Enhanced Survey Programme (ESP), Condition Assessment Programme (CAP), etc., see Paik et al. (2006) for an overview. The evaluation of the thickness measurements is on the other hand given by the classification rules applicable to the specific ship. Here, two approaches need to be distinguished.

The Common Structural Rules (CSR) for bulk carriers and tankers are based on the net thickness approach, IACS (2005b). The net thickness is the thickness “required solely based on the structural strength aspect which is the minimum scantling that must be kept throughout the service life of the ship”. The corrosion addition is then found by adding wastage allowances which depend on the type of compartment on both sides of a respective plate.

In the traditional approach, which is applied in most classification rule sets apart from the CSR, the dimensioning procedure directly yields the required newbuilding thickness. The minimal residual
thickness is reached typically when the as built thickness is reduced by a certain fixed percentage or a fixed amount. This approach is in general easier to apply, since it does not need additional information except the as built thickness (and the maximal diminution rule) during assessment.

Both types of rules are based on experience. E.g. the CSR corrosion additions have been derived from a statistical evaluation of a large number of measurements on aged ships. The next step towards a direct strength assessment of corroded structures is taken by the evaluation of the remaining hull girder section modulus. E.g. the CSR JBP require “The current hull girder section modulus determined with the thickness measurements is not to be less than 90% of the section modulus calculated according to Ch 5, Sec 1 with the gross offered thicknesses.”

Despite a number of publications on the subject of statistical evaluation of the hull girder sectional modulus of corroded ships or the time varying structural strength of corroded members have appeared in the literature, e.g. Guo et al. (2008), no publications on the subject of an FE analysis of corroded ship structures are known to the authors or have been reported in the last ISSC report.

Several problems need to be solved for a strength assessment of a corroded ship structure with the help of FE models:

1. The thickness (or thickness distribution) for each plate element needs to be defined based on the measurements of the corresponding plate. In particular the case of several differing measurements on one plate needs to be considered. An alternative could lie in the design of special finite element types for corroded structures.
2. Assumptions for the thickness of plates which have not been measured but are present in the FE model must be made.
3. A procedure of how to automatically transfer measurement data from the HC model to the FE model based on the algorithms defined in steps 1 and 2 needs to be developed.
4. For assessment of the FE analysis results (buckling, fatigue, …) additional information about structural details need to be gathered.
5. The thickness used in the buckling or fatigue post-processing needs to be defined. This can be handled differently from the procedure in 1. Again, special element types could be defined for this post-processing step.

This paper describes a procedure for items 3 and 4.

3. Available 3D Assessment Models

3.1. Finite Element Model (FE model)

During the design phase of a vessel an FE model serves as numeric assessment model for the strength of the ship structure, Fig.2. In general there is not a single FE model that is suitable to capture all strength relevant issues. Depending on the particular strength aspect, special finite element models might be created:

- Finite element models for determining the nominal stresses of the structure caused by real wave situations (global FE models). These models are characterized by a high level of abstraction. Plate contours are strongly idealized; in particular element edges do not match plate boundaries. Cut outs may be represented as reduced plate element thickness and stiffeners are summarised as truss or beam elements on element edges. Most design details are omitted or represented in a very simplified fashion.
- Finite element models determining the utilization of the cargo hold area of the structure caused by loads specified by classification societies (cargo hold FE models). The level of abstraction for such cargo hold FE models is moderate: The contour of a plate is modelled within the tolerance of the chosen mesh density (typically 700-1000 mm edge length), thus small overhangs of plates at junctions caused by production, Fig. 1, are just shifted to the
respective junction. Cut outs are represented as holes in the FE model but typically with a simplified contour. Stiffeners are modelled as one dimensional line elements coinciding with the real trace line within the mesh density. The connection of a stiffener to a primary (plate) member is realized by common node degrees of freedom at their junction instead of a detailed model of the real connection design.

- Finite element models for determining the fatigue strength of dedicated structural details like brackets, profile cut outs or hopper knuckles according to the Hot-Spot or Notch-Stress approach (fatigue FE models). These models will be typically realized as separate local FE models. Due to the applied fatigue approach these FE models exhibit only a very low level of idealization to compute the required local stress state as precisely as possible. Hence plate contours and geometric details like fillets are modelled close to reality.

![FE models created during the design phase. Details of a) a global FE model, b) Cargo hold FE model and c) FE model for fatigue assessment](image)

The above models are generated by the ship yard, a design office or the class society with different tools (generic FE pre-processors, CAD systems, in-house developments) based on diverse data sources (drawings, pure surface descriptions, 3D production data) mostly in an elaborate way. After the production process is finished these FE models are generally not further used but could also be applied for strength assessment purposes during the life cycle of the vessel.

Independently of the used FE pre processors and solvers all above mentioned FE models contain at least the essential data for solving the linear static FE problem: nodes, element topology, element properties and materials. Moreover, almost all FE systems supporting the arrangement of finite elements into sets allowing addressing of special model regions like dedicated structural members as for example the outer shell. The FE models and results are in general insufficient to provide all the information needed to assess all strength relevant issues, Wilken et al. (2008). Direct physical results from the FE solution like deformations and stresses can be easily assessed without any additional structural information. More sophisticated strength relevant items like buckling utilization or fatigue life however require structural information that was lost during the idealization process. Hence a
buckling assessment for instance can not be performed based purely on a global FE model since relevant model data like the real plate thickness or the exact stiffener distribution can not be retrieved. In practice this kind of data will be incorporated in different, non standardized ways in the further assessment procedure.

3.2. Hull Condition Model (HC model)

The most prominent, open and standardized ship hull condition data model came up within the European funded CAS project, Cabos et al. (2008), and is further developed by the OpenHCM consortium (see http://sourceforge.net/projects/openhcmstandard/). This hull condition data model (HCM) defines a data exchange format enabling the assignment of inspection results to a 3D structural model built up by plates and stiffeners, Fig.3. A plate will be modelled by a boundary polygon describing the contour of the plate and several assigned attributes like approved plate thickness or structural functionality. A stiffener is represented by its trace line defined as a polygon and its orientation by direction vectors at the vertices of the polygon. The profile shape is not described in detail but as an assembly of a rectangular web (defined by height and thickness) and a flange (described by width and thickness). Plates and stiffeners are modelled close to reality to assign measurements as precise as possible to the structure – for example to support special assessment procedures or to facilitate possible repair planning. Hence the degree of abstraction concerning the shape of plates and stiffeners of a hull condition model is much lower than for the above mentioned FE models. Similar to the element arrangement in finite element models, plates and stiffeners are uniquely assigned to a so called “member” group. Therewith different plates and stiffeners are arranged to superior structural members like a transversal bulkhead.

A design principle of HCM “is that it shall not be necessary to enter more information in preparation and conduction of measurements than today”, http://sourceforge.net/projects/openhcmstandard/. Thus any data not directly required for inspection purposes is not contained in the HCM. This facilitates the three-dimensional model creation – in particular the transfer of structural data from CAD systems into the HC model – but might cause problems for further model processing. Among other things, topological relations between plates and stiffeners are missing. Hence an automatic detection of more complex structural systems like a side shell frame as an assembly of plates and stiffeners is difficult.

Fig. 3: Detail of a HC model. Although very similar to an FE model at first sight, there are systematic differences.

The HCM supports in particular plate (and stiffener) thickness measurements as strength relevant items. In doing so, thickness measurements are represented as Cartesian points assigned to a plate or a stiffener with an attached measured thickness value. Present structural assessments are based on the
comparison of the measured thickness with fixed permissible thickness defined by class or IACS regulations. The direct impact of plate or stiffeners diminution on the local or global strength behaviour of the structure can not directly be derived from the HC model since it is no simulation model. Moreover, the topology of the model, i.e. the connection of plates and stiffeners defining the distribution of forces, is missing but essential for strength assessment.

3.3. Strength Assessment based on the Combination of both Models

If one wants to assess the actual hull condition of a ship structure based on inspection results and using existing FE models the following steps have to be performed:

1. Transfer of inspection results, i.e. measured thicknesses to the FE model(s)
2. Computation of structural deformation of the updated FE model(s)
3. Provision of additional information required for the particular strength assessment items, e.g. buckling field geometry for buckling checks or structural details for a fatigue check
4. Assessment of the mentioned strength relevant items using the structural deformations from 2 and the additional information from 3 into account.

As described above neither the existing FE model(s) nor the HC model is suitable to perform such an assessment alone. The obvious solution is to use both the FE model(s) and the HC model: The HC model provides actual plate thicknesses as well as detailed structural information; the FE model provides load-induced structural deformations. The combination of both models can be used to retrieve the actual local or global (depending on the type of FE model) structural strength. For that purpose a coupling between HC model and FE model is required. Typically, HC model and FE model are independently created so that there is no direct relationship between these models: A plate within the HC model has no direct link to an element within the FE model and vice versa. Due to the complexity of the structure this link can only be established manually to a limited extent. Hence it must be built up automatically as far as possible.

![Life-cycle phases of a ship and 3D models built up in these phases.](image)

Fig. 4: Life-cycle phases of a ship and 3D models built up in these phases.

4. Assessments Methods using FE analysis

To describe the methodology for strength assessment based on HC Model and an FE Model, we are assuming in the following that a global FE model is available from design phase.

4.1. Preparing the FE model

When starting the first inspection-based assessment, a FE model of the structure might already be available. However that model was usually prepared for analysing the initial design. Therefore, it will typically not reflect the actual as-built state of the ship. For example:

- The original FE analysis revealed certain issues that were resolved later for the final design.
• During plan approval, the classification society asked to increase certain scantlings.
• The shipyard further increased other scantlings with respect to the required scantlings.

All the above changes are conservative with respect to the initial FE model. Thus, the final design is typically not re-analysed by means of the FE method again, Fig. 4. The later design changes were not maintained and are not accounted for in the last available version of the FE model. Thus, information about the as-built scantling should be gathered. The FE model should be updated accordingly. Otherwise, subsequent analyses could not take into account the increased as-built scantlings.

The FE model should also be checked for alignment with the HC model since they represent different abstractions of the ship hull. The models also use different concepts to represent faces. E.g. HC plates are represented by a boundary polygon while plane-stress-elements are represented by bilinearly distorted quads. Thus, the elements of both models cannot always geometrically coincide for principal reasons. When HC and FE entities are matched by means of distance criteria, a search tolerance value needs to account for that.

However, there may also be some rectifiable mismatches. E.g., the interpretation of moulded line offset might be different. Although such minor geometric deviations are negligible for FE analysis, fixing such misalignments allows for using smaller search tolerance values. This, in turn, might improve the quality of HC-to-FE data transfer. Further, misalignments might indicate HC or FE model errors that can be fixed but would have been undetected without an alignment check.

4.2. Using measured thickness values

For each plate, there are several associated thickness values, Fig. 5:

• The as-built (from the shipyards perspective) thickness
• The thickness as approved by the classification society
• Renewal thickness (minimum thickness tolerated after an inspection)
• Several measured thicknesses at different locations of the same plate

In principle, the HC data model provides for representing all above thickness types. However, it cannot reasonably be expected that all related data is available for each plate.

![Diagram of plate thickness values](image)

Fig. 5: Plate associated thickness values

Measured thickness values will not be available for all plates because the set of measured plates increases with the age of the ship. The first thickness measurement campaign is restricted to a small set of plates. At the end of design life, thickness measurements are required and will be available for almost all plates.
The type of stored initial plate thicknesses will typically depend on the creator of the HC data base. If it is provided by the shipyard, based on their manufacturing data model, as-built thickness values are likely included. Other thickness types are not needed for manufacturing purpose. Thus, approved or renewal thickness values will likely be missing within such HC models. For another scenario, where the model is provided by the classification society, it will include the approved thickness values. Those had been passed to the society for plan approval, but the society does not know about later yard additions. Thus, the model will not contain final as-built thickness values. However, the approved thickness values are a conservative lower bound for the as-built thickness.

The thickness values applied in the calculations should reflect the actual state of the ship as precisely as possible. In principle, these are the most recent measured thicknesses. In practice, as already mentioned above,

- measured thickness values are not necessarily available for all plates and
- for each measured plate, there will be several potentially different measured thickness values.

Thus, a strategy for selecting appropriate ‘model thickness’ values must be developed.

Our assessment procedure is clearly separated into two subtasks:

1. Some (global) design loads are applied to a global FE model of the ship. As a result, we obtain some local structural responses like deformations or nominal stresses. Those responses will depend on thickness values of (in principle) every finite element.
2. The local responses obtained from the FEM calculation are used as loads for a local assessment model, e.g. related to buckling of an elementary plate panel. The buckling capacity will depend on the elementary plate panel’s thickness value.

Both subtasks relate to different models. Those models serve different purposes. Therefore, different strategies to select appropriate model thickness values are reasonable. The local buckling assessment procedure strongly depends on the thickness of the elementary plate panel. It does not directly depend on other (global) plate thicknesses. Only the local loads need to be computed realistically. The larger the distance of finite elements to the elementary plate panel, the lower the effect of their thickness values on local load. Thus, if no measured thickness values are available for certain plates, we simply use the as-built thickness value for the FE model. If several different measured thickness values relate to a plate, the effect on local loads will be modelled most appropriately by using the average of the measured thickness values for the FE model. The situation is different for local buckling assessment. As long as the local buckling model assumes a single thickness value for the whole elementary plate panel, we conservatively use the plate’s smallest measured thickness value for local buckling assessment. In principle, it would be possible to use buckling assessment methods that account for local thickness variations. However, was left for further study. We do not need to handle plates without measured thickness values. For such plates, measurement-based buckling assessment is not performed, anyway.

