Journal of Marine Science and Technology (2018) 23:754–766
https://doi.org/10.1007/s00773-017-0509-z

ORIGINAL ARTICLE

Virtual prototyping for maritime crane design and operations

Yingguang Chu[1,2] · Lars Ivar Hatledal[2] · Houxiang Zhang[2] · Vilmar Æsøy[2] · Sören Ehlers[3]

Received: 27 November 2015 / Accepted: 17 October 2017 / Published online: 25 November 2017

Abstract

This paper presents the implementation of the virtual prototyping system for maritime crane design and operations. The study is designed to bridge the following gaps in maritime crane system simulations. First, the virtual prototyping system introduces an open and flexible platform oriented to overall product and system design, modeling, simulation and visualization. Second, the virtual simulator for operations based on the proposed framework is reinforced by high fidelity models of physics and dynamics. The paper discusses the challenges in virtual prototyping of complex multi-domain systems from the perspectives of modern complex engineering system design, modeling and simulation of multi-domain dynamic systems, large set of data exchange for communication and visualization. The software architecture of the system is based on the application of the functional mock-up interface standard. This utilizes the current available modeling and simulation tools, and allows for the exchange and reuse of models. Simulations in a virtual environment permit the evaluation of multiple trade-offs and alternative solutions from early design stages. As a case study and verification, the knuckle boom crane systems were implemented. The virtual crane simulator proved the effectiveness of the proposed virtual prototyping system in solving the long-existing challenges in simulations of complex multi-domain systems.

Keywords Virtual prototyping · Maritime crane · Dynamics · Hydraulics · Visualization · Co-simulation

1 Introduction

Maritime cranes are the main deck machineries in offshore and subsea applications for transferring, lifting, and handling operations. Maritime crane design is an interdisciplinary process involving, e.g., mechanical, hydraulic, dynamic, and control-related tasks. Since the last few decades, more and more modeling and simulations have been involved throughout the product development and verification processes. However, compared to other engineering industries such as aerospace and automobile, maritime industry still lags behind in utilizing the computer-aided technologies for product and system design. This is partly due to the fact that modeling and simulations of the physics of these dynamic sub-systems are non-trivial tasks by themselves. What's more, the various preferences of the software tools favored by different users offer many challenges for system model integration [1]. To simulate the integrated models of the complete system is even more challenging, in particular, for real-time operational applications.

The increasing competitive context of the global market demands quality design within shorter time period at lower cost. However, the current crane design process is far from

✉ Yingguang Chu
yingguang.chu@ntnu.no

Lars Ivar Hatledal
laht@ntnu.no

Houxiang Zhang
hozh@ntnu.no

Vilmar Æsøy
vilmar.aesoy@ntnu.no

Sören Ehlers
ehlers@tuhh.de

[1] Department of Marine Technology, Norwegian University of Science and Technology in Aalesund, Postboks 1517, 6025 Aalesund, Norway

[2] Department of Ocean Operations and Civil Engineering, Norwegian University of Science and Technology in Aalesund, Postboks 1517, 6025 Aalesund, Norway

[3] Institute for Ship Structural Design and Analysis, Technical University of Hamburg, Room 4.008 Schwarzenbergstraße 95 (C), 21073 Hamburg, Germany

adequate to the requirements that change frequently. The typical design process of maritime crane systems starts from customer requests, concept design, and embodiment design, to system modeling and simulation for testing and evaluation, in addition to the development of operational training simulators. Many computer-aided software tools are used in several stages for design, modeling, analytical simulations, and 3D virtual reality animations. On one hand, the use of computer-aided tools improves the work efficiency and thus saves time and cost. On the other hand, the time and cost of the sequential design process still have much potential to be compressed. Currently, modeling and simulations start no earlier than the substantial completion of design, even though the essential functional parameters (e.g., geometry dimensions, work space and constraints, load capacity) are already known from the beginning. Model-based simulations are dependent on each other and hard to integrate due to not only the complexity of the models of the physical systems from different domains but also the interfacing between the different domain-favored modeling tools.

Another crucial field regarding maritime crane simulations is related to operational applications. Due to the inherited issues such as heavy lifting, positioning accuracy, load sway, and security, maritime crane operations are demanding tasks considering both work efficiency and safety. Unlike land-based cranes with fixed working platforms, maritime crane operations are affected by the ship motions and the pendulum load. Today, crane manipulations still rely on the skills of experienced operators, which usually takes a lot of time and cost to train the operators for various types of cranes and operation scenarios. Several compensation approaches have been presented in the literatures [2–6]. However, advanced intelligent control of maritime cranes also requires simulations for testing and analysis of both the software and hardware performances. The existing maritime crane operation centers, e.g., Offshore Simulation Center (OSC), are used nearly exclusively for crew training purposes to reduce the potential risks of human-caused accidents [7]. The dynamics of the simulation scenarios are only partly integrated and usually simplified, including inertia properties, contact boundaries, flexible wire, simple hydraulics, and hydrodynamics. Having high fidelity dynamic models of physical systems for real-time simulations is crucial but rather challenging to achieve.

