

Virtual Prototyping System for Maritime Crane Design and Operation Based on Functional Mock-up Interface

Yingguang Chu, Lars Ivar Hatledal, Filippo Sanfilippo, Vilmar Æsøy, Houxiang Zhang
Department of Maritime Technology and Operations
Aalesund University College
Aalesund, Norway
{yich, laht, fisa, ve, hozh}@hials.no

Hans Georg Schaathun
Department of Engineering and Natural Sciences
Aalesund University College
Aalesund, Norway
hasc@hials.no

Abstract—This paper presents the framework of a virtual prototyping system for the design and simulation of maritime crane operations. By combining the rapid-prototyping approach with the concept of interchangeable interfaces, different demanding operation scenarios can be simulated including models of the corresponding physical systems, the vessel and the surrounding environment. Multiple tradeoffs and alternative solutions can be evaluated during the design phase. This process can be achieved within a short time period, allowing for the reduction of lead-times as well as for the abatement of mistakes or system failures that may otherwise cause fatal accidents in real tests. In addition, the virtual simulator can also be used for training purposes allowing for a substantial improvement in working efficiency and operation safety. The software architecture of the proposed framework is based on the application of the Functional Mock-up Interface standard. This utilizes the current available modelling tools and allows for the exchange of dynamic models and for co-simulation of different models according to the current designing needs. The development of the framework and involved modules of maritime crane systems are described. Preliminary simulations are presented to show the effectiveness and flexibility of the proposed framework.

Keywords—*virtual prototyping, maritime crane, Functional Mock-up Interface.*

I. INTRODUCTION

Maritime cranes are widely used as important sub-systems to handle and transfer objects from large container ships to smaller lighters or to harbor quays. Crane operations are challenging, and involve many problems such as load sway, positioning accuracy, suppression, collision avoidance, and manipulation security. The current crane designing process is still carried out in a traditional way, which lags behind the changing requirements within a short time span. When considering both work efficiency and operation safety, the current crane design process is far from optimal. As a result, maritime cranes are designed heavier, stronger and bigger than necessary in order to meet the requirements regarding working space, lifting capability, operational efficiency and redundancy, and manipulation security.

The simulation of maritime cranes relies on complex models of the physical systems as well as an equally complex model of the operational environment. Efficiency, flexible performance, operational safety, environmental issues and cost targets are urgent topics to be tackled in the new generation crane design system. The main challenge is to be able to develop and configure realistic models within short time frames, thus, evaluating multiple design concepts can be done effectively, where potential trade-offs and alternatives can be evaluated within a short time period.

The goal of the current research is to develop a Virtual Crane Prototyping (VCP) framework for overall crane design and simulation of operations, including mechanical sub-system, control sub-system, hydraulic sub-system, and verification of the operational performance as a part of the design process. The emphasis and objective of this paper is on the development of a standard VCP framework for maritime crane design and operations. The rest of the paper is organized as follows. Section II presents the state of the art of virtual prototyping approaches for industrial engineering systems. Section III introduces the development of the VCP framework and its key features. Part of the preliminary results of the implementation of the framework for maritime crane design and operations is shown. At last, the conclusion and future work are given.

II. STATE OF THE ART

Since the last few decades, many good tools were used for modelling and simulating physical systems domain-specifically, for example, Flexcom3D for structure FEA, Adams for multi-body dynamics, PSCAD for the power systems, dSPACE for control systems, and GT-Suite for engine systems. These simulation software tools have proved of significant values in the design and development of components and sub-systems. Complex engineering systems such as a maritime crane comprises components and sub-systems from many different domains, each with their own modelling methodology and software preference. Specialist knowledge is required in order to efficiently use the analysis capabilities of domain-specific software and the software packages are predominantly “stand-alone” and not capable of

This project work is supported by the Research Council of Norway via Innovation Projects for the Industrial Sectors “MAROFF”.

interconnection. To interface these sub-modules is non-trivial for the design process of product and system design.