5. Transfer of data between different models

5.1. Transfer of inspection results to the analysis model

Intuitively, the process of transferring a thickness value from an HC panel to associated finite elements is very clear. In practise, it turns out to be a rather complex task. One source of complexity is the difference in purpose and abstraction. The HC model aims at providing a data exchange format for inspection results. The FE model aims at simulating load-induced structural deformations. Although both models provide for representing the structure’s geometry, this is not their principal purpose. In a different manner, both geometry representations will likely differ from the real geometry to a certain degree.
Even if both geometry representations coincided exactly, the corresponding element grids would not match. Usually, there are plates that extend about several finite elements. And there are finite elements that overlap with several plates.

Fig. 6: Non-matching and slightly misaligned HC and FE meshes

Associating HC plates to finite elements and mapping related thickness values resembles a problem occurring in fluid-structure-interaction simulations with non-matching grids. The latter requires transferring pressure loads from a hydrodynamic mesh to a finite element mesh. Thus, the problem of thickness mapping can be attacked by similar algorithms. Therefore, our thickness transfer method was based on a pressure mapping algorithm that was already successfully implemented for the GL ShipLoad program, Eisen and Cabos (2007). Some modifications were however necessary and robustness requirements turned out to be more demanding than for the fluid-structure-interaction scenario. This resulted in the following principle algorithmic steps:

1. For each HC plate, the closest finite element is determined.
2. The finite elements adjacent to the initially matched finite element are investigated. If those elements overlap with the HC plates, their neighbour elements are also investigated for overlap, recursively.
3. The intersection area of the HC plate and each related overlapping finite element is computed. A weight factor (finite element area)/(intersection area) is computed for all overlapping elements.
4. The thickness of each finite element is set to the weighted sum of the overlapping HC plates’ thicknesses.

Searching the initial closest finite element of an HC plate started with determining a reference point and normal direction for the plate. In the HC model, plate representations are neither explicitly equipped with reference locations nor normal directions. Thus, they need to be computed from the plate’s boundary polygon. This was achieved by computing a simple triangulation of the plate. The triangulation’s centre of gravity was chosen as reference point. A weighted sum of the triangles’ normal directions was chosen as normal direction. The applied weight factors were simply the triangles’ area.

After reference locations and normal directions are determined, searching the closest finite elements continues as in the fluid-structure-interaction case: For each HC plate, there is a line running along the plate’s normal direction and containing the plate’s reference point. All finite elements are checked for intersection with that line. The intersected element with the smallest distance (in normal direction) to the reference point is finally selected.
Identifying the finite elements that overlap with the HC plate is performed by the fluid-structure-interaction algorithm. The HC plate and the finite elements are projected into the HC plate’s reference plane, i.e., the plane orthogonal to its normal. A plate is considered to overlap with a finite element if the intersection of their projections is non-empty. The algorithm starts with the initially determined finite element that is closest to the plate. The edges of the finite element are projected into the normal plane. If the projected edge intersects with the projected plate, the finite elements connected to that edge are added to the set of overlapping elements and their edges are investigated for overlap and adjacent elements, recursively.

While an unconstrained version of that algorithm works well for the fluid-structure-interaction mapping, it sometimes ran into problems when applied to HC plate mapping. Problems occur when a finite element edge is shared by more than two elements. This may occur frequently, e.g. where the outer shell is connected to a deck, Fig. 7. By means of the above criterion, a finite element of the deck might formally overlap with the outer shell plate. For the fluid mesh mapping, only the elements of the outer shell are considered as mapping targets and the problem cannot occur. For HC mapping, there are source plates related to the outer shell as well as to the deck. Subsequently, the set of potential mapping targets must contain finite elements related to the outer shell as well as to the deck.

The problem was attacked by an additional angle criterion. If an overlapping finite element edge was shared by more than two elements, the angle between the HC plate and the new finite element normal vectors were computed. Only the element with the smallest angle was added to the set of overlapping elements, the other elements were explicitly removed.

In principle, the weight factors $w_{ij}$ for partially overlapping HC plates $p_i$ and finite elements $e_j$ are computed as in the fluid-structure mesh mapping algorithm. The boundaries of the plate and the elements are projected into the plate’s reference plane, forming 2-dimensional polygons $P_i$ and $E_j$. The weight factor computes as $w_{ij} := A(P_i \cap E_j) / A(E_j)$, where $A(\cdot)$ denotes the area of the polygons. In practice, fluid element faces are usually convex while HC plates are frequently non-convex. Thus, the implementation must be careful about computing intersection areas of non-convex polygons correctly. Once the mapping weights are computed as above, we could map thickness values $t(p_i)$ from HC plates to element thickness values $t(e_j)$ by means of

$$t(e_j) = \sum_i w_{ij} t(p_i).$$
If a finite element $e_j$ overlaps with only one HC plate, one related weight should be 1, all others should be 0. Thus, the finite element will receive the same thickness value as the overlapping plate. If a finite element $e_j$ overlaps with several HC plates, the related weights should be less then 1, but as the sum of all intersection areas should equal the element area, the sum $\sum w_{ij}$ should be 1. This would hold exactly if the HC and FE models represented exactly the same geometry.

Real-life models do not fulfil that condition. E.g., at the edge of the HC model, the edge of the finite element might be slightly outside the HC plate area, Fig. 6. This will result in a partial area of the finite element that does not intersect with any HC plate and the sum of weights will be less then 1. Another example might consist of two adjacent HC plates belonging to a curved surface. Their normal directions and reference planes will be different. If a finite element is close to both plates, the element’s projection might be strictly inside one plate’s projection and the corresponding weight will be 1. But as the other plate applies a different reference plane, the element projection into the other plane might still intersect the other plate’s projection, resulting in another weight greater than 0. Consequently, the sum of weights will be greater than 1.

The weight sum’s deviation from 1 is usually small (typically a few per cent) as long as the geometric mismatches are only caused by typical discretisation errors. But this might already cause thickness error magnitudes similar to thickness diminution magnitudes. Such errors are not acceptable. The problem was attacked by normalizing the weights:

$$\tilde{w}_{ij} = \frac{w_{ij}}{\sum_k w_{ik} t(p_i)}$$

and computing the element thicknesses as

$$t(e_j) = \sum_i \tilde{w}_{ij} t(p_i).$$

5.2 Identification of buckling fields

The HC model provides geometrical information about plates and stiffeners which are represented by closed resp. open 3D polygons, i.e. for plates, the outer contour is known. (This poses some problems for curved plates, see below.) Plates and stiffeners are categorized by structure members that provide basic type information (e.g. “transversal bulkhead”). Detailed assembly information/topology is not available. In particular, “plate panels”, denoting the contiguous plates of a structure member, have to be identified by computation of topological information from plate geometry.

To this end, first a trivial “PlateTopology” data structure is established by adding each contour vertex to a node list and by expressing contours by node indices rather than by vertex coordinates. In a second step, nodes are merged, i.e. for each node, nodes with distances smaller than a prescribed tolerance are removed from the node list, and contour node indices are correspondingly updated. Next, “hanging nodes”, i.e. nodes that belong to only one of two geometrically coincident contours, are identified and removed. To this end, for each contour segment, each node that is on the segment but is not the start or end node is inserted into the contour. A node is considered to lie on the segment if the height of the triangle formed by the node and the start and end nodes of the segment is smaller than the tolerance, and at the same time the inner angles at the segment start and end nodes do not exceed 90 degree.

The term "buckling field"/"elementary plate panel (EPP)" denotes an otherwise unsupported area of a plate panel that is bounded by stiffeners attached to the same plate panel or by adjacent plate panels. An EPP may at the same time consist of partial plates and extend over more than one plate, Fig. 8(c). The identification of EPPs that extend over more than one plate requires topological information. Unfortunately, the result of the aforementioned procedure is very sensitive to the tolerance, and often no global tolerance can be found that would close all gaps e.g. due to finite plate thicknesses without
creating artificial holes by too aggressive merging.

On the other hand, topological information is inherently available in the FE model. “Element panels” (denoting, in analogy to plate panels, the topologically connected elements of an element group) can be effectively computed by the following algorithm:

- start with arbitrary element
- assemble and mark as processed all elements that are connected via edges that are common to exactly two elements (the current element and its neighbour). In this way, element panels do not cross edges where more than two elements meet, e.g. at transversal bulkheads in Fig. 8b.
- Restart algorithm with arbitrary unmarked element until all elements are processed.

Detailed stiffener information is generally not available because stiffeners are usually accounted for by an effective element thickness rather than by actual elements. For this reasons, EPPs are computed from both element panels resulting from the FE model and stiffeners found in the HC model.

![Fig. 8: Close-up of a transversal bulkhead in (a) FE model, (b) processed FE model with stiffeners, and (c) HC file. The highlighted regions mark (a) an element, (b) an element panel, and (c) a plate. Elementary plate panels (EPPs) are regions bounded by longitudinal bulkheads, stringer decks and stiffeners. Stiffeners are idealized as additional stiffness and are hence not geometrically available in the finite element model (a), whereas plates extend beyond longitudinal bulkheads/stringer decks in the HC model (c). The hybrid model (b) is best suited for identification of EPPs.](image)

In order to subdivide element panels according to HC stiffener information, an "is on" relation between stiffeners and element panels has to be established. In contrast to HC plates, stiffeners and element panels are not related via common structural members (although usually some basic type information e.g. “transversal bulkhead” is available from both element groups in the FE model and structure members in the HC file).

A stiffener "is on" a plate, if, with some prescribed tolerance,

- the bounding boxes of the stiffener and the plate do overlap,
- stiffener and plate refer to the same structure member (not applicable to element panels),
- local z coordinate of the stiffener's vertices in the plate coordinate system is zero (usually not applicable to curved plates),
- the plate normal and the stiffener orientation point in the same direction.

In a second step, the panel is subdivided by intersection of panel edges and the stiffener. Unfortunately, 3D intersection of lines will often fail due to tolerance issues caused by skew lines resulting from e.g. different abstraction levels in finite element and HC models. For this reason, the 2D projection of the stiffener on the panel is used for intersection. The highlighted region in Fig. 9 is an example of an EPP resulting from this procedure. One more benefit from identification of EPPs in the finite element model is that for buckling assessment, finite element stresses do not need to be mapped to HC plates. The aforementioned algorithms have been implemented in a C++ class library “HcfGeometryTools” that is also used in e.g. the computation of compartments in the HC model based on NAPA bounding surfaces information.
5.3. Transfer of element stresses to elementary buckling panels

For buckling assessment of elementary plate panels, we need to get the FEM stresses from an appropriately related finite element. This can be done using a similar procedure. For each elementary buckling panel, a reference point and a normal direction need to be computed. This is used to locate the closest finite element. A conservative approach might perform buckling assessment based on the stresses of each overlapping finite element. The worst case buckling utilisation can be selected as the finite result for that elementary buckling panel.

5.4. Numerical example of thickness mapping

We performed the thickness mapping procedure for all plates of a container ship. In Fig. 10, finite element and HC plate thicknesses are shown for a transverse bulkhead. The central sub-figure shows the plates as represented by the HC model. The thickness values shown are the thicknesses that have been submitted to and approved by the society during the classification process.

The uppermost sub-figure shows the finite element discretisation of the bulkhead. The element thicknesses shown had been used for the initial FE-analyses during the early design phase. In most areas the FE thickness values exactly match the HC plate thicknesses. But there are also a few plates that are thicker or thinner than the related thicknesses of the initial FE model. Apparently, an early FE analysis revealed that certain plates were unnecessarily strong while some other plates were not strong enough.

The lowermost sub-figure shows the finite element model used for re-assessing structural strength. As the FE model from the initial design phase was re-used, the mesh geometry remained unchanged. But the element thickness values are different because the previously described method was used to transfer as-approved thickness values from the HC model to the FE model. Therefore, element thickness values now match HC plate thicknesses everywhere. Finite elements that are only covered by HC plates of identical thickness receive exactly this HC thickness value. Other finite elements extend about plate boundaries where a thickness transition occurs. Those elements partially receive different thickness values from several plates. Thus, the FE model also contains element thickness values that are a weighted average of adjacent plates’ thicknesses.
Fig. 10. Transversal bulkhead as a) FE model used in design phase, b) HC model with actual plates and c) FE model with element thickness mapped from HC model b)
6. Conclusions

The strength assessment of corroded hull structures is today based on a comparison of measured and required thickness values. In some cases, a longitudinal strength analysis is added. The latter is based on the computation of section moduli at the measured cross sections.

Additional steps are required for a re-analysis of the structural strength with the help of an FE model. Today, for many newbuildings such models are available. Moreover 3D technology can support the inspection and thickness measurement process. Based on this, it appears to be a natural step to combine these technologies for a more detailed analysis of remaining strength.

Several problems need to be solved to efficiently carry out such a re-analysis. Two of these, namely

- the technology for a transfer of measured thickness values between the inspection model and the FE model and
- the determination of structural details for post-processing of the strength analysis

have been described in this paper. A solution has been presented which can be regarded as a prerequisite for carrying out FE re-analysis in practice.