To tackle these problems, this study introduces a Virtual Prototyping (VP) framework for general crane design and dynamic system simulations. An extended abstract was presented previously in [8] describing the model integration part. Simulations in virtual environment rely on complex models of the physical systems as well as an equally complex model of the interaction environment. The main challenge is to be able to develop proper fidelity models within short time frames so that evaluating design concepts could be done effectively. The VP system offers an open and flexible simulation environment allowing for pre-testing, fault finding, analysis, and control algorithm verification. This will improve the work efficiency of the maritime industry in a collaborative way. Operation simulators based on the VP framework are also enhanced by integrating the dynamic models of the physical systems.

The rest of the paper is organized as follows. Section 2 provides a brief review of the evolution of the engineering system development process. It focuses on the support of computer-aided technologies, the challenges in modeling and simulation of complex multi-domain dynamic systems, and the advantages and disadvantages of alternative solutions. These then lead to the introduction of VP in general and VP applications in engineering system design. Section 3 describes the proposed VP framework for maritime crane design and operations. Sections 4 and 5 present the development and integration of the dynamic models based on a knuckle boom crane (KBC) system. Section 6 presents the simulation results of operating the crane in the VP simulator to show the behaviors of the crane systems in addition to the 3D visualization scenes. The challenges in communication including data exchange for co-simulation and visualization ware discussed. At last, Sect. 7 outlines the conclusions and the future work of this study.

2 State of the art

2.1 The evolution of the engineering system design process

Engineering design originates from a concept, through representations in the form of either digital models or physical prototypes, until it matures and can be realized into a product that fulfills certain functions. The conventional sequential model of the product development process can be tedious and costly when physical prototypes must be built, tested, and re-built after every unsuccessful trial. Faulty decisions in early stages of the design process have profound effects on those that follow. Poor design practice results in high cost and long product delivery time, which can be devastating to a developing company.

The advent and development of computers has changed the design process significantly in many aspects. Computer-aided design (CAD) tools, which could once only create 2D drawings and geometric models, can now create 3D solid and functional models [9]. The computer-aided tools (CAx) and their user interfaces have improved significantly. But mathematical models nonetheless only reflect certain properties of the physical system. Different sub-systems are modeled through different methods and model implementation and handling usually performed in different tools. Without a single modeling method or software tool suitable for integrating the modeling and simulating of all of the subsystems, CAx tools can only be used for modeling and simulation of specific aspects of the entire system. Currently, computer-aided engineering based on computer-aided models supports the design via simulations such as finite element analysis, multi-body dynamics, and computational fluid dynamics. Due to the communication and interoperability of the domain-favored software tools, interest in integrated simulations of multi-domain systems has grown, especially for system design beyond the mechanical domain. The challenges in model integration of different domains and interfacing between different tools are discussed. Specifically, simulations of the maritime crane systems involve mechanical systems, hydraulic systems, and control algorithms, as well as environmental effects such as hydrodynamics for the vessel, wave, and wind.

2.2 Modeling and simulation of dynamic systems

For systems like aircraft, automobiles, and maritime crane, developing physical prototypes or performing field tests involves many other parties and is tightly constrained by the environmental conditions. Simulations in virtual environment not only alleviate the time and cost associated with testing physical prototypes but also allows the designer to predict and prevent inadequate design alternatives as early as possible. As a result, the design trade-offs can be evaluated through the collaborations with the customers and system engineers.

Currently, simulations of the physics and dynamics of complex multi-domain systems are carried out separately and supported by different domain-favored software tools. The mathematical models representing the interested properties within a certain domain are usually simplified, and the effects of the rest are idealized, simplified or even neglected. Assembly of these heterogeneous simulations is difficult because of the interactions between large-scale complex dynamic models, especially, where the real-time performance of simulations is essential. The classification of different simulation scenarios in different detailed modeling levels must be defined depending on the purposes. Another important aspect of modeling is the flexibility in describing the dynamic system, which means the models should be designed open and adjustable to the user. In the last few decades, many good tools have been used to model and simulate dynamic systems domain-specifically. The particular interest in better comprehending the growing complexity of the modern systems has led the software developers to make modeling and simulations of multi-domain systems in the same environment with several software packages, extensions and toolboxes, (e.g., Matlab/Simulink, SimulationX and 20-sim). These software tools have proved to be rather useful to support system design and analysis.