The aviation and defense industry has developed advanced methods for both design and operation driven by customer and regulatory requirements. This includes the use of simulation as a design tool, as well as for operator training by implementing simulation models from a distributed collection of models and simulator sites. The maritime industry has partially adopted the results of this process, mostly for crew training. Existing training simulators are mostly related to the maneuvering of ships for the Dynamic Positioning (DP) system or at a sub-system level to learn to use a special piece of equipment decoupled from the total system and even a crane operation simulator [1], [2]. The use of the simulators for product and system design and analysis however is rather limited. A few case studies were presented in the last few years, and solved certain parts of the problem using individual approach [3], [4].

Recent advances in virtual prototyping have been significant. The automotive industry has been particularly innovative, with projects such as the European Modelisar project, which leads to the Functional Mock-up Interface (FMI) among other things [5]. A major insight is that the plethora of modelling tools within different disciplines exists for very good reasons, and there will never be one tool that can be perfect for every stage and every branch of the design process. In order to reuse and exchange the existing models, the solution lies in shared, standardized interfaces (e.g. FMI), and model transformations on shared format (e.g. XML/C-code). Hence, component sub-models can be developed in different tools depending on their disciplinary and the designers' preferences. Model integration, or rather modelling of systems of systems, requires a separate tool or an integration platform.

There are still challenges to be tackled with domain-specific solutions. For instance, FMI tells us how to specify the parameters and variables to exchange and how to exchange them. However, the definition of variables is domain dependent, and domain sub-models and their meta-models must be developed considerably to ensure compatible components. The integration platform and system simulation must also be tailored to the domains to provide an authentic, virtual testing environment. A good example of such an integration platform is the CarMaker [6]. However, CarMarker is still far from maritime requirements as such a simulator is intended to be used for both design and operational simulation. Currently maritime operations are becoming more and more demanding. Heavy lifting and handling at depths of several thousand meters, precise installation of subsea modules weighing several hundred tons, platform support in the ice and the cold of the northern regions, etc. The complexity increases even further when taking into account the fact that these operations require a much greater coordination between professionals, for example during ship maneuvering and crane, winch and ROV operations. The operational performance has to be considered in the designing phase.

The solution to the integration of maritime crane design and operation simulation as a whole system is not simple. There are several gaps between the current simulation tools and the urgent industrial demands in crane design. The simulation tools

for sub-systems have different focuses, which make them special. Generally, there isn't any software tool that is able to bring the whole crane design process and operation performance together. There is a need for standardization of what constitutes the components of a maritime crane and their interfaces to allow implemented models to be re-used in different simulation settings, be it design, operation or training.

III. VIRTUAL CRANE PROTOTYPING FRAMEWORK

The FMI is a standard for tool independent implementation of executable simulation modules. Since the first version released in 2010, the standard has been well received in the simulation world and quickly adopted in industry, in particular, the automotive industry. FMI version 2.0 released in July 2014 is currently supported by over 35 tools all over the world. The FMI standard support both model exchange and co-simulation of dynamic models. The main difference is that co-simulation requires the FMU to be complete with a numeric solver, while a FMU for model exchange only implements the model itself, relying on the solver of the importing tool. A Functional Mock-up Unit (FMU) comprises an XML-file describing the dynamic variables to be exchanged, and the simulation model either as C source code or a compiled linked library (or both).

Although most modelling tools offer a range of interfaces to support co-simulation and interaction with control systems, it does not make system integration easy. In most cases, the integration of complex system simulators depends on a lot of manual work and ad hoc solutions. A standardized exchange interface, such as FMI, is more flexible, and component models from a large number of different tools can be exchanged easily. The proposed VCP framework uses JavaFMI [7] to import FMUs for co-simulation. Thus component models can be integrated from tools that are able to export FMU.