Still several open issues remain. In particular, the procedure how to appropriately choose the thickness values (as complemented to the technology to actually transfer them) of the corroded elements in the FE model and in a post-processing analysis need to be defined.

References


JARAMILLO, D.; CABOS, C.; RENARD, P. (2006); *Efficient data management for hull condition assessment*, Int. J. CAD/CAM 6, pp.7-15

PAIK et al. (2006), *Condition Assessment of Aged Ships*, 16th International Ship and Offshore Structures Congress (ISSC), Vol. 2, Committee V.6, Chapter 2

A New Proposal of Communication in a Numerical Model Basin Simulator

Fernando Henrique e Paula da Luz, TPN-Poli-USP, São Paulo/Brazil, fernando_luz@tpn.usp.br
Humberto S. Makiyama, TPN-Poli-USP, São Paulo/Brazil, betomaki@tpn.usp.br
Liria Matsumoto Sato, LAHPC-Poli-USP, São Paulo/Brazil, liria.sato@usp.br
Kazuo Nishimoto, TPN-Poli-USP, São Paulo/Brazil, knishimo@usp.br

Abstract

The numerical model basin simulator (TPN) aims to simulate the behavior of a moored offshore system in different environments (such as waves, currents and winds). The analysis is made by the interaction between floating bodies and lines in time domain. In the present architecture, each line is handled as an individual process, which has its own communication channel in the simulator. This paper presents a proposed new architecture, grouping a number of processes to use the same communication channel. A viability study of this new proposal was made.

1. Introduction

The Numerical Offshore Tank Laboratory (TPN) was created to perform research for the design and analysis of offshore platforms, ships, barges and other floating structures in deep water. In the development of floating units it is necessary to perform verification and validation tests with scale models in physical tests tanks, but in study scenarios in deep-water there are several physical limitations to the conventional tests, especially due to transversal dimensions and water depth which to do not allow faithful reproduction and the necessary precision needed for the dynamics of the whole system Nishimoto (2005), Nishimoto et al. (2003). The impossibility to duplicate these conditions is the necessity of a very large tank or the construction of smaller models. This is an example of a case when High Performance Computing is needed to perform analysis without the limitations of physical models.

On the other hand, high Performance Computing is an effective tool in research activities nowadays. Through the use of HPC it is possible to solve larger and/or complex problems that are impossible or impractical on a single computer, provide concurrency to perform several tasks simultaneously, and execute simulations in a much faster way. Over the past decades HPC has continuously improved the processing capacity, especially due to constant improvements of hardware elements (such as memory, processors, communications networks, etc…). More recently, in 2005, a paradigm change happened where the architecture of parallel machines was affected. Major vendors of HPC industry announced the microprocessors used in their machines would henceforth rely on multiple processors or cores Asanovic et al. (2006). Parallel algorithms used to solve a determinate problem have large association with the parallel hardware architecture that they will run on, such as network topologies, hardware nodes homogeneity, and file system of clusters.

This work focuses on optimizing the algorithm used in communication between processes in TPN Solver execution to adapt the changes of paradigm cited.

2. TPN Facility

The computational environment of TPN lab is composed by:

- **Cluster TPN-SUN**: 192 nodes (physical blades), each node has eight cores with hyperthreading capability, resulting in 16 virtual cores Intel Nehalen 2.8 GHz with 16 Gb of RAM memory. The total performance of cluster is 15 TFlops\(^1\).
- **Filesystem**: The file system used in both clusters is the Network File System (NFS), allowing

\(^1\) flops: FLoating point OPerations per Second. Tflops is equivalent to \(10^{12}\) flops.
every node to access the central storage through a network in a similar way to local storage. The total storage capacity is 150 Tb.

- **Network:** The connections between nodes are provided by an infiniband network, with 96 Gbits/s of traffic velocity.

### 3. Current TPN Solver Implementation

TPN Solver calculates a dynamic coupled analysis between bodies and lines connected to it. Each entity is calculated as separated processes, that is, one body can constitute one single process. The same is applied to lines. Because of that, TPN can demand a huge amount of computer resources. This issue leads us to implement a parallelized version as described by Luz et.al (2009).

The current infra-structure of Numerical Offshore Tank Solver was shown in Fig. 1. In this document, details regarding communication in TPN computer cluster are presented as follows:

![Fig. 1: Connection between processes created in a TPN Solver execution](image)

There are two kinds of processes:

- **Body Process:** Responsible to evaluate the environmental forces, such as wind, current and waves. Furthermore, it receives the tensions of each line connected and performs a calculation of positions to send to line processes as described as follows.
- **Line Process:** Responsible to perform the line’s dynamics and send the value of the force in its extremity to the body through a communications channel. It waits an update of its position to evaluate the next simulation step. Lines could be:
  - Catenary equation which is a parameterized equation used for describing the line,
  - Preadyn that utilizes a lamped mass formulation Silveira (2001)

The communication in a time cycle procedure is shown as follows:

1. The body calculates a new position to be sent to line process. Then it enters in standby mode until the lines send back a force value.
2. Each line process receives a new position and calculates a force value. Then it sends back the result to body and enters in standby mode.
3. After all line processes finish sending new forces to body, it starts a new simulation cycle.

### 3.1. Communications

In designing of current parallel algorithm of TPN Solver, each created process was to have its own channel of communication to other processes, and it was implemented through sockets, Hall (1999). Because of these constructions, it raises a number of alternatives to treat the problems regarding communication issue. In this paper, we address the key points presented below:
• **Channel of communication**: In older machines, the use of one communications channel per process did not have a negative influence on the execution time since the machines were constituents of a single core node. In the present clusters, the nodes have multiples cores where process jobs can be placed, reducing the need of sockets between them in a node.

• **Communication cost**: In parallel executions, there is a fixed cost to send and receive information between processes. In newer machines the time spent with certain types of calculations has greatly diminished so some calculations may not benefit from parallelization due to the extra communication cost between the processes.

4. Proposal of Grouped Communication

According to Rabenseifner *et al.* (2009), it is possible to obtain a better performance using a hybrid implementation for the parallel processing in a cluster Rabenseifner (2003), i.e., the MPI makes the inter-node communications and the OpenMP communicates through shared memory.

4.1. Hybrid implementation with MPI and OpenMP

The main modifications to be made in the TPN Solver is to switch the way that inter-process communications are performed by changing the use of MPI sockets and restructuring the process to allow the use of OpenMP. Changing sockets to Message Passing Interface (MPI) provides a higher level of abstraction, avoiding low-level coding, like using sockets directly. Then, the communication system can be set from outside, depending on the system is utilized by the software. Its standard is defined by an open process between manufacturers of parallel computers, scientists and application developers, http://www.hlrs.de/mpi/mpi22/. OpenMP is an API that supports the implementation of multi-threading programming in an easy and flexible mode, this interface is defined by a joint effort of the leading suppliers of software, hardware, and the scientific community, http://openmp.org/.

Fig. 2: Current and hybrid communication to TPN Solver

The main advantages achieved from applying hybrid communications are:

• Reduction of time spent to execute the simulation, because the Body process has to manage fewer connections, and thus, reduce the execution time per time cycle.
• Optimized use of available resources in the cluster. That is, since the time spent by a heavy line process define the duration of one time step, grouping a number of lines with light consumption process into just one core may reduce the usage of computation resources.

Fig. 2 shows the difference between the current model of communication and the proposed method. In current method, each line process has its own communication channel, overloading the system. On the other hand, the proposed method, using hybrid communication, processes share the same communication method. As result, each node has only one channel instead of many.

5. Proposal Communication Analysis

In the TPN Solver, time markers were inserted in the code, indicating what time the program spent in each code segment. During a time step, we divide a simulation step flow in three main phases:

• Wait - The process waits for an update of its connection position.
• Calc - The process is carrying out the calculation of its dynamics, and this time can vary due to several factors, such as type of methodology used to describe the dynamics of the line, number of line segments, etc.
• Send/Receive - Time that the process is sending or receiving data.

The TPN Solver source code, in architecture terms, is hard to manage, where communication is implemented by using low level socket functions. Thus, a test program was implemented to validate the proposal communication method.

Both, real and emulated cases were performed on the cluster, the first with two nodes and later with just one node.

5.1. Description of the real case analysis.

The case analyzed constitutes of 25 lines and 1 body, as follows:

• Line Type 1: 10 lines using Preadyn method with high quality discretization, causing a moderate processing load
• Line Type 2: 7 lines where the dynamics uses the Preadyn method, where the level of processing load is moderate and the discretization used in the lines was moderate.
• Line Type 3: 8 lines in which the dynamics uses the catenary equation. Also a moderate level of discretization was used.

5.2. Description of the simulated case analysis.

In order to validate the method proposed in this paper, processes that emulated the calculations of the line dynamics were created so that the time spent with each step could be measured. In this analysis two communication environments were prepared. The first one is based on pure MPI and the second is based on the hybrid communication. The case is constituted by 32 Lines and 1 Body, as follows:

• Line Type 1: 4 processes with calculation duration around 2.0 s;
• Line Type 2: 8 processes with calculation duration around 0.7 s;
• Line Type 3: 18 processes with calculation duration around 0.1 s;

5.3. Results

The results are presented as statistical analysis, showing the time spent by each kind of line. Also a staked plot was generated in order to have a better idea of the evolution of all evolved processes in time. In this graphic, x-axis represented each described processes and y-axis the temporal evolution.
5.3.1. Study case:

The mean value and standard deviation per step was calculated for each of the three types of lines; they are shown in Table I. The results show that the receiving and sending times are the same, even we change the number of utilized computer nodes. The total time spent in a time cycle can be determined by summing of “Calc” and “Wait” times. On average, 1 node spends 1.5 s to accomplish one cycle. On the other hand, 2 nodes spend 1.2 s to accomplish the same simulation cycle. Fig. 3 presents these measurements in a stacked plot. X-axis stands for the processes created in one simulation and Y-axis for timeline.

<table>
<thead>
<tr>
<th>Line Type</th>
<th>1 Node</th>
<th>2 Nodes</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Calc</td>
<td>Wait</td>
</tr>
<tr>
<td>line type 1</td>
<td>1.1 ± 0.1</td>
<td>0.3 ± 0.2</td>
</tr>
<tr>
<td>line type 2</td>
<td>0.21 ± 0.01</td>
<td>1.3 ± 0.2</td>
</tr>
<tr>
<td>line type 3</td>
<td>0.0002 ± 0.0001</td>
<td>1.5 ± 0.07</td>
</tr>
</tbody>
</table>

Fig. 3: Analyses using stacked plot of the real case study execution in TPN Solver, using 1 node and 2 nodes, respectively.

Since there are only 16 processors on one node, the TPN Solver takes longer to accomplish a simulation when executed in one node than two as shown in Fig. 3. The resource available is not enough to have an optimized distribution of processes; one core could handle more than one task. Because of that, the processor needs to switch from one task to another, compromising the performance of execution.

5.3.2. Simulated case study:

Table II presents the spent time in sending and receiving processes are despicable for either 1 or 2 nodes. However, the time spent in “Calc” plus “Wait” modes results in a total cycle time. For 1 node, it takes 2.3 s, i.e. 0.2 s longer than the case using 2 computer nodes. Fig. 4 shows its respective results in stacked plot. The graphic shows a disordered distribution of process in time when executed in one node. As a result, the time to accomplish four simulation cycles is longer than others. The hybrid version has almost the same performance for 1 or 2 nodes in computer cluster. The results show that changes in communication can directly affect the performance of a parallelized solution.
Table II: Mean and standard deviation per step in emulated simulation

<table>
<thead>
<tr>
<th></th>
<th>Current:</th>
<th>Proposed:</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>1 Node</td>
<td>2 Nodes</td>
</tr>
<tr>
<td></td>
<td>1 Node</td>
<td>2 Nodes</td>
</tr>
<tr>
<td></td>
<td>Calc</td>
<td>Wait</td>
</tr>
<tr>
<td>line type 1</td>
<td>2.000 ± 0.006</td>
<td>0.3 ± 0.1</td>
</tr>
<tr>
<td>line type 2</td>
<td>0.71 ± 0.03</td>
<td>1.6 ± 0.2</td>
</tr>
<tr>
<td>line type 3</td>
<td>0.14 ± 0.04</td>
<td>2.0 ± 0.2</td>
</tr>
</tbody>
</table>

Fig. 4: (a) Simulation of current communication executed in 1 node; (b) Simulation of proposed communication executed in 1 node; (c) Simulation of current communication executed in 2 nodes; (d) Simulation of proposed communication executed in 2 nodes;

6. Conclusions

Hybrid method has a better performance for TPN solver architecture. The bottle necks in simulation are not critical. Then, processor cores can easily handle more than one process without significant losses. This is possible because processes in TPN do not demand resources equally, allowing an optimized distribution of them on the processor. That is, some cores can handle light tasks utilizing the available waiting time determined by heavier ones. On the other hand, the results show that pure MPI method does not handle properly when the solicitation in one core surpasses 1 task per core.
The opportunity to gain a better management of communications became evident in the results of the emulated case. Processors equipped with multiple cores using hybrid method results in high inter-process communication performance. It allows a better utilization of computational resources.

A study case of this new proposal of communication was important to have a prediction of what would be the impact of this change on the implementation in TPN Solver. The results provide good perspectives for implementing these changes.