There are two developing trends to the modeling of complex multi-domain systems [10]. First, modeling the entire system with one consistent approach has the advantage of supporting deep understanding of the equation-based model and the flexibility of modification. This requires the modeling language to be suitable for various physical domains. In theory, multi-domain dynamics models could be assembled because the same general state variables are used; however, the challenges in real-time performance of the simulation may arise when the model gets complex. The author previously presented an integrated model for offshore crane operations using the bond graph (BG) method and implemented in 20-sim [11]. The simulations get rather stiff when all the complex models are handled simultaneously, due to the computations of the dynamic equations describing the nonlinear properties of the physical sub-systems, such as the fluid dynamics of the hydraulic systems and the multi-body dynamics of the tightly coupled rigid bodies of the crane.

The other alternative is to distribute the computations of these sub-models and define a unified interface standard for model exchange or co-simulation. This offers an efficient way to cope with the interaction of the heterogeneous simulation tools, and allows for model development using specialized tools for different physical domains. The main challenge is the interfacing between these sub-models favored by different modeling tools and solving the integrated model in the same environment. Related to maritime systems, Chin presented simulations based on Matlab and Simulink models of dynamic positioning (DP) of a drilling ship for thrust optimization design and control under environment disturbances [12]. Terashima presented a virtual plant of a shipboard crane combines CFD with mechanical dynamics [13]. Ku developed a dynamics kernel for multi-body systems considering external hydrostatic and hydrodynamic effects for offshore crane operations [14]. Lee presented the dynamic response of a floating crane and a suspended heavy cargo considering the nonlinear effect of hydrostatic force [15]. Li and Wang presented a visual simulation system for shipborne crane control using Open Graphics Library (OpenGL) and VC++ [16]. Thekkedan et al. presented simulations of underwater positioning controls using Matlab and Simulink [17]. It includes a dynamic model of the 6-DoF Underwater Robotic Vehicle (URV), and the Fuzzy logic controller for the positioning of the URV. The 3D visualization is developed using a Virtual Reality Modeling Language (VRML), which can import CAD models to create a virtual scenario. The output data from Simulink is connected to the VRML model via the VR sink block.

These studies pointed out several common challenges in multi-domain system simulations as discussed, and proposed different solutions for their specific cases. However, a generic and flexible framework is needed for the implementation of overall operational machinery systems. Matlab/Simulink, for example, is good for modeling and simulations of control systems and provides diverse hardware supports. What's more, simulators or simulations, in general, are usually implemented after the completion of system design and based on simplified models of the interested sub-systems. Models with original detailed design are usually too complex to compute when the real-time performance of the simulations comes in a higher priority. Up to date, open, flexible and integrated real-time simulations of the complete system do not exist and are still challenging for maritime crane operations. Considering both the real-time performance of the simulations and reuse the existing knowledge in modeling, it seems a more rational choice to establish a unified format for interfacing and supported by the software tools. As a result, models for the sub-systems and components can be developed in different tools depending on their disciplines and the preferences of the users. Model integration, or rather modeling of systems, requires a separate tool or an integration environment. Modularization of a complex system into sub-systems is also convenient for modifications, exchange and reuse of models, and collaborations between different departments and companies.

Recent advances in the exchange of simulation models have been significant. The automotive industry has been particularly innovative with projects such as the European Modelisar project, which led to the Functional Mock-up Interface (FMI) specification, among other things [18]. The FMI 1.0 and 2.0 is now partly or completely supported by over 100 software tools in two concepts for simulation exchange, namely model exchange and co-simulation. The main difference is that co-simulation requires the compilation of the Functional Mock-up Unit (FMU) with a numeric solver, while an FMU for model exchange only implements the model itself and the computation relies on the solver of the integrating tool. It has shown that significant computation time speed-ups can be regained at the price of a moderate loss of accuracy for strongly coupled systems using FMI co-simulation [19]. Erdelyi et al. presented their implementation of the vehicle dynamics case. An air-spring FMU is compiled from Modelica code for model exchange with the front suspension of a vehicle modelled in LMS Virtual.Las Motion [20]. Neema et al. presented a model-based integration platform called C2WT for FMI co-simulation of cyber-physical systems. A vehicle thermal management system is implemented for studying the interaction of thermal management systems within a vehicle [21]. Drenth and Henningsson implemented an engine model and an engine cooling system model of a vehicle in the FMI toolbox for the use in Matlab. Both systems are modeled in Dymola and imported as FMUs and simulated in Matlab coupled with the controllers [22, 23]. The FMI standard provides a powerful solution for multi-disciplinary system simulation model exchange and co-simulation.