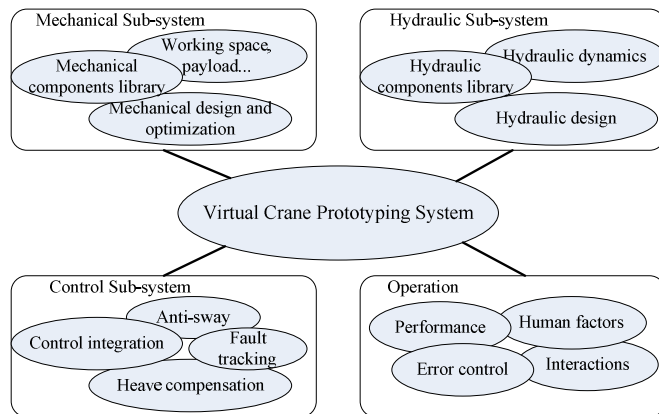


Fig. 1. VCP simulator model structure

First of all, to achieve the proposed VCP simulation system, generic models for all key structures, parts and sub-units of maritime cranes was defined, Fig. 1. The key features of the VCP simulator consist of:

- VCP simulator high level layer: include parameter definition, system performance metrics (energy consumption, payload, system dynamics, system costs, operational safety and efficiency etc.)

- Model library: library of generic models of all key maritime crane systems including main machinery systems (power supply, actuation, transmission machinery), auxiliary systems (support systems, mechanical structures) and control systems.
- Variant customer-orientated design: allows easy adjustment of geometric configurations, DOFs, link lengths, cylinder positions and dimensions, etc.
- Control box: contains non-traditional control suits designed for flexible control algorithm, heave compensation and anti-sway functions.
- User interfaces: include haptic input device for control, visual demonstrations and feedback with graphical interface showing key features and results (system metrics, charts) for specific cases and examples. Demonstrate areas of future application and identify implications and requirements for further research.

A. The software architecture

A general, high-level architecture is shown in Fig. 2. It follows a standard Model-View-Controller (MVC) pattern, which dictates a separation with low coupling between the logic (model), the presentation (view), and the input (controller) [8]. This is important to be able to reuse simulation models with different views, and to use visualization code with different models. The user interface is the most important example of views and controllers. The views must be portable and accessible from many different platforms, including different operating systems and mobile devices. This is facilitated by using Web technologies. Most importantly, it offers 3D visualization using Web Graphics Library (WebGL), which turns the web browser into a powerful visualization vehicle. WebGL is a cross-platform web standard for a low-level 3D graphics API based on OpenGL ES 2.0, and allows GPU accelerated usage of physics and image processing. The prototyping system described in this paper uses the popular and open-source JavaScript 3D Library (three.js) to render the graphics [9]. A second webpage offers a controller, where the user can set input signals, and also view result data from the simulated system. Further views and controllers can be added as needs arise.

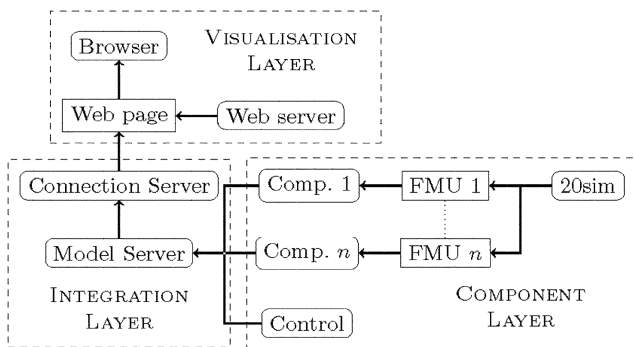


Fig. 2. High level architecture of the framework in MVC pattern

The simulator model is divided into two layers. The component layer contains separate simulators for different

components and sub-systems, while the integration layer implements the system model comprising the different components and sub-systems. The components are typically created in a modelling tool and exported as a Functional Mock-up Unit (FMU), but they may also be implemented using general purpose programming languages or other tools. The integration layer implements the system model and acts as a master node for co-simulations, with the co-simulation slaves running in the component layer.

In the component layer, we use a thin Java wrapper around the FMU to export the functionality via Remote Method Invocation (RMI). Components or sub-systems (such as the control system in our prototype) can be implemented directly in Java mimicking the FMU interface over RMI.