Acknowledgments

The authors would like to thank all the researchers from the TPN laboratory for their comments, contribution and guidance through the research. Special appreciation is given to Caio da Silva Assunção (USP).

References


LUZ, F.H.P.; GASPAR, H.M.; NISHIMOTO, K. (2009), System architecture of a numerical model basin simulator, 8th COMPIT Conf., Budapest, pp.474-485

NISHIMOTO, K. (2005), Development of numerical offshore tank for ultra deep water oil production systems with multi-bodies, Int. Symp. on Technology of Ultra Deep Ocean Engineering, Tokyo


RABENSEIFNER, R. (2003), Hybrid parallel programming on HPC platforms, 5th European Workshop on OpenMP, Vol. 3, pp.22-26

RABENSEIFNER, R.; HAGER, G.; JOST, G. (2009), Hybrid MPI/OpenMP parallel programming on clusters of multi-core SMP nodes, Parallel, Distributed and Network-based Processing, Weimar, pp.427-436

Driving the Adoption of Cutting Edge Technology in Shipbuilding

Denis Morais, ShipConstructor Software Inc., Victoria/Canada, Denis.Morais@shipconstructor.com
Mark Waldie, ShipConstructor Software Inc., Victoria/Canada, Mark.Waldie@shipconstructor.com
Darren Larkins, ShipConstructor Software Inc., Victoria/Canada, Darren.Larkins@shipconstructor.com

Abstract

Despite being an industry full of bright and qualified people, the shipbuilding industry is often slower to adopt and develop new technologies compared to other sectors such as the plant, aerospace and automotive industries. This paper will explore the trends in the relative adoption of a number of technologies including laser scanning, digital prototyping and shop floor 3D. It will examine a number of contributing factors to the overall slower rate of adoption in shipbuilding and discuss potential strategies and technological developments that could mitigate these factors.

1. Introduction

Shipbuilding is one of the most complex and demanding of the manufacturing industries combining aspects of both direct product manufacturing and capital project development. Because of this complexity, shipbuilders could benefit from the adoption of various productivity enhancing tools. The adoption of sophisticated technology is particularly important in high-wage countries which are under competitive pressure from shipbuilders in low cost emerging markets.

The shipbuilding industry is full of bright and qualified people yet it lags other sectors such as the plant, aerospace and automotive industries. Sometimes there are technical reasons for this lag and other times the reasons have more to do with the structure and the culture of the industry. Fortunately, there are strategies and technological developments on the horizon that could assist shipbuilders to cost-effectively implement cutting edge tools and techniques.

2. Laser Scanning (High Definition Surveying)

One technology that is sometimes used in shipbuilding but could not as frequently as it could is laser scanning (High Definition Surveying). High definition surveying is a non-intrusive means of rapidly collecting detailed and accurate as-built data. This technique uses a narrow laser beam to sweep across a target object so that hundreds of thousands of closely spaced measurements can be taken in a matter of minutes. When these scanned measurements are displayed on a computer, a dense representation of the target results (a point cloud) can be viewed. This can be viewed and navigated much like a 3D model. CAD objects can be modeled around this background or the point cloud can be used to generate a CAD model. In a manufacturing context, key applications for this technology are to aid assembly, validate manufacturing and to assist repair and refit activities.

2.1. History of Laser Scanning in Other Industries

3D laser scanning was first used for industrial purposes back in the late 1980s. The scanners were small “table-top” units which were used for a variety of purposes such as reverse engineering of small objects. Larger scale laser scanning (high definition surveying) for commercial purposes started in the late 1990s. In 1998 Cyra Technologies (now Leica Geosystems) introduced the Cyrax 2400 laser scanner and other companies such as MENSi and RIEGL entered the market at approximately the same time. RIEGL initially focused on using laser scanning for the mining industry while Cyra and MENSi had close connections to plants. Plants became the most enthusiastic customers in the early years and the technology spread so widely that by the year 2005 it became mainstream to scan as-built models. Offshore oil and gas platforms, being essentially floating plants, are now commonly scanned by contractors constructing and repairing oil rigs.
The automotive and aerospace industry also regularly use high definition surveying for retrofitting factory floors and have long used laser scanning in general to measure parts for accurate fit.

In recent years, the cost of a laser scanner has come down to about $30,000. Due to the mission-critical nature of plant operation, $30,000 is a paltry sum since it can cost $500,000 to lease an oil rig for a single day. If the rig needs to be repaired, every day it is out of commission is a substantial loss of money. A power plant or factory can also not be “down” for very long without significant repercussions.

If repairs need to be done and no CAD model previously exists, a scan will be done to generate an accurate as-built model. Even if a CAD model has previously been created, laser scanners are used to check what was actually constructed because this almost always varies from what was originally planned.

Fig.1: Laser scanner for high definition surveying

Fig.2: Mitsubishi automobile factory renovation. Plant layout checked with point cloud derived via laser scanning in fall of 2010. Data input into Autodesk Navisworks
2.2. Laser Scanning in Shipbuilding

The shipbuilding industry has similar reasons to adopt laser scanning as in other industries. Just as in the aerospace industry, putting the parts and blocks of a ship together is analogous to connecting the assemblies of an aircraft. Accurate measurement helps ensure that everything fits. Quality control is naturally always important but perhaps the biggest benefit comes in repair and refit activities. Today, an average large tanker or military ship will be in the water for anywhere from 35 to 50 years. The majority of these vessels lack basic engineering drawings and blueprints, let alone computable design data for designers and engineers to reference when attempting to repair damaged or aging vessels, convert vessels for other uses, or retrofit them for more efficient and safe operation. Even the safe disposal of older ships has become a significant challenge as potential dangers from lead paint and other chemicals were not adequately documented when the ships were built or repaired. Today’s shipbuilders and manufacturers need fast, accurate, and cost-effective methods to redesign, recreate, reinvent, salvage, and even destroy ships.

Several progressive shipyards such as Meyer Wert Gmbh, Signal International, and Babcock International use laser scanning but the utilization of this technology is not standard amongst shipbuilders and is often opposed. For instance, when Norway’s Marathon Petroleum Company was modifying its FPSO ‘Alvheim’, Project Manager Bryan Wallace noted in a ‘Marine Log’ case study about the project that the use of laser scanning was resisted by various individuals, Greaves (2006). However, according to Ole Martin Dahle, a member of the engineering team working on the project, laser scanning showed several benefits. Laser scanning was used to create a model of the piping and structure and the model information was used to determine the piping that would be demolished to put in reserve volumes for new equipment. This had to be completed before sending the ship to the Keppel Shipyards in Singapore for the actual modifications. Applying topside clash controls method-
ology to the scanning-based 3D model revealed significant piping clashes. As this was found early in the design process they were easy to fix, saving significant late “field modifications” which would have been costly and much more difficult without the scan data.

2.2.1. Signal International’s Usage of Laser Scanning

Even though most shipbuilders do not use laser scanning technology, one shipyard that does use this technology is Signal International, a shipbuilder with multiple facilities in the US Gulf Coast. Signal uses a Faro laser scanner on as-built models to check both new production as well as to generate CAD models for refit projects. After going through some intermediary software, point cloud information is ultimately imported into the AutoCAD based ShipConstructor CAD/CAM application to create:

- Accurate bill of materials
- General arrangements
- Pipe arrangements
- Pipe ISO’s by system
- Pipe spool drawings
- Equipment details
- Structural arrangement

2.2.2. Lack of Use in Shipbuilding

Shipyards building and repairing oil and gas platforms regularly use laser scanners but a question that needs to be answered is why this technology is not used more frequently in the construction and repair of ships? Why is less accurate and more time consuming hand-measuring still the norm?

This is somewhat surprising in that the shipbuilding industry was intimately involved in the early history of the technology. The prototypes of the Cyrax 2400 back in the late 1990s were developed in conjunction with the US Navy and one of the specifications for the product was that it had to be able to fit through a standard manhole on a naval vessel. The shipyard, Babcock International was also one of the first customers ever to purchase the Cyrax 2400.

Another point to be noted is that the shipbuilding and repair business is similar to the construction and repair of a plant if one considers the economic cost of inaccuracies. Shipbuilders often face massive consequences for delays; penalties of hundreds of thousands of dollars per day are not unheard of. That means that a single day’s penalty would thus cost more than the purchase of a scanner. Shipbuilders could benefit from more widespread adoption of laser scanning technology. Reasons why they do not do so will be examined in a later section of this paper.

3. Digital Prototyping

Another cutting edge technology is digital prototyping. A quick perusal of shipbuilding magazines over the last decade will reveal that there has been much talk about the application of 3D models and utilizing Virtual Reality. This all presupposes the existence of a “digital prototype”, in other words, a digital or virtual mock-up of a ship. The concept behind using digital prototypes is to do as much work, analysis, and communication as possible within a 3D digital environment, rather than in the physical world. This involves using computer simulations more than physical models and viewing videos and fly-throughs, rather than interpreting paper drawings.

The concept of digital prototyping goes beyond simply creating product designs in 3D. In other industries, it would involve product development teams assessing the operation of moving parts to determine whether or not the product will fail and seeing how the various product components interact with subsystems. In a shipbuilding context, it would involve activities such as FEA and CFD analysis of hulls. It also extends into detail design and production and even into PLM activities.
3.1. The Aerospace Industry

The aerospace industry is in the forefront of the push towards digital prototyping and operating in the virtual world. Digital mock-ups appear to have been standard in the industry since the early 1990s. The automotive industry is following while in shipbuilding, digital prototyping is not used as extensively.

3.1.1. Boeing 777 & 787

The Boeing 777 and Boeing 787 are commonly considered to be “poster children” for the concept of digital prototyping. The Boeing 777 is often advertised as being the first paperless designed aircraft since the 777 was the first commercial aircraft to be designed entirely on computer. Boeing extended the use of the digital prototype into all aspects of the company. They developed their own high-performance visualization system, FlyThru, later called IVT (Integrated Visualization Tool) to support large-scale collaborative engineering design reviews, production illustrations, and other uses of the CAD data outside of engineering. The result was an enormous success. The use of CAD/CAM/CAE software allowed a virtual aircraft to be assembled in simulation to check for interferences and to verify proper fit of the many thousands of parts, thus reducing costly rework. In fact, over 3000 assembly prototypes were eliminated. Virtually all of the company’s aerodynamic modelling was also done without using physical models in a wind tunnel. The development of the Boeing 787 Dreamliner further extended this success story. Only a dozen prototypes were needed for aerodynamic testing and the extension of the digital prototyping concept saved time and money in other areas as well.

Being able to virtually simulate not just the parts, but the plane’s processes was a great boon for Boeing, giving them more flexibility to make adjustments during the design phase. For example, at one point, the chief pilot for the 787 was doing a virtual test flight and was able to see some issues related to fin control. Using a digital prototype, designers were able to evaluate 50 new possible fin configurations, test them and make the appropriate changes to the rest of the design in only about four weeks. With the old way of working, Boeing might have been able to evaluate three or four new fin configurations and it would have taken at least three or four months. Boeing’s success with digital prototyping has led to this process being the standard in the aerospace industry.

3.2. The Automotive Industry

The automotive industry has not as fully integrated digital mockups into their entire organizations as the aerospace industry but the use of virtual representations of vehicles for analysis and training purposes is still fairly advanced. Indeed several interesting applications of the concept have appeared in the last few years.

3.2.1. Jaguar Land Rover’s Digital Cave

One recent application of Virtual Reality is the “digital cave” from Jaguar Land Rover. In 2008, Jaguar Land Rover unveiled this tool. As with other digital prototyping systems, it allows engineers to simulate testing in wind tunnels, drive vehicles in a variety of conditions and to design mechanical components, Weaver (2010). Instead of looking at a computer terminal however, it lets engineers enter a “cave” surrounded by four acrylic screens and uses eight Sony SRX-R104 projectors to create an image with 4096 x 2160 resolution—more than four times the definition of a 1080p HDTV. Two projectors on each wall create the stereoscopic perspective. The Virtual Reality cave allows an operator wearing 3D glasses to manipulate images using a wand, similar to a modern games console controller. According to Jaguar Land Rover, the digital cave has saved more than £8m in development costs.
Ian Anderson, Advanced Engineering Programme Manager, said in a UK Automotive Council publication, “While the time taken to develop a car is still about three years, the greater complexity of vehicles has significantly increased the amount of work required. The Virtual Reality system helps us to ensure that we still deliver the highest quality products, because we’re using our time in a much more efficient way.”

3.3. The Use of Digital Mock-ups in Shipbuilding

As previously mentioned, references to Virtual Reality, and “digital ships” etc. are common within the shipbuilding industry though these tools do not have universal usage. Shipbuilders are utilizing digital prototyping for Finite Element Analysis (FEA) as well as for Computational Fluid Dynamics (CFD) calculations. They are also using Virtual Reality for design review. However, none of these tools is used as extensively as in the airline or automotive industry at the moment.

3.3.1. FEA and CFD

The usage of FEA is quite common in the shipbuilding industry but CFD, though gaining in popularity, is currently less frequent. Examining the situation with CFD more closely, one might consider wind tunnel testing for aerodynamics to be roughly analogous to tank testing of a ship model for hydrodynamics. CFD algorithms are used in both cases to approximate results and to reduce the amount of physical prototypes needed. One difference is that there has never been anywhere near the amount of tank testing done on new ship hull designs as there has been with airplane wind tunnel testing of wings. After all, the performance of a hull is not nearly as critical as the ability of a plane to lift off the ground. Consequently, it is of no surprise that less CFD analysis on digital prototypes is performed in the shipbuilding industry.