2.3 Virtual prototyping in engineering design

VP entails the integration of dynamic modeling and simulations during the design process to analyze the effects of design trade-offs on the overall system performances [24]. Since a prototype is defined as the representation of a class of design, a virtual prototype should include all the functional aspects of a physical prototype could offer. VP is essentially modeling and simulations of all the related aspects of a product, system or system of systems. VP also extends the conventional simulations to include human interactions. In engineering fields, the current technology could only offer limited realizations in visual sense, tactile and haptic (force feedback), and audio (critical for some cases). The application of haptic technology in crane operations also adds a sense of touch to the operators to improve safer operations [25, 26]. The aerospace and automobile industry has taken leading roles in VP for product design, manufacturing and simulation [27, 28]. Chin and Lum presented rapid control system prototyping of an underwater robotic vehicle (URV) based on the dynamic models in Matlab and Simulink [29]. The system also includes the controller of the URV (an industrial computer which runs the software). The communication to the Matlab/Simulink model is realized via Ethernet link. Further, Prats et al. developed an open source tool called UWSim for the simulation of URV applications [30]. The simulator is implemented in C++ and uses OpenSceneGraph (OSG) and osgOcean libraries for rendering. It also provides interfaces to external architectures such as control programs and dynamic models in Matlab through the Robotic Operating System (ROS). Physics of contact is facilitated by the physics engine Bullet wrapped in osgBullet for its use with OSG.

A conceptual diagram of VP for system design and operations is shown as in Fig. 1. Given a platform which can cope with the inputs and outputs from all interested users as early as possible during the design process, the product development process would become more efficient and effective. From the previous discussion on multi-domain system modeling and simulations of physical systems, several common features could be recognized, including models of physics, computation servers, communication interfaces, and visualizations. Models, computation and visualization/user interfaces are separated in different layers [31]. The decomposition brings the challenges in interfacing and communication, but facilitates the improvement of simulation efficiency in particular, for stiff systems. As a result, the VP platform closes the design loop, and becomes open, flexible, and software tool independent for modeling and simulations. Virtual operation simulators also allow for real-time

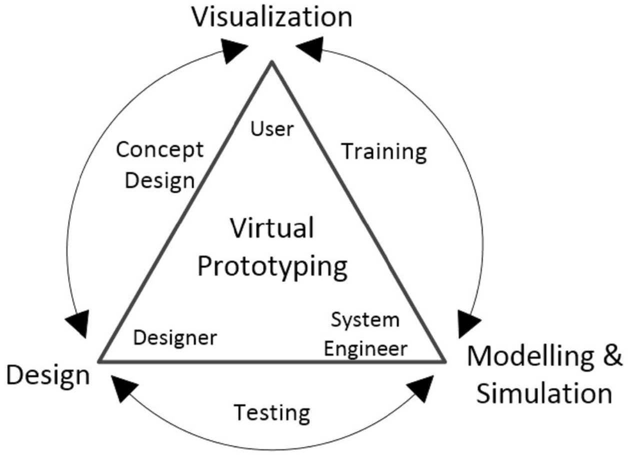


Fig. 1 Virtual prototyping for product and system design, modelling and simulation, visualization

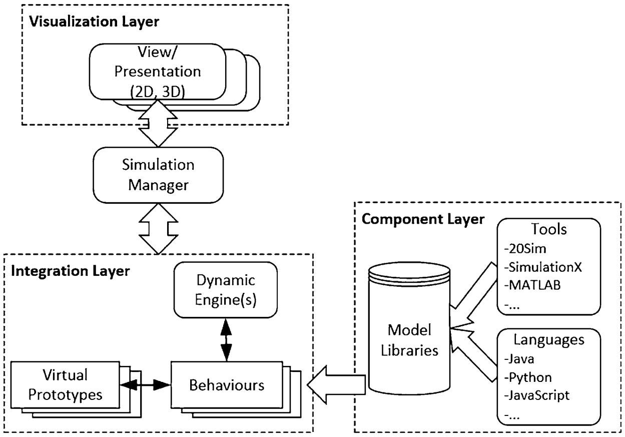


Fig. 2 The high-level software architecture of the VP framework

simulations reinforced by dynamic models of the physical systems.

3 Virtual crane prototyping framework: software architecture

A general high-level architecture of the VP framework is proposed for simulation and visualization of marine crane operations, as shown in Fig. 2. The VP framework is divided into three layers. The core of the VP framework is rendering-agnostic and effectively decoupled from the visualization layer. This is important for the reuse of simulation models with different views, and to use the visualization code with different models. The user interface, for example, includes views and controllers of the simulator, which must be portable and accessible from many different platforms and different operating control systems. This is facilitated using web technologies, which offers 3D visualization using the Web Graphics Library (WebGL), and turns the web browser into a powerful visualization tool. WebGL (Web Graphics Library) is a JavaScript API for rendering interactive 3D and 2D graphics within any compatible web browser without the use of plug-ins. WebGL does so by introducing an API that closely conforms to OpenGL ES 2.0 that can be used in HTML5 <canvas> elements. The VP crane simulator described in this paper uses the popular open-source JavaScript 3D library (three.js) to render the in-browser graphics of the crane bodies and other objects. An alternative implementation for 3D rendering on the desktop is based on the cross-platform game engine jMonkeyEngine3.