We have developed a prototype to demonstrate the architecture. The main purpose of the prototype is to demonstrate the loose coupling between the 3D visualization in WebGL and the model which is implemented in Java where FMUs and other components are imported. Bi-directional communication with the visualization layer was implemented using the WebSocket protocol. A similar case is presented by Pang on the development of an eLearn platform for the study of high performance, energy efficient building design, construction and operations [10].

B. The component layer

- Mechanical sub-system

3D model of the crane was built in CAD software such as Solidworks, NX, etc. Finite Element Analysis (FEA), static analysis can be achieved via these programs. For integration with the virtual prototyping system, the objectives regarding the mechanical part include: firstly, parameterization and configuration of a crane for the study of work space and static load capacity, typically in the concept design phase; secondly, kinematic and dynamic models should be included for the study of control algorithms and operational performances; at last, 3D models provide graphical data for visualization.

A workspace computation and visualization method is proposed and presented [11]. Static forces to the joints and actuators are calculated simultaneously. A dynamic model of the crane is developed and presented [12]. The model can be exported as a separate FMU for co-simulation in the proposed VCP system. One alternative for 3D visualization is to export CAD models as COLLADA files then import to WebGL.

- Hydraulic sub-system

Maritime cranes are mostly fitted with hydraulic power unit to achieve a high operational capacity. Modelling of the hydraulic system was done in 20sim based on Bond Graph (BG) method [13]. BG method is a modelling approach based on identifying the energetic structure of the system. A physical system can be decomposed into several basic properties and represented by the interrelated idealized elements that represent these properties. The interaction in between is called the energy or power bond. A beta version is supported for FMU export for co-simulation. Other tools are possible for use as long as they support FMU exportation for co-simulation.

- Control sub-system

In order to achieve safer operations of maritime cranes in adverse working environment, intelligent control algorithms including heave compensation and anti-sway functions were proposed [14]. Cranes that are simulated in the proposed VCP framework can be controlled by the user in a variety of different ways. This includes the direct joint-by-joint control as well as alternative inverse kinematics algorithms for position or velocity control. Heave compensation and anti-sway algorithms are based on the inverse kinematics algorithm. The principle is to move the crane tip to the opposite of the wave and load sway. The control algorithms were implemented as modules directly in Java and exposed over RMI similarly to the FMUs, allowing the control algorithms, which might be CPU intensive, to run as a separate process or on a different host. As with the FMUs, new control algorithms can be added to the system for testing and evaluation [15].

C. The integration layer

The integration layer manages a system model composed of several component models, each of which has a certain level of complexity in its own right. The integration layer was implemented entirely in Java, based on a simple and generic model of a maritime crane. The kinematic structure of the crane was modelled as part of a scene graph describing the parent-child relations of all objects included in the simulation. The behaviors of the objects in the scene graph can be implemented either in FMUs or directly in the Java code.

D. The visualization layer

By means of WebGL technology, users can view the simulated models in a WebGL enabled browser without having to install any additional software. As the simulation is running on a centralized server, the users need not a particularly powerful computer to interact with it, even though the simulation may be of a very complex nature. Any 3D models supported by three.js can be visualized. A WebGL scene of the crane was implemented in the VCP framework, Fig. 3. The scene was updated accordingly as the position data of each object in the scene pulled from the server. The workspace mesh of the crane operation is shown.