However, due to the rising cost of fuel and constant pressure for yards to construct “green ships”, one might think that there would be more demand for this technology than there currently is. The United States Navy, for instance, is spending large sums of money researching algae based biofuels and there is talk of ideas for cold ironing in harbours. Therefore, one might think that analysis that could be done to improve the fuel efficiency of a hull could give a shipbuilder a significant competitive advantage. If CFD could be proven useful, surely it would be regularly used rather than expensive tank testing or no tank testing at all.
In defence of the industry on the other hand, one can legitimately point out significant limitations in the current CFD algorithms used for marine applications. Analyzing a ship hull’s motion through waves is a massively more complex problem to solve than analyzing the flow of air over a wing. CFD is currently where FEA was twenty years ago so it could be argued that it is not surprising that shipbuilders do not use CFD tools as much as aerospace manufacturers. Having said that, future developments in technology may mitigate this problem and these will be explored later on in this paper.

3.3.2. Virtual Reality for Design Review

The use of Virtual Reality (VR) for design review is becoming more common as CAD vendors keep promoting the concept. ShipConstructor Software Inc., for example, has encouraged its clients to utilize the Autodesk Navisworks VR viewer to check designs and numerous clients have adopted this tool into their 3D design process.

3.3.3. VR Case Study: Chetzemoka Project

A good case study regarding the use of Virtual Reality for design review is the design and construction of the most recent Washington State Ferry, the ‘Chetzemoka’, launched in 2010. Washington State Ferries was actively involved in the design and construction process which involved multiple different companies including Guido Perla & Associates, Todd Pacific Shipyards, Jesse Engineering, Nichols Brothers Boat Builders and Everett Shipyard. The 3D CAD model was created in the ShipConstructor AutoCAD based application from which a Navisworks VR model was derived. All the different parties involved had regular meetings to view the Navisworks model and this viewing of the Navisworks model significantly aided the multiple organizations to work together to complete the project.

3.3.4. Commonality of Older Methods

Notwithstanding the strides being taken by numerous shipbuilders to adopt the digital approach, simple two-dimensional, paper drawings created in basic software are still the backbone of the shipbuilding industry, especially in lower wage countries. Even when 3D models are used for design review it is usually only managers and engineers who are using the data. Workers in the shipyard rarely participate in these reviews as they do in the aerospace industry. Because of this lack of inclusion, designs are not optimized for efficient production.

4. Shop Floor 3D

Having engineers view digital prototypes on a screen in their office is one thing but getting the 3D model down onto the shop floor extends the concept even further. This goes beyond simply having production workers view the design. It involves labourers utilizing the CAD model on a day to day basis. The idea behind shop floor 3D is to allow the manufacturing team to better understand their jobs. To truly integrate the digital prototyping concept, the 3D model cannot be stuck in the office.

Autodesk Inc. is promoting the use of the shop floor 3D concept as a logical extension of its digital prototyping tools used in the Inventor Publisher Program. There are limitations on the number of parts that Inventor can handle so Inventor Publisher would not be as suited for the large 3D CAD models utilized in shipbuilding. However, unfortunately, few shipbuilders have even been asking for this functionality at the present time. Only limited strides have been taken in the direction of shop floor 3D at even the most progressive shipbuilding companies.

In shipbuilding, a 3D model is transformed into 2D drawings which must be interpreted by workers in the yard. Engineers who create the instructions determine what the amount of detail they think is required and draw objects the way they think is best. The use of 3D data would be a more effective method of communication and a more powerful tool for production workers.
4.1. Japanese Manufacturing – Shintec Hozumi

By contrast, Japanese manufacturers have truly brought the benefits of digital prototyping to the shop floor. An example of a company using what a Japanese Manufacturing textbook calls a, “familiar use of 3D data at the workplace,” is a company named Shintec Hozumi, Hiroshi (2009). Shintec Hozumi is a company with 400 employees whose business involves both manufacturing of automobile production facilities and factory distribution systems, as well as documentation services for digital content. The company converts facility design data from 3D CAD to XVL, and then displays that data without needing the CAD platforms. Large monitors and displays on carts are installed across the plant so that staff can search for required manufacturing information as well as refer to 3D data. As a result, production staff members are able to directly access information that shows the 3D shapes of products being assembled alongside information such as parts names, and so on. The company has found that this tool aids in communication and productivity.

Fig.5: Display installed in factory shows products being manufactured

Fig.6: Drawings used in prior process
4.2. Shop floor 3D in Shipbuilding – Royal Huisman Case Study

Even amongst the most sophisticated Asian shipbuilders, shop floor 3D is still considered to be exotic. One shipyard is known to use this technique however and that company is the prestigious yacht builder, Royal Huisman from the Netherlands. It makes use of a Navisworks terminal in its production facilities. The terminal is not used to show assembly sequences but shipyard workers can use it to fly-through, zoom in and check items in a Navisworks 3D model.

4.3. Future Directions in Shop Floor 3D

Simply allowing production staff to view the 3D model is only a first step. Showing complete build sequences via 3D animations would dramatically aid in communication compared to simply viewing static models. The use of tablet computers would also allow mobile shipyard workers to bring the digital prototyping experience with them, further enhancing productivity. Currently, other industries are starting to investigate these types of cutting edge tools and it is hoped that shipbuilders can further embrace this technology. It is true that due to the size and complexity of 3D CAD models utilized in shipbuilding, currently there are limitations imposed by the software and hardware that affect the viability of this approach. However, other technological developments may possibly overcome this difficulty and they will be explored later on in this paper.

5. Reasons Why Shipbuilding Lags Other Industries

It has been shown in this paper that the shipbuilding industry often lags other industries in the adoption of cutting edge technologies. There are multiple reasons why this is the case. Some of the reasons have to do with the limits of hardware and software and the current state of CFD algorithms. Shipbuilding is far more complex than most other manufacturing challenges and technology is often not available to meet shipbuilders’ needs. In other cases however, the challenges have more to do with the structure of the industry itself. This affects the culture of shipbuilding and also affects the money available to develop and implement cutting edge solutions.

5.1. Complexity of Shipbuilding

The most obvious reason why shipbuilders have not always been the first to adopt cutting edge technology is that the technology has not always been suitable for shipbuilding. The CAD models required for shipbuilding are far more complex than in other industries, especially on naval projects. For instance, a car consists of about 3000 parts and a Boeing 777 has 103,000 parts but a SSBN submarine consists of over 1 million different pieces! The software and hardware required to handle the complexity of digital prototyping can therefore be dramatically different from what is required in other industries. This is a significant issue that should not be minimized. Sometimes naïve observers underestimate the problem with virtual simulation by comparing shipbuilding fly-throughs to a video game but this is a false comparison. Despite the apparent complexity and detail of a video game, the information displayed in a video game uses simple surfaces. Objects do not have as much detailed information associated and more importantly, objects do not have to be exact, they just have to fool the eye.

Meanwhile, the motion of a large hull through waves is a far more difficult problem to analyze than airflow over a wing. The CFD algorithms simply have not been developed to handle this level of complexity and unlike with FEA, throwing more computing power at the problem will help but will not solve the underlying mathematical issue since CFD analysis does not converge to a solution in the same way.

5.2. Design & Engineering a Smaller Proportion of Cost

The Automotive Council in Britain says that it can take 3 years to develop a new car model. However, other sources note that a car only takes 23 hours to build. It takes between 5 to 10 years to
develop a new airframe but according to Boeing, only about a year to build a plane. Material costs are high but design and engineering costs are much more important. Design is not as significant to the budgets of a shipbuilder as it is to an aerospace or automotive manufacturer who is going to spend proportionately a longer amount of time in research and development. It is understandable then that shipbuilders tend to focus more on the cost of materials and labour, rather than on implementing the most innovative systems for initial design and production engineering. The affect of this issue goes beyond just money. From a cultural standpoint, design and engineering are not seen to be as critical as they are in other industries.

5.3. Smaller Series Runs

Using the Boeing 747 as an example, over 1000 planes in this series have been produced since 1969. As for cars, hundreds of thousands of vehicles are produced for a single popular model each year, e.g. about 500,000 Toyota Corollas were produced in 2009. This means that the cost of design and engineering can be amortized over a large number of vehicles in other industries but in shipbuilding this is usually not the case. Often, only a single ship is produced from a design. An order for ten vessels would usually be considered large in most contexts. It is therefore considered less justifiable to spend more money on design and engineering because the cost cannot be spread out as much as in other types of businesses.

5.4. Requirement to Make Money on Each Ship

Closely related to the previous point of amortizing costs is that in shipbuilding, companies are expected to turn a profit on each vessel being produced. There is therefore less opportunity to invest in implementing a new technological solution that will only pay off in the long run but in the short term will eat into profitability.

5.5. Build Before Design is Complete

Unlike with the automotive industry, shipbuilders are still designing as they are building. This means there is less opportunity to plan and implement the best methods of design and production.

5.6. Lack of Payment during the Design Phase

In the aerospace industry and automotive industry, a company can more readily raise money for the design phase. Meanwhile in shipbuilding, payment is tied to certain milestones in design. There is therefore an incentive to quickly reach each individual milestone instead of taking the time to implement a new solution that will benefit that shipbuilder in the long run.

5.7. Smaller Industry

Shipbuilding is a niche industry in terms of the overall CAD Market. According to Ed Martin, Sr. Industry Manager of Autodesk Inc., the automotive industry makes up about 25% of the CAD market and aerospace makes up an additional 20%. Shipbuilding only constitutes a tiny percentage of the customer base. Thus, there is less incentive for CAD vendors to cater to the needs of a complex industry when more money can be made in other market segments. An aerospace manufacturer is almost always going to have more pull than a shipbuilder ever will.

5.8. High Fragmentation of the Shipbuilding Industry

There are 34 aerospace companies listed in the Yahoo directory and 50 companies that are listed as members of the International Organization of Motor Vehicle Manufacturers. These numbers include just about everyone in the business, large and small though both industries are dominated by a much smaller number of firms. Meanwhile, there are 1574 shipbuilders doing new buildings listed in the Worldwide Shipyards 2011 Directory. This means that any individual shipyard has less buying power
to convince a CAD vendor to listen to its needs. It also means that there is much more room for an inefficient shipyard to survive. If Boeing adopts a certain technology, Airbus is forced to adopt it as well in order to compete. This is not as pressing a demand in the shipbuilding industry.

5.9. Many Shipbuilding CAD/CAM Software Vendors

Just as there are numerous shipyards, there are several different CAD programs used in the shipbuilding industry including ShipConstructor, AVEVA Marine, Nupas Cadmatic, Foran, CATIA, Intergraph Smart Marine, MATES, Autoship, PTC, CADD S4, MasterShip, Bentley etc. While there are complaints in the automotive industry that there are too many different software solutions used, there is still less fragmentation partially because there are less car manufacturers. In aerospace, a large company such as Boeing can simply mandate that all suppliers use a particular software application. Because there are so many CAD/CAM vendors in the shipbuilding market, each company is by consequence smaller and thus has less capital available for the R&D needed to implement cutting edge solutions.

5.10. Lack of Large Immediate Payoff for Change

As noted before in the section on laser scanning, there is a sense of urgency to complete the repair of an automotive factory or the retrofit of an oil rig. While there always is a cost to having any ship out of commission and there is always a cost to inaccurate designs, these issues are felt less strongly in the general shipbuilding industry. Indeed, the shipbuilders most likely to implement laser scanning are those involved in the offshore industry.

6. Overcoming Challenges

If you examine each of the above reasons for not implementing cutting edge technology, you will see that they can be broken down into four different underlying themes:

1. Lack of money available to CAD vendors to develop shipbuilding solutions
2. Culture of the shipbuilding industry
3. Lack of money available for shipbuilding design and engineering
4. Complexity of shipbuilding

Solutions have to therefore address those four fundamental issues. It has been noted that currently there are progressive shipbuilders who are utilizing laser scanning, digital prototyping and shop floor 3D so there is reason to believe that others will join them in the future. It is believed that the strategies and trends mentioned below will help make the adoption of the technologies mentioned in this paper more widespread throughout the shipbuilding industry.

6.1. Utilizing Shipbuilding Specific Software Tied to a Large Generic CAD Vendor

The first development that will aid the adoption of advanced technology is the growing popularity of shipbuilding software tied to a larger generic CAD vendor. As mentioned before, there are numerous different CAD applications used in the shipbuilding industry. Often these are produced by relatively small companies compared to those that dominate the aerospace and automotive sectors. Unfortunately, small niche companies by themselves do not have the R & D budgets to implement solutions such as the integration of laser scanning point cloud data. On the other hand, there are undeniable advantages to picking a software package that is tailored towards shipbuilding rather than another industry. A way to get the best of both worlds is to purchase a software package specialized in shipbuilding but yet is tied to a larger more generic CAD vendor developing cutting edge technologies. If the tie between the niche shipbuilding CAD company and the larger vendor is close enough, the shipbuilding-specific CAD vendor can help influence the development of the larger company so that it transfers technologies from larger industries into shipbuilding. For example, ShipConstructor software is a niche shipbuilding application built on top of an AutoCAD foundation,
tied to Microsoft SQL Server as well as other Autodesk products such as Navisworks. Autodesk is actively pursuing development of all of the technologies described in this paper.