Real-time bi-directional communication between the visualization and integration layer is facilitated using the TCP/IP and/or WebSocket protocols, while static files are served using HTTP. The JSON RPC standard is used to facilitate data transmission between the clients and the server. Any tool or language that can communicate with the integration layer over these communication protocols can be used to interact with and/or visualize the running simulations. The clients can choose to either pull updates from the server at their own pace, or subscribe to updates at a specified frequency. In the latter case, the server will try to accommodate the desired update speed of the client but naturally cannot send updates more frequently than the frequency of the simulation itself.

The component layer contains separate simulators for different components and sub-systems; while the integration layer implements the system model comprised of these sub-models. The component models are created by the modeling tools and exported as FMUs. It is also possible to implement any model or function using general-purpose programming languages or other tools which do not support the FMI standard. In the latter case, a communication protocol to the server must be set up. The integration layer implements the system model and acts as a master for co-simulations, with the co-simulation slaves running in the component layer. The simulation manager is responsible for managing references to the available simulations. Initially, the clients must negotiate with the simulation manager to get the IP address of a simulation. Successively, the clients are able to query the simulations directly. In the component layer, we use a thin Java wrapper around the FMUs to export the functionality via Remote method invocation (RMI). This allows the possibility to distribute FMUs on the network. By doing so, FMUs with different compilation targets can also be supported. For example, an FMU compiled for Linux would run in a Linux box, and accessed over RMI by the VP system running in Windows.

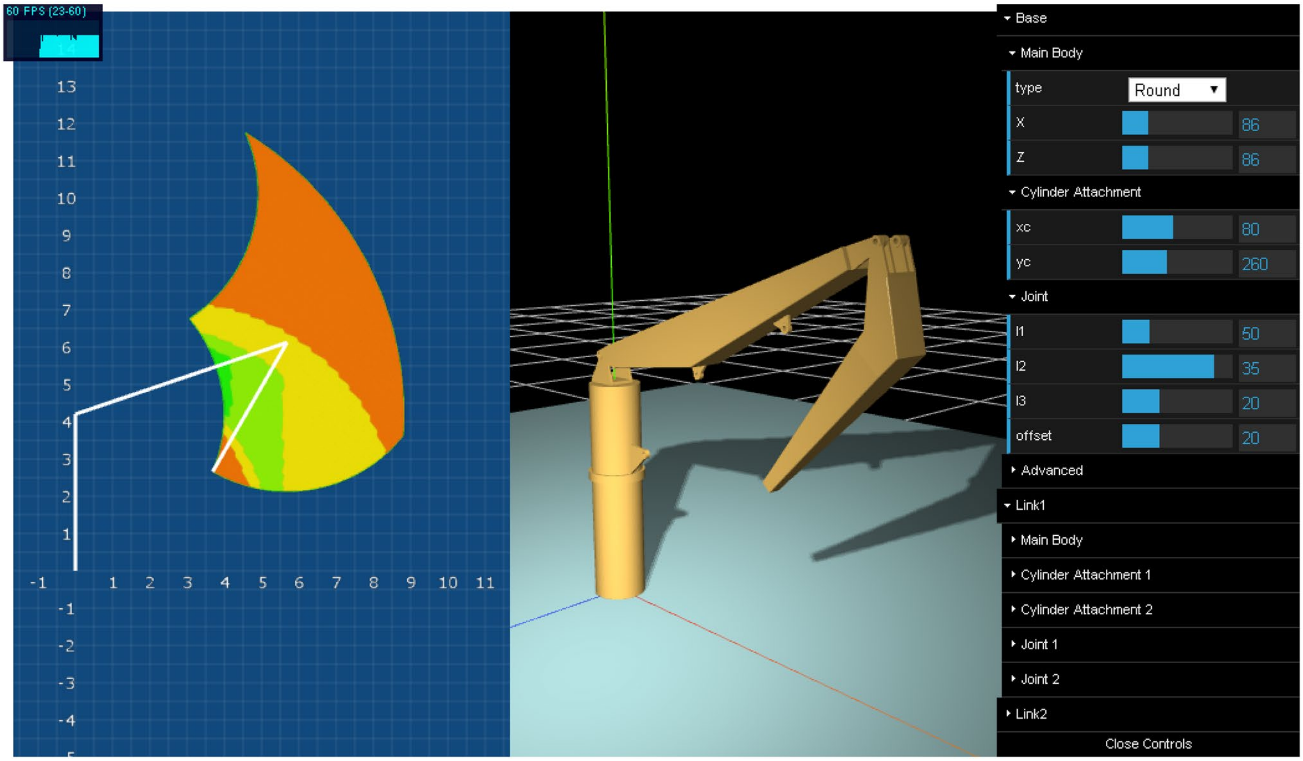
4 The component layer: modeling of dynamics