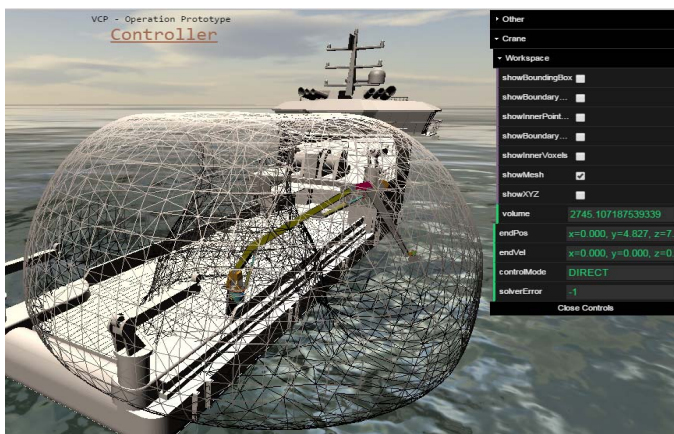


Fig. 3. A WebGL scene of the crane operation and its workspace

IV. CONCLUSION AND FUTURE WORK

In the above sections, the development of the framework of a proposed virtual crane prototyping system was introduced. The interfacing and integration is based on the FMI standard. To achieve a generic solution for such a system that suits for design and operational simulation, the model structure of the framework was separated into three layers. Involved system modules and their meta-models were discussed. Results of preliminary implementation were presented. Future work includes establishing the framework as a standard format for implementation, developing component and system model libraries. Interfacing CAD design and simulation is challenging in terms of parameterization and 3D visualization.

REFERENCES

- [1] Offshore Simulation Center AS, Crane simulators, available at: <http://offsim.no/Products/Crane-Simulators#>, 2014.
- [2] Algoryx Simulation AB, Offshore cranes, available at: <https://www.algoryx.se/by-industry/offshore/>, 2014
- [3] N. Ku, S. Ha, and M. Roh, "Crane modelling and simulation in offshore structure building industry," *J. of Computer Theory and Engineering*, 6(3), 2014, pp. 278-284.
- [4] K. Terashima, R. Ito, Y. Noda, Y. Masui and T. Iwasa, "Innovative integrated simulator for agile control design on shipboard crane considering ship and load sway," in *Proc. of the IEEE Multi-Conference on Systems and Control*, Yokohama, Japan, September 2010, pp. 1293-1300.
- [5] Modelisar Association, Functional Mock-up Interface, available at: <https://www.fmi-standard.org/>, 2014.
- [6] IPG Automotive, Carmaker, available at: <http://ipg.de/simulationsolutions/carmaker/>, 2014.
- [7] JavaFMI, available at: <https://bitbucket.org/siani/javafmi/wiki/Home>, 2014.
- [8] T. Reenskaug, "The Model-View-Controller (MVC) Its past and present," University of Oslo, Oslo, Norway, 2013, available at: http://heim.ifi.uio.no/~trygver/2003/javazone-jaoo/MVC_pattern.pdf, 2014.
- [9] JavaScript 3D Library, three.js, available at: <http://threejs.org/>, 2014.
- [10] X. Pang, R. Dye, T. S. Nouidui, M.I Wetter, and J. J. Deringer, "Linking interactive Modelica simulations to HTML5 using the Functional Mockup Interface for the LearnHPB platform," in *Proc. of the 13th IBPSA Conference*, Chambéry, France, August 2013, pp. 2823-2829.
- [11] L.I. Hatledal, F. Sanfilippo, Y. Chu, and H. Zhang, "A voxel-based numerical method for computing and visualising the workspace of offshore cranes," in *Proc. of the 34th International Conference on Ocean, Offshore and Arctic Engineering*, St. John's, Newfoundland, Canada, May 2015, in press.
- [12] Y. Chu and V. Æsøy, "A multi-body dynamic model based on bond graph for maritime hydraulic crane operation," in *Proc. of the 34th International Conference on Ocean, Offshore and Arctic Engineering*, St. John's, Newfoundland, Canada, May 2015, in press.
- [13] Y. Chu, Ø. Bunes, V. Æsøy, and H. Zhang, "Modelling and simulation of an offshore hydraulic crane," in *Proc. of the 28th European Conference on Modelling and Simulation*, ECMS, Brescia, Italy, May 2014, pp. 87-93.
- [14] Y. Chu, F. Sanfilippo, V. Æsøy, and H. Zhang, "An effective heave compensation and anti-sway control approach for offshore hydraulic crane operations," in *Proc. of the IEEE International Conference on Mechatronics and Automation*, Tianjin, China, August 2014, pp.1282-1287.
- [15] F. Sanfilippo, L. I. Hatledal, Y. Chu, H. Zhang, and K. Y. Pettersen. "A benchmarking framework for control methods of maritime cranes based on the Functional Mock-up Interface," Submitted to the Elsevier Journal of Ocean Engineering, 2015.