6.2. Implementing a scalable solution

It is becoming widely recognized throughout the business world that a common denominator in the successful implementation of new processes and technology is a high level of engagement from all levels of the workforce. This can be aided if the implementation of a new process is done in a cost-effective, non-threatening, step-by-step fashion whereby new technology is seen to benefit everyone involved. The cost of any new technology mentioned in this paper is usually not that expensive in and of itself because purchasing new software and hardware is relatively inexpensive. The real cost comes with the implementation of those tools which is why a scalable solution is generally recommended.

The Royal Huisman example mentioned earlier is an example of how a scalable approach was successfully utilized to implement the shop floor 3D concept. First, Royal Huisman made black and white Navisworks drawings available to workers on the shop floor and this proved to be immediately popular. The workers liked the black and white drawings but found that they had difficulty differentiating various objects so they asked to have colour printouts instead. This request was granted and the drawings were produced for them in colour. From there it was a simple transition to providing a Navisworks terminal on the shop floor. The process was a natural evolution which was seen to benefit everyone. Extrapolating from this scenario, after shop floor 3D has been implemented and the manufacturing team starts using virtual models for production, there will likely be more communication between the design and manufacturing team. Shop floor workers would more likely be included in the design process and designers will start recognizing the value of their input. These are cultural issues and they are best addressed via scalable software solutions. A scalable approach also helps from an economic perspective because the costs can be spread out over time. ShipConstructor prides itself on being lightweight and scalable. Users can adopt more and more digital prototyping functionality, and adopt shop floor 3D and laser scanning as they grow in sophistication.

6.3. Cloud Computing

Cloud Computing is a buzzword at the moment in the computing industry that has famously been mocked by Larry Ellison of Oracle. However, if one ignores the hype and hyperbole, there is actually something profound happening. One key benefit of Cloud Computing is the capability for “infinite computing” in that it harnesses the computing power of distributed computers around the world, scaling to the needs of usage at any given time. Data can be accessed and applications can be run from remote locations without expensive hardware and software. This has given rise to many applications being offered via a (Software As A Service) pay-per-use model. The potential applications of the various aspects of Cloud Computing are significant and could help solve the cost problems previously mentioned.

A perfect example of this is how Cloud Computing could spur the increased adoption of CFD analysis in the shipbuilding industry. As noted earlier in this paper, the underlying algorithms could still use improvement but this will happen over time. As CFD is used more frequently in shipbuilding applications, there will be increased incentive to improve the CFD programs which will create a positive feedback loop and Cloud Computing is possibly the technological development that will start this process in motion. The reason that CFD analysis will likely be used more is that the cost will come down via the adoption of a Software As A Service (SAAS) model. One reason why the shipbuilding industry performs CFD analysis less than it could is related to the cost of purchasing the hardware and software required. Since CFD would typically only be done on significantly different new hulls, at the present time, it might be difficult to justify the large, infrequent expense. On the other hand, if a naval architect accessed a CFD program on the cloud and only paid for the usage required, the cost could be noticeably less. No hardware and software would have to be constantly updated; the SAAS CFD company would look after that detail.
The “risk” would also be lower in that if one was not happy with the results of a particular experience, another vendor with a different algorithm could be chosen. Indeed, several different CFD programs could be used to perform analysis on the same hull and the results could be compared. In fact, the cost might be reduced to the point that multiple analyses could become much more common. The increased computing power available due to the cloud infrastructure could also scale to the complexity required as the CFD shipbuilding algorithms become more advanced. Therefore, Cloud Computing could help mitigate both the cost and the complexity problems hindering the adoption of some aspects of digital prototyping as identified earlier in this paper.

6.4. Parallel Processing

Perhaps the most noteworthy technological development that could solve the complexity problem is the fact that new hardware is increasingly utilizing parallel processing. This is happening at the CPU level via multi-core technology and can also be seen with recent trends in Graphics Cards. The performance improvement of the additional processing units is largely dependent on the implementation of the software program architecture. If CAD vendors adapt their software to take advantage of this development in hardware architecture, performance will be dramatically improved. Both Autodesk and ShipConstructor have recognized the importance of this issue and are developing software in such a way as to utilize the power of the increasing processing units.

7. Conclusion

Shipbuilding is a complex industry with a unique structure. While some shipbuilders are utilizing new technologies such as laser scanning, digital prototyping and shop floor 3D, these technologies are not as common as they are in other industries. There are several reasons why this is the case and these reasons involve complexity, culture and money. Fortunately, new technological developments are helping mitigate each of these factors. These factors include scalable shipbuilding software solutions tied to large generic CAD vendors with high R&D budgets, Cloud Computing and Parallel Processing. Together, these factors could help shipbuilders cost effectively adopt cutting edge processes and tools.

References


HIROSHI, T. (2009), *Improving lean manufacturing through 3D data*, Lattice Technology, pp.4.5-4.6
Increasing Efficiency in the Ship Structural Design Process

Tomi Holmberg, Napa Ltd, Helsinki/Finland, tomi.holmberg@napa.fi
Shaun D. Hunter, DRS Defense Solutions LLC, Stevensville/USA, SHunter@drs-ds.com

Abstract

This paper presents new possibilities and practices that can enable shorter design cycle times by utilizing currently available commercial-off-the-shelf structural modeling technology combined with full-ship FEA and limit state analysis technology. Essentially, the use of a single 3D structural design model in both design and analysis has the potential to reduce the overall time needed in the design process by decreasing the time needed to run the design cycle anew for another variation, resulting in better designs.

1. Introduction

Traditionally the early structural design of ships is done in 2D, i.e. the design information is stored in 2D drawings. All the information needed to validate design integrity is read from the drawings and input to several different systems where the actual feedback for the structural integrity is coming. The same information is re-entered multiple times making the process very inefficient and exposed to human errors. Often, the information for a structural design calculation is not required to be as a 3D representation of the structures because cross-sectional information is adequate enough (e.g., information for longitudinal scantling checks).

The benefits of having 3D information right from the beginning of the design process has been discussed for years; however, few organizations have changed their way of working to support this notion. The main obstacle is the resistance to change the way of performing design work. When 3D information is the main source of the design input, it requires a change in the existing working methods in order to realize any benefits of this 3D approach. It is necessary to implement a new way of working for the ship engineering process in which the project team collaborated for utilization of the product model. The previous big change in the design process was done when moving from hand-drawing to computer aided drafting tools. This change was in some ways much easier to do, because the design process could remain the same as the design information was stored the same way, in 2D drawings, but now in electronic format.

One major statement against 3D models is it takes time to create one. Everything is relative. It depends on the type and the quantity of the design information to justify if 3D models bring any benefit. If the requirement is to create just a few structural drawings then it is hard to speak on behalf of 3D models. The drafting is not the only task needed to be carried out in order to get the structural drawings ready. For example, the scantling information needs to be initialized and created, which means additional information to support particular calculations needs to be derived. The 3D model has this type of information; therefore, the 3D model can help to reduce the time for carrying out these types of additional activities.

Fig. 1: Design spiral in two different approaches for storing design data
Fig. 1 illustrates how two different design approaches tend to behave in the design spiral relative to the time. In the early adaptation of the 3D design approach, the 2D approach tends to be faster to get the first round activities completed. However, the ship design is usually very iterative and the numbers of modifications increase when an optimum design is explored. Updating the design information in a centralized 3D model compared to multiple sets of 2D drawings make more sense. Also, the probability to have consistent data is much higher in a 3D model because the information is not duplicated. The benefits of a working product model concept are undeniable.

2. FEA as a Part of Structural Design Process

Not having the design information in a 3D format is one of the reasons why finite element analysis (FEA) is not typically introduced in the very early stages of design though it acts as a validation tool in later design stages. FEA is many times the first and also the only reason for the creation of a 3D model in the early structural design phases. This means that the finite element (FE) model is possibly the only place where the design information is in 3D. Usually, the geometry and property information of FE models are too idealized to be fully utilized by other disciplines; therefore, the FE model cannot be considered the optimum archive for design information.

FEA is a very time consuming task, normally carried out as few times as possible, and serves more as a validation tool of the design. Therefore the first FEA is carried out rather late in the design process, Fig. 2. However, if 3D design information was available, it could dramatically reduce the time in creating the FE model and makes it possible to carry out the analysis earlier in the design process.

During the design process different types of FEA analyses are carried out. The main purpose of FEA in the early design stages is to make sure the global responses of the ship are within tolerable limits and to ensure the general arrangement of ship structures are reasonable. As stated before, the FE models needed for the global response often require a long time to create with general purpose FEA systems. Additionally, local models may have to be created to support the design and validation of critical structural details as well as areas of high stress concentration.

In many ship types (e.g. tanker and bulkers), only the cargo area is subjected to FEA to satisfy the requirements of Classification authorities. The remaining structural members are validated through prescriptive rules issued by Classification authorities. Due to the above mentioned reasons, new and innovative designs are not pursued as they tend to be labor intensive (i.e., very expensive). This situation could be changed if the FEA was carried out in a much shorter time; therefore, making it a more attractive undertaking even though Classification authorities are not requiring it.

It would lead to better design if FEA could be introduced earlier in the design process. The process would be even more efficient if the FE model was extracted from the most recent design information for every analysis. Often, the FE model mesh for the new analysis is based on the previous FE model mesh and it is updated according to the new design information. With a 2D approach this would be the only reasonable way of working, but if the 3D design information is available, it opens a new
opportunity. Fig. 3 illustrates the ideal process for different FEA. Whenever there is a need for a new analysis, the creation should start from the most recent design information than updating the previously defined FEA information. It might be feasible to update the FE model mesh manually in later design stages where changes for design are not that frequent, but this is hardly ever the case in the early design stages. If the extraction of the FE model mesh for the FEA is well supported, it reduces the time and possibility for human errors. It is always better if the design information is maintained in as few systems as possible and FEA should not increase the number of systems.

![Design Information Maturity / Time](image)

**Fig. 3:** The location of FEA activities in ship design process typically and ideally

### 3. Steps in the FEA Process in Detail

The overall FEA process can be decomposed into key activities and individually examined. These activities are introduced in Fig. 4 and discussed in detail in the following sections.

![Time budget for FEA process entities (relative to 2D based design approach)](image)

**Fig. 4:** FEA time budget

![The creation of FEA in different design approaches](image)

**Fig. 5:** The creation of FEA in different design approaches

#### 3.1 Modelling and Meshing

The ultimate goal of the modelling and meshing process is to generate an adequate and valid calculation mesh (i.e., FE mesh) for submittal to an FEA solver. A significant part of this process is to define the appropriate structural properties of the design at hand (e.g., material properties, plate
thickness, and cross-sectional quantities). The differences in the selected design approach (i.e., 2D vs. 3D) have a significant effect on the performance of the modelling and meshing work. These differences are introduced in the detail below, but simply stated, it means there is always a greater degree of human interaction when the FE model is created on the basis of 2D design information while a true 3D design approach has a less degree of human interaction, Fig. 5.

3.1.1 The 2D Design Approach

If the design information is stored in 2D drawings it is tedious to extract the information necessary to create the FE mesh. The work usually starts with the manual idealization process wherein structural details are left out, or their effect taken into account in the scantling property definition. This work is done "on paper", typically by drawing grid lines on top of the structural drawings that serve as guidelines to the modelling process and the creation of a valid FE mesh. Therefore, the geometry in the FEA system is started as an idealized 3D model in order to reduce the time in creating the FE mesh and to get as few degrees of freedom for the calculation. When the design is changed (a common event during any design process), it has to be updated manually within the FEA system. Therefore the modelling is often the most time consuming task in the FEA, Doig et al. (2009).

3.1.2 The 3D Design Approach

If the design information is stored in a 3D format, the potential to save time in the process of creating an FE mesh is enormous. The amount of time that can be saved depends mainly on the capabilities of the 3D design system rather than the FEA system. There are bigger variations on the capability support to generate FE meshes among the different 3D design systems than in the geometry modelling tools of FEA systems.

The idealization process can be automated when the design information is in a 3D format. Different methods for idealizing structural details are introduced in Doig et al. (2009) and Kurki (2010). The main idea is to keep the 3D design information in an as-built state and not to introduce any idealization in the modelling of the geometry there. Otherwise, the design information is not adequate for the other disciplines if simplifications are carried out.

There is an extra step, finalizing the mesh, in the FEA process when the mesh is generated from the 3D model however. This activity requires the designer to clear the errors produced by the automatic mesh generation process (e.g., eliminating bad nodes, elements, and connections), which is required to successfully run an analysis. Although this task is present when an FE mesh is created straight in a FE pre-processor, it typically occurs throughout the creation of geometry and the FE mesh. The time required to perform this work depends on the system used in this process. Often, the mesh is only created once on the base of the design information, because it is easier to update the FE model in the FEA system than to manually correct the mesh incorrectness produced by the poor output from the 3D design system. For instance, there are many tools to create FE models based on a general CAD files, but the quality of the mesh is not as good as in well-integrated systems. Also, the transfer of scantling and material property information for the finite elements is insufficient.