4.1 Mechanical design

3D solid models are often created in CAD software packages such as Solidworks, NX, and CATIA. FEA and static analysis can be performed by these programs. However, to integrate mechanical design directly in real-time simulations is hard for the currently available CAD tools. Usually, simplifications of the 3D models must be done to perform any dynamic simulation. But manual transcriptions tend to be time-consuming and error-prone. During the concept design stage, in particular, for new product design space exploration, it benefits the latter processes to check the main geometrical and functional features by simulations. Hameed et al. introduced a VP tool for crane design optimization focusing on the working space and load capacity [33]. It uses the genetic algorithm to optimize the dimension design of the crane structure. The VP tool, however, does not include the dynamic models, e.g., of the multi-body crane or the hydraulic power system. Therefore, a lightweight and flexible solution-based is presented for mechanical design, and provide the physical models for simulations in the VP system [34].

The crane designer tool consists of a workspace editor (WE) and a mechanical editor (ME). The WE interprets the requirements from the customer including the crane type, workspace and load capability into initiative technical specifications, and sends it to the ME. The ME is a 3D scene editor based on Three.js JavaScript library/API that uses WebGL to display 3D computer graphics in the browser. A parametric component library is pre-defined containing a variety of basic components, e.g., main bodies, joints, cylinder attachments. The components in the library are 3D mesh models composed of vertices and faces. The arrangement of the vertices of each component is controlled by a set of parameters, which means manipulating these parameters modifies the shape of the component. The physical properties, such as mass and inertia, can either be assigned manually or calculated by the total volume of the mesh model in fair approximations given a density. The crane workspace and load chart can then be computed by defining the joint constraints, and visualized in a browser, as shown in Fig. 3.

For the purpose of fast prototyping, the mesh models of the mechanical parts should be simplified but retain sufficient features for both 3D visualization and dynamic simulations in the later stages. The products of each body part from the crane designer are saved after evaluation. This includes the mesh model, kinematic properties and mass properties specifications into a single data collection, which can be



▾ Base
▾ Main Body
type Round ▾
X 86
Z 86
▾ Cylinder Attachment
xc 80
yc 260
▾ Joint
l1 50
l2 35
l3 20
offset 20
▸ Advanced
▾ Link1
▸ Main Body
▸ Cylinder Attachment 1
▸ Cylinder Attachment 2
▸ Joint 1
▸ Joint 2
▸ Link2
Close Controls

Fig. 3 A snapshot of the crane designer tool, workspace and load chart visualizations in a web browser

exported as a .ZIP-file and imported into the VP framework. Multi-body dynamics is then handled by a physics engine and uses a Java wrapper for interfacing with other sub-models in the virtual simulator.

4.2 Hydraulic system

Maritime cranes are mostly powered by hydraulic actuators because of their high-effort outputs, fast response characteristics and system robustness in rough operation conditions. Simulations of the dynamics of the hydraulic system are important for the study of the effects of control algorithms to the system performance stability. A simplified hydraulic diagram of the KBC is shown in Fig. 4. Modeling of the hydraulic system for co-simulation can be done using any other FMI supported software tools. The components models must fulfill the requirements of easy modification, modularized for model reuse, and classified detail-levels for different simulation purposes.

A previous study presents the hydraulic models of the KBC using bond graphs [35]. BG method is a modeling technique based on identifying the energetic structure of the system [36]. A physical system can be decomposed into several basic components and properties represented by interrelated idealized elements that represent these properties. The component models of the hydraulic system of the KBC include a pressure compensated pump as the flow source, lumped pipe flow, pressure compensation valve for the direction valve, direction valve, load control valve, double acting cylinder, displacement hydraulic motor, gear transmission and return tank. For the implementation of FMU co-simulation, the complete hydraulic system of the KBC is divided into three sub-systems: the slewing motor at the crane base, and the cylinders at the main boom and the outer jib of the crane. The BG model implementations in 20-sim are shown in Fig. 5. The input control signal is the gain to the direction control valve. The preferred output of the hydraulic FMUs is the velocity of the actuators and the feedback is the acting force (or torque) from the crane. A beta version of the software tool 20-sim currently supports exporting FMU co-simulation.

4.3 Control algorithms

As mentioned, the ship motions and the pendulum load cause many problems for offshore crane operations. Current heave compensation solutions using the winch have experienced many critical problems with the winch power system and the heave lifting wire. Therefore, an effective heave compensation using the crane body was proposed based on the inverse kinematics algorithm [6]. Using this algorithm, it is easier for the implementation of the compensation algorithms regardless of the structure of the crane and for operators to position the load. The kinematic sketch of the KBC is shown in Fig. 6 based on the Denavit–Hartenberg (D–H) method [37]. Cranes implemented in the VP simulator can

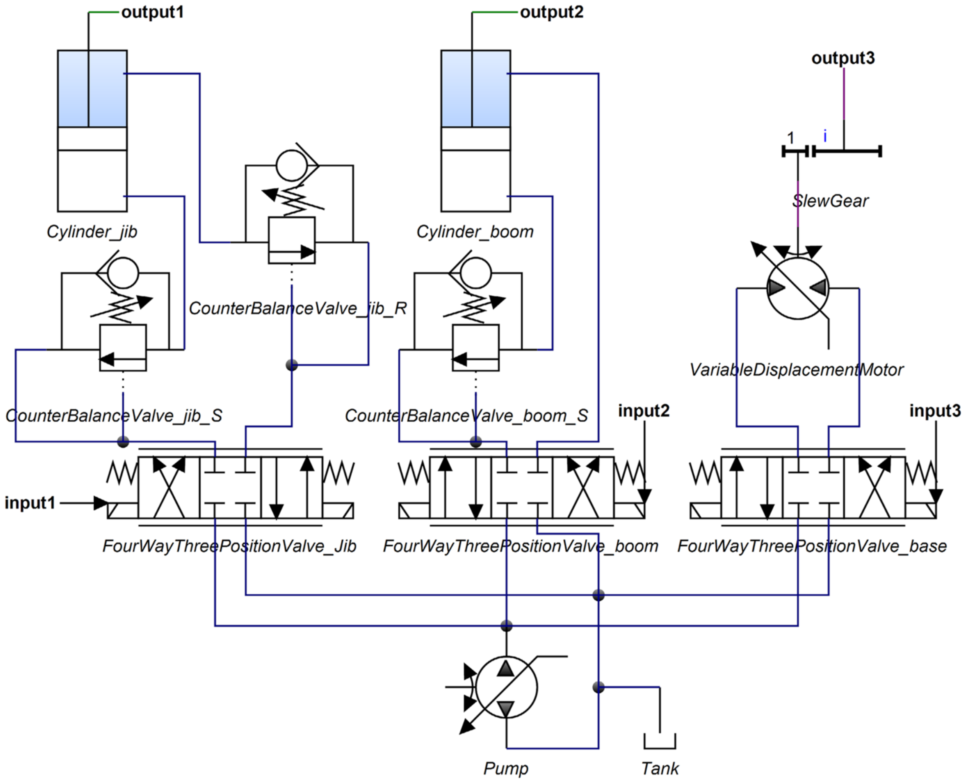

output1
output2
output3
Cylinder_jib
Cylinder_boom
1
i
SlewGear
CounterBalanceValve_jib_R
VariableDisplacementMotor
CounterBalanceValve_jib_S
CounterBalanceValve_boom_S
input2
input3
input1
FourWayThreePositionValve_Jib
FourWayThreePositionValve_boom
FourWayThreePositionValve_base
Pump
Tank