3.2 Applying Loads and Boundary Conditions

The correct application of loads is a critical factor to sound structural design assessment. In some ways, the correct application of loads is most important. In ship design, there are several common load "patterns" that need to be considered: e.g. lightship distributions, tank loading, dead loads, hydrostatic loading, and in some ship types, hydrodynamic loading. The effort required to complete this activity within the FEA process can also be a large and tedious task. Similar to the modelling and meshing activities described in the previous sections, the process of applying loads can benefit from the data captured in the development of a 3D product model. The following sections describe what typical data is required to compose the complete loading scenario and how the 3D design approach can make this a more efficient process.
### 3.2.1 The 2D Design Approach

There are a number of data sources the designer must first locate, check its relevance, and extract for purposes of loading the FE model properly. Common input data include: weight reports, tank capacity plans, existing stability analysis reports (or run files), and perhaps even vendor data for significantly large weight items. Once the designer has this data, it is their task to model these loads appropriately using the available capability of the chosen FEA system. Most general purpose FEA systems have a base level of common loading patterns to accommodate the varying load experienced by ships.

![Image of a ship](image)

**Fig. 6: Ship-specific loading**

There are fewer ship-specific FEA systems that facilitate the modelling of these common loading patterns found in ship design, Fig. 6. For example, Fig. 7 shows an FE model that has tank loading defined as well as localized deck loading. Hydrostatic loads are another common ship load that ship-specific FEA systems can easily define, Fig. 8. The task of defining the loads must continue until all aspects of loading are accounted for and the loads are in equilibrium resulting in sound distributions such as those shown in Fig. 6.

![Image of an FE model](image)

**Fig. 7: Manual loading pattern creation**
The final step for the designer prior to processing is to define the proper boundary conditions such that the possibility of rigid body motion is avoided while minimally supporting the structure so as not to influence the natural response of the ship. This is a well-known strategy to FEA analyst.

3.2.2 The 3D Design Approach

Similar to what was described in the Modelling and Meshing section, information stored in the 3D model also has the potential to save significant time in the process of creating loading scenarios. The amount of time that can be saved depends on the extent of the loading definition in the 3D design system and the ability of the FEA system to consume the loading data. Fig. 9 illustrates the data flow for loading information where the 3D definition can automatically be consumed by the FEA system. As the loading definition changes with the changing design, the loading information can be re-exported for FEA consumption. Next, the task of performing load equilibrium checks and defining boundary conditions must still be accomplished by the designer as does the process of creating the appropriate boundary conditions. Any time savings that can be realized for these tasks are a function of the FEA system’s capability.

3.3 Analysis and Post Processing

To perform structural assessment, it is first required to find the structural response of the design based on the defined loading scenarios. In this step, the FEA system (using finite element methodologies) must perform calculations to determine the ships deformations and stresses. FEA systems are
designed to present to the designer these computed deformations and stress. This usually entails the recovery of results from the FE model. Fig. 10 shows some example stresses that would be expected to be recovered from a ship FE model. Further, stress results are graphically plotted, which allows the designer to effectively post-process a given structural response.

3.4 Limit State Analysis

Structural design assessment does not end with deformation and stress assessment. Comprehensive structural assessment should include evaluating structural stability and load-carrying capacity. This includes the assessment of different types of structural failure: stiffened panel collapse failure modes, local member failure modes, and hull girder ultimate strength.
The following are specific examples of six failure modes, Fig. 11:

- Mode I: Overall collapse after overall buckling
- Mode II: Collapse of the plating between stiffeners without their failure
- Mode III: Beam-column type collapse of a stiffener with attached plating
- Mode IV: Local buckling of stiffener web
- Mode V: Flexural-torsional buckling of a stiffener
- Mode VI: Gross yielding

3.5 Providing Results to Product Model

After conducting the finite element analysis, limit state analysis, and post-processing the results, the designer can revise the scantlings in the 3D product model. These changes in the structural arrangement and scantling definition can then be rerun through the FEA process as described in the previous sections. This feedback loop, within the context of a 3D design approach is how the FEA process becomes more active in the earlier phases of the structural design. When the structural design is adequate and sufficiently optimized to meet the objectives of the owner, the next step is to produce a complete set of structural drawings (i.e., the scantling plans) suitable for submittal to a Classification authority. At this juncture in the design process, the updated 3D product model serves as the source for creating these 2D drawings. This leads to a remarkable savings in developing Class drawings.

4. FEA Process Supported by NAPA and MAESTRO Software

NAPA, Naval Architectural Package, is a design tool specializing mainly in the early design stages of ship design process. NAPA contains a wide range of design solutions with the topological 3D product model as the core. The structural design tools, NAPA Steel, has been developed solely for the initial and basic design phases offering functionalities for multiple disciplines of which FEA is of main interest in this paper. Similar to NAPA, MAESTRO is used during early stage ship structural design. MAESTRO is a design, analysis, and evaluation tool specifically tailored for floating structures and has been fielded as a commercial product for over 20 years and has a world-wide user base. MAESTRO’s history is rooted in rationally-based structural design, which is defined as a design directly and entirely based on structural theory and computer-based methods of structural analysis (e.g., finite element analysis). MAESTRO core components are: rapid coarse-mesh finite element modeling, ship-based loading, finite element analysis, limit state analysis (e.g., at the hull girder level, stiffened panel level, and local member level), and design evaluation.

![Fig. 12: The summation of time spent in different FEA process entities](image)

Another fact that makes NAPA-MAESTRO combination interesting is the cooperation between the companies, Napa Ltd and DRS Defense Solutions LLC, who are developing these ship design specific tools. This cooperation has resulted in an interface between NAPA and MAESTRO that will significantly shorten the overall time in the FEA process compared to many other current market solutions. The efficiency gained through this interface is described in the detailed in the following chapter for each individual FEA process entities. Fig. 12 shows a summary of the spent time.
4.1 Modelling and Meshing

The creation of FE model in NAPA is based on a process where the start point is the real, as-built, representation of ship structures, Fig. 13. The first task is naturally to create the 3D model of ship structures. Usually, this work is done for other purposes therefore actual modelling work is minimal for the FEA. However, if the 3D model does not exist NAPA Steel could be used to create the model from scratch solely for the FEA purposes. The modelling tools have been proven to be very efficient in NAPA and the model can be created in a matter of days to accurate enough for the global FEA.

4.1.1 Idealization

The idealization process is done on the base predefined set of parameters in NAPA. The user is able to modify the values and store them as individual sets of rules. By applying different values to the rules various kinds of FE models can be generated from the same 3D structural model. The user is simply applying rules to get different detail level of FE model and not conducting any modelling work.

The parameters and rules define two main components; which structural details are considered in the FE model and how they are considered. An example of the rules is illustrated in Fig. 14. Different idealization methods are well introduced in Doig et al. (2009) and the idealization capabilities of NAPA in Kurki (2010).

One of the main advantages of having the mesh created inside the same tool as the 3D design information is to be able to have full control on the topology of the geometry. This will make the idealization and mesh generation more robust and offers better possibility make simplifications correctly compared to finding the connections between geometry on the base of a general purpose CAD output.

It is very efficient to create different kind of FE models when the generation is done by applying a set of rules for a product model. There is no need to create new, more detailed geometry appropriate for the target analysis though different representation of structures can be extracted on the fly by following the user defined rules in NAPA system, Fig. 15.
For instance, the stiffeners can be described the following ways depending on the target analysis:

1. Taken into account in the properties as lumped stiffeners where the influence of neglected stiffeners are merge to beam elements on the element boundaries
2. As beam elements, line segments with cross sectional properties
3. Web as surface elements and flange as beam elements
4. Web and flange as surface elements

In a local analysis one typical approach in the mesh generation is to define the area of interest with small elements and the surroundings with more coarse mesh to reduce the size of the model and to get better representation of the global behavior in the analysis. To reduce the time in creating such models NAPA has capabilities to define a different set of rules to limited area where the idealization and the mesh size differ from the surroundings. This will enable to create refined areas to any selected place in the 3d model without additional modeling worked to be carried out. The examples of refined model are illustrated in the Fig. 16.
4.1.2 Properties for the FE Model

If not as tedious job as creating geometry of the mesh at least equally important task is to define the properties for the FE mesh. In case of a global FEA, the idealization plays an important role i.e. a lot of structures are neglected as such, but their effects are taken into account. The options and parameters for deriving properties are illustrated in Fig. 14. Naturally, the actual properties of the 3D design information can be automatically inherited to the FE model reducing the time for creation of the properties significantly.

The connection to the structural design information is very important in order to have the latest property information available. The compartment information in the ship is also very important to have for reducing the compartment information can be utilized especially in the loading, which is described in more detail on the following chapter. The information on compartments is also important when applying the properties for the mesh. Typically, the design information is presented as gross scantlings whereas the FEA is carried out often with net scantlings i.e. gross scantlings deducted by the corrosion addition. The calculation of corrosion addition is heavily based on the information on compartments and especially on their contents. NAPA model has the information on compartments and their contents making it possible to derive the net scantling information for the mesh automatically.

4.2 Finalizing the Mesh

In the early design stages the 3D design information accuracy and the correctness of the geometry is not always sufficient to generate flawless meshes automatically. The more the mesh generation is based on a topological 3D model these imperfections can be corrected in the idealization process. In case of incorrect geometrical information it is better to try to correct the errors in the model as early stage as possible in the FE model creation process, Fig. 13. Here are the different options for finalizing the mesh in a recommendable order:

1. **Modify the 3D design information.** If the geometry is wrong producing bad quality mesh it should be corrected into the 3D NAPA model. Then it is also available for other design disciplines. FEA is recognized as a good tool for validating the design information.
2. **Define additional information to produce better quality meshes.** The user can define additional helping lines to NAPA structural model to guide the automatic meshing to produce better or more desired results. For instance, new traces can be modelled to topology to be
used in the in the element generation only. It is good to introduce these in the topology level as they can be utilized again when the 3D design information is changed and new FE models are needed.

3. Correct the resulting mesh manually. The error in the mesh can be corrected manually in MAESTRO.

4.3 Loads and Boundary conditions

MAESTRO’s loading capability addresses both general loading patterns as well as ship-specific loading patterns. The following are some specific loading capability found within MAESTRO’s existing system:

1. **Tank Loading.** Using the existing FE mesh definition, elements are collected to form the tank boundary. With the tank boundary defined, the designer can specify the tank contents and the amount of content found within the tank. The tank loading can be different for different loading scenarios.

2. **Hydrostatic Loading.** The hull definition is deemed wetted in MAESTRO terminology (see figure) and has the ability to be automatically loaded with hydrostatic loading. The definition of wetted elements, within the FE model, greatly facilitates the application of different still-water and wave conditions experienced by ships. This automatic application of hull pressure also plays an important role in properly finding force equilibrium for a given loading scenario.

3. **Longitudinal Distribution.** Achieving the correct lightship distribution can be accomplished by defining a known weight density or weight at defined longitudinal locations. Further, this definition allows the designer to define the transverse and vertical center of gravity for the total weight distribution to achieve proper nodal distribution.

Napa and DRS AMTC have collaborated to extract the pertinent loading information from the 3D product model and translate it to the corresponding MAESTRO loading capability described above. Currently, the loading data includes: longitudinal weight distributions, longitudinal bending moment distributions, hull definition for hydrostatic loading, Fig. 17, tank boundary definitions, Fig. 18, tank content and fill definitions, and hydrostatic equilibrium definition (i.e., trim and heel).

Fig. 17: NAPA hydrostatic loading data to MAESTRO
4.4 Response Analysis, Limit State Analysis, and Post Processing

MAESTRO has the ability to perform comprehensive structural assessment for floating structures. This includes performing response analysis (i.e., deformation and stress analysis) and limit state analysis. The limit state analysis includes hull girder collapse analysis, stiffened panel buckling analysis, and local member buckling analysis.

The first step to structural assessment is conducting response analysis. This encompasses the computation of deformations and stresses. MAESTRO’s response analysis has been verified against theoretical and other industry standard FEA software results. MAESTRO’s FEA solver uses the Intel Pardiso Sparse solver, which is a high-performance, robust, memory efficient, and easy to use solver for solving large sparse symmetric and non-symmetric linear systems of equations on shared memory multiprocessors. Deformation and stress can be recovered from individual elements as well as stiffened panels, Fig. 10.

The next step in structural design assessment, limit state analysis, has been a core component to MAESTRO from its inception. MAESTRO has a comprehensive structural assessment capability and includes the evaluation of structural stability and load-carrying capacity. The formulation of MAESTRO’s limit state analysis is covered in Hughes and Paik (2010) and Paik and Thayamballi (2003). These textbooks constitute the theoretical manual for MAESTRO’s limit state analysis. MAESTRO’s limit state analysis capability computes a number of different stiffened panel collapse failure modes, local member failure modes, and hull girder ultimate strength, including the six modes of failure previously described and illustrate in Fig. 11. MAESTRO’s limit state analysis is done automatically and comprehensively for the entire FE model and for all loading conditions. To properly perform this strength assessment, the true stiffened panel must be found and assessed in the FEM. This is done by automatically searching the entire model and collecting multiple finite elements (plates or beams) so the true boundary conditions and true spans are represented, Fig. 19.
4.5 NAPA-MAESTRO Interface Summary

Using a 2D approach is certainly one way to build a 3D FE model; however, this interface provides a more efficient method by leveraging a 3D approach. Although tools like FEMAP, PATRAN, etc., all offer the capability to build a 3D FE mesh using a 3D surface model, what makes the NAPA 3D model unique is its 3D surface model’s tight coupling to the 3D product model, its capability of structural idealization, its ability to generate different FE mesh models from the same 3D product model to support different analyses, and the linking of the NAPA hydrostatic model.

Combining this technology with a tool like MAESTRO has a great potential to improve the efficiency of the structural design process and brings FEA more so to the early stage ship structural design, analysis, and evaluation process. It does so by allowing the designer to leverage one 3D model from start to finish within the scope of structural design and direct analysis activities. This will eliminate the very common practice of recreating 3D structural models to serve different activities (e.g., one 3D model for Classification drawings and one 3D model for structural analysis). Further, by interfacing these two products, the designer does not have to recreate key loading scenarios in different products.

At the core of the interface is the MAESTRO Neutral File, which contains the NAPA generated data that is pertinent for creating and analyzing the MAESTRO finite element model. Currently, Napa and DRS AMTC have successfully translated all of the finite element mesh and scantling information (e.g., unit system, FE nodes, material properties, and finite elements). Added to this, Napa and DRS AMTC have also been able to translate the pertinent loading information, which makes this interface unique. The loading data will include: longitudinal weight distributions, longitudinal bending moment distributions, hull definition for hydrostatic loading (i.e., the wetted elements in MAESTRO terminology), tank boundary definitions, tank content and fill definitions, and hydrostatic equilibrium definition (i.e., trim and heel).

4.6 Future Development Topics in NAPA-MAESTRO Interface

The current version of the interface supports well the FE model creation process in one direction i.e. NAPA pushes a lot of information to MAESTRO where the actual response of the structure is evaluated. Currently, NAPA also pushes pertinent loading information for consumption by MAESTRO. In order to support the design process better the information generated in the FEA should be fed back to the design information. This should now be done manually.

All the tasks that are carried out in the FEA consisting manual work is under investigation. For instance, it helps the handling of a large FE model if the elements are grouped. Certain groups are already now created, but new groups and other similar supporting information is under considerations to be included in the interface.

5. Conclusions

There are many advantages maintaining design information in a 3D model throughout the whole design process. This is even more emphasized when FEA is closely present in the design activities. Many times the FEA is carried out only in the mandatory cases because it is considered as tedious job. This will lead to designs reliant on the previous knowledge. With combination of NAPA and MAESTRO the FEA can be carried out in much shorter time enabling it to be used in the earlier design stages giving confidence that new innovative designs are functional.

Acknowledgement

This paper is cleared for open publication: Ref. DRS Defense Solutions Case # DRSAMTC-11002.
References


HUGHES, O.F.; PAIK, J.K. (2010), Ship Structural Analysis and Design, SNAME


<table>
<thead>
<tr>
<th>Author</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>Abels</td>
<td>88</td>
</tr>
<tr>
<td>Ahlers</td>
<td>109</td>
</tr>
<tr>
<td>Andrews</td>
<td>426</td>
</tr>
<tr>
<td>Antonov</td>
<td>31</td>
</tr>
<tr>
<td>Aulinas</td>
<td>171</td>
</tr>
<tr>
<td>Bade</td>
<td>369</td>
</tr>
<tr>
<td>Barth</td>
<td>195</td>
</tr>
<tr>
<td>Bertram</td>
<td>8</td>
</tr>
<tr>
<td>Bibuli</td>
<td>74</td>
</tr>
<tr>
<td>Bralić</td>
<td>317</td>
</tr>
<tr>
<td>Bruinessen</td>
<td>397</td>
</tr>
<tr>
<td>Bruns</td>
<td>466</td>
</tr>
<tr>
<td>Bruzzone</td>
<td>74</td>
</tr>
<tr>
<td>Burger</td>
<td>347</td>
</tr>
<tr>
<td>Cabos</td>
<td>501</td>
</tr>
<tr>
<td>Caccia</td>
<td>74</td>
</tr>
<tr>
<td>Cage</td>
<td>317</td>
</tr>
<tr>
<td>Caiti</td>
<td>181,377</td>
</tr>
<tr>
<td>Calabrò</td>
<td>181,377</td>
</tr>
<tr>
<td>Caprace</td>
<td>271</td>
</tr>
<tr>
<td>Christensen</td>
<td>109</td>
</tr>
<tr>
<td>Christiansen</td>
<td>466</td>
</tr>
<tr>
<td>Conte</td>
<td>379</td>
</tr>
<tr>
<td>Couser</td>
<td>206</td>
</tr>
<tr>
<td>Danese</td>
<td>221</td>
</tr>
<tr>
<td>Daniels</td>
<td>19</td>
</tr>
<tr>
<td>DeNucci</td>
<td>397</td>
</tr>
<tr>
<td>Diez</td>
<td>245</td>
</tr>
<tr>
<td>Dorri</td>
<td>31</td>
</tr>
<tr>
<td>Drap</td>
<td>379</td>
</tr>
<tr>
<td>Drikos</td>
<td>74</td>
</tr>
<tr>
<td>Ehlers</td>
<td>283</td>
</tr>
<tr>
<td>Eich</td>
<td>74</td>
</tr>
<tr>
<td>Eisen</td>
<td>501</td>
</tr>
<tr>
<td>Erikstad</td>
<td>237</td>
</tr>
<tr>
<td>Faller</td>
<td>474</td>
</tr>
<tr>
<td>Fischer</td>
<td>109</td>
</tr>
<tr>
<td>Frank</td>
<td>317</td>
</tr>
<tr>
<td>Friedewald</td>
<td>253</td>
</tr>
<tr>
<td>Garcia</td>
<td>171</td>
</tr>
<tr>
<td>Georgoulas</td>
<td>335</td>
</tr>
<tr>
<td>González</td>
<td>134</td>
</tr>
<tr>
<td>Grammatico</td>
<td>181</td>
</tr>
<tr>
<td>Hamel</td>
<td>442</td>
</tr>
<tr>
<td>Harries</td>
<td>206,411</td>
</tr>
<tr>
<td>Hefazi</td>
<td>442</td>
</tr>
<tr>
<td>Heinig</td>
<td>253</td>
</tr>
<tr>
<td>Hess</td>
<td>474</td>
</tr>
<tr>
<td>Holmberg</td>
<td>536</td>
</tr>
<tr>
<td>Hopman</td>
<td>292,395</td>
</tr>
<tr>
<td>Horner</td>
<td>347</td>
</tr>
<tr>
<td>Hunter</td>
<td>536</td>
</tr>
<tr>
<td>Huuva</td>
<td>66</td>
</tr>
<tr>
<td>Ignatius</td>
<td>391</td>
</tr>
<tr>
<td>Jakobsen</td>
<td>305</td>
</tr>
<tr>
<td>Jungleowitz</td>
<td>369</td>
</tr>
<tr>
<td>Klanac</td>
<td>317</td>
</tr>
<tr>
<td>Kluwe</td>
<td>326</td>
</tr>
<tr>
<td>Koch</td>
<td>98</td>
</tr>
<tr>
<td>Koechert</td>
<td>188</td>
</tr>
<tr>
<td>Koelman</td>
<td>292,357</td>
</tr>
<tr>
<td>Koningh</td>
<td>292</td>
</tr>
<tr>
<td>Korbetis</td>
<td>335</td>
</tr>
<tr>
<td>Kotiranta</td>
<td>264</td>
</tr>
<tr>
<td>Koveos</td>
<td>74</td>
</tr>
<tr>
<td>Kraus</td>
<td>195</td>
</tr>
<tr>
<td>Kroffke</td>
<td>109</td>
</tr>
<tr>
<td>Krömer</td>
<td>501</td>
</tr>
<tr>
<td>Larkins</td>
<td>523</td>
</tr>
<tr>
<td>Lemburg</td>
<td>109</td>
</tr>
<tr>
<td>Leshchina</td>
<td>31</td>
</tr>
<tr>
<td>Lladó</td>
<td>171</td>
</tr>
<tr>
<td>Lödding</td>
<td>253</td>
</tr>
<tr>
<td>Lukas</td>
<td>326</td>
</tr>
<tr>
<td>Luz</td>
<td>516</td>
</tr>
<tr>
<td>Makiyama</td>
<td>516</td>
</tr>
<tr>
<td>Martins</td>
<td>271</td>
</tr>
<tr>
<td>Mesing</td>
<td>326</td>
</tr>
<tr>
<td>Mikalsen</td>
<td>124</td>
</tr>
<tr>
<td>Morais</td>
<td>523</td>
</tr>
<tr>
<td>Mrykin</td>
<td>31</td>
</tr>
<tr>
<td>Munafò</td>
<td>181</td>
</tr>
<tr>
<td>Nagel</td>
<td>148</td>
</tr>
<tr>
<td>Nieuwenhuis</td>
<td>195</td>
</tr>
<tr>
<td>Nishimoto</td>
<td>516</td>
</tr>
<tr>
<td>Oers</td>
<td>397</td>
</tr>
<tr>
<td>Ortiz</td>
<td>74</td>
</tr>
<tr>
<td>Parker</td>
<td>19</td>
</tr>
<tr>
<td>Pawling</td>
<td>426</td>
</tr>
<tr>
<td>Penas</td>
<td>134</td>
</tr>
<tr>
<td>Peri</td>
<td>245</td>
</tr>
<tr>
<td>Peric</td>
<td>8</td>
</tr>
<tr>
<td>Petersen</td>
<td>305</td>
</tr>
<tr>
<td>Petillot</td>
<td>171</td>
</tr>
<tr>
<td>Pettersson</td>
<td>66</td>
</tr>
<tr>
<td>Renard</td>
<td>56</td>
</tr>
<tr>
<td>Rigo</td>
<td>271</td>
</tr>
<tr>
<td>Rosow</td>
<td>466</td>
</tr>
<tr>
<td>Salvi</td>
<td>171</td>
</tr>
<tr>
<td>Santos</td>
<td>488</td>
</tr>
<tr>
<td>Sato</td>
<td>516</td>
</tr>
<tr>
<td>Scaradozzi</td>
<td>379</td>
</tr>
<tr>
<td>Schleusener</td>
<td>253</td>
</tr>
<tr>
<td>Schmitzener</td>
<td>442</td>
</tr>
<tr>
<td>Singer</td>
<td>19</td>
</tr>
<tr>
<td>Solem</td>
<td>237</td>
</tr>
<tr>
<td>Steinhauer</td>
<td>453</td>
</tr>
<tr>
<td>Stifani</td>
<td>181</td>
</tr>
<tr>
<td>Thomson</td>
<td>156</td>
</tr>
<tr>
<td>Tillig</td>
<td>206,411</td>
</tr>
<tr>
<td>Trevisani</td>
<td>271</td>
</tr>
<tr>
<td>Veelo</td>
<td>359</td>
</tr>
<tr>
<td>Vögele</td>
<td>74</td>
</tr>
<tr>
<td>Waldie</td>
<td>523</td>
</tr>
<tr>
<td>Wilken</td>
<td>411,501</td>
</tr>
<tr>
<td>Winther</td>
<td>305</td>
</tr>
<tr>
<td>Zanoli</td>
<td>379</td>
</tr>
<tr>
<td>Zaraphonitis</td>
<td>411</td>
</tr>
</tbody>
</table>
11th Conference on
Computer Applications and Information Technology in the Maritime Industries
COMPIT'12
Liege/Belgium, 16-18 April 2012

Topics: CAX Techniques / Virtual Reality / Simulations / Robotics /
Web Technology / Artificial Intelligence / Decision Support Systems /
Ship design methods and tools
In Design, Production and Operation of Maritime Systems

Organiser: Volker Bertram (volker.bertram@GL-group.com)

Advisory Committee:
Lanfranco Benedetti   CESA, Europe
Volker Bertram FutureShip, Germany
Marcus Bole AVEVA, UK
Andrea Caiti Univ Pisa, Italy
Stein Ove Eriskstad NTNU, Norway
Carlos Gonzalez SENER, Spain
Stefan Harries Friendship Systems
Darren Larkins ShipConstructor, Can
Yvonne Masakowski NWC, USA
Igor Mizine CSC, USA
Daniele Peri INSEAN, Italy
Philipp Rigo ANAST, Belgium
Bastian Veelo SARC, Norway
Richard Vogelson ANR Global, USA

Venue: The conference will be held in the Ramada Plaza Hotel:

Format: Papers to the above topics are invited and will be selected by a selection committee. There will be hard-cover black+white book with ISBN produced in addition to the online pdf version in colour. Papers may have up to 15 pages.

Deadlines: < 15.12.2011 Optional “early warning” of intent to submit paper
21.12.2011 First round of abstract selection (1/3 of available slots)
15.1.2012 Final round of abstract selection
08.3.2012 Payment due for authors
08.3.2012 Final papers due (50 € surcharge for late submission)

Fees: 600 € / 300 € regular / PhD student
Refund¹: 100 € / 50 € regular / PhD student for payment received by 8.3.2012

Fees are subject to VAT reverse charge mechanism
Regular fees include proceedings, lunches and coffee breaks and conference dinner
PhD fees include lunches and coffee breaks

Sponsors: AVEVA, GL Group, NAPA, SENER, ShipConstructor Software
Information: www.compit.info

¹ A discount for COMPIT 2013 is given.