Fig. 4 A simplified hydraulic diagram of the knuckle boom crane

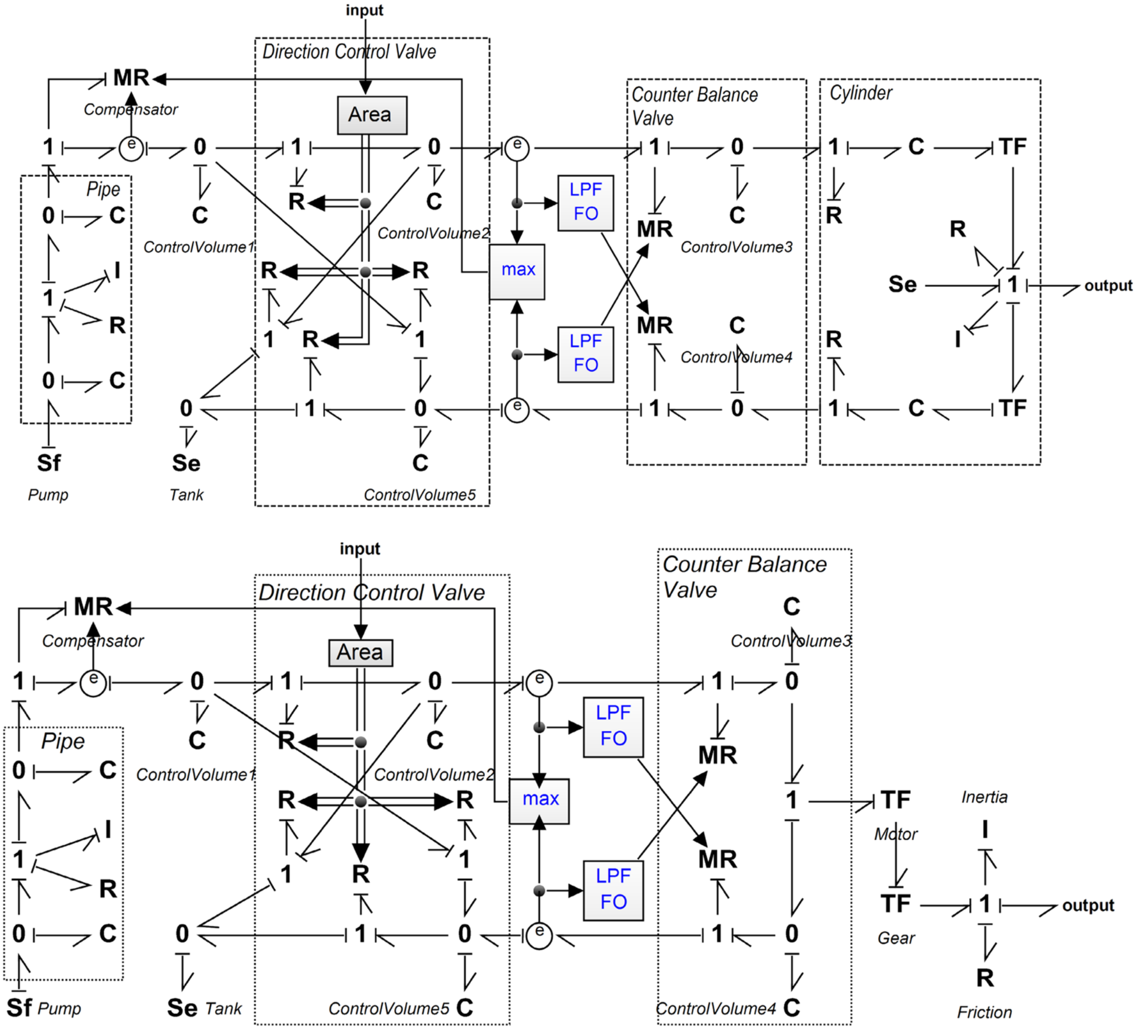


Fig. 5 BG model implementations of the hydraulic cylinder and hydraulic motor

be controlled in a variety of different ways. This includes direct joint-by-joint control, and inverse control with the compensation functions. The control algorithms are implemented directly in Java in the KBC simulator. The kinematic analysis of the KBC for heave compensation are presented, see also reference [38].

The forward transformation matrix is given by Eq. 1:

$$
{}^0_4T = {}^0_1T\,{}^1_2T\,{}^2_3T\,{}^3_4T = \begin{bmatrix} c_1c_{23} & -c_1s_{23} & s_1 & c_1\left(L_2c_2+L_3c_{23}\right) \\ s_1c_{23} & -s_1s_{23} & -c_1 & s_1\left(L_2c_2+L_3c_{23}\right) \\ s_{23} & c_{23} & 0 & L_1+L_2s_2+L_3s_{23} \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{1}
$$

where $c_i = \cos\left(\theta_i\right)$, $s_i = \sin\left(\theta_i\right)$, $c_{ij} = \cos\left(\theta_i+\theta_j\right)$, $s_{ij} = \sin\left(\theta_i+\theta_j\right)$ $(i, \mathrm{j}=1{,}2{,}3)$.

The velocity Jacobian relating the joint angular velocities and the crane tip velocities in Cartesian space can be derived, as in Eq. 2:

$$
{}^0J(\theta) = \begin{bmatrix} -s_1\left(L_2c_2+L_3c_{23}\right) & -c_1\left(L_2s_2+L_3s_{23}\right) & -L_3c_1s_{23} \\ c_1\left(L_2c_2+L_3c_{23}\right) & -s_1\left(L_2s_2+L_3s_{23}\right) & -L_3s_1s_{23} \\ 0 & L_2c_2+L_3c_{23} & L_3c_{23} \end{bmatrix} \tag{2}
$$

Hence, given the crane tip velocity, the joint angular velocities can be calculated with the following Eq. 3:

$$
\dot{\theta} = {}^0J(\theta)^{-10}v \tag{3}
$$

Ideally, active heave compensation can then be achieved by feeding an equivalent inverse velocity to keep the crane tip. The joint velocities are can be calculated by Eq. 4:

$$
\dot{\theta} = J(\theta)^{-1}\left[\,0\;\;0\;\;-v_{\mathrm{heave}}\,\right]^{\mathrm{T}} \tag{4}
$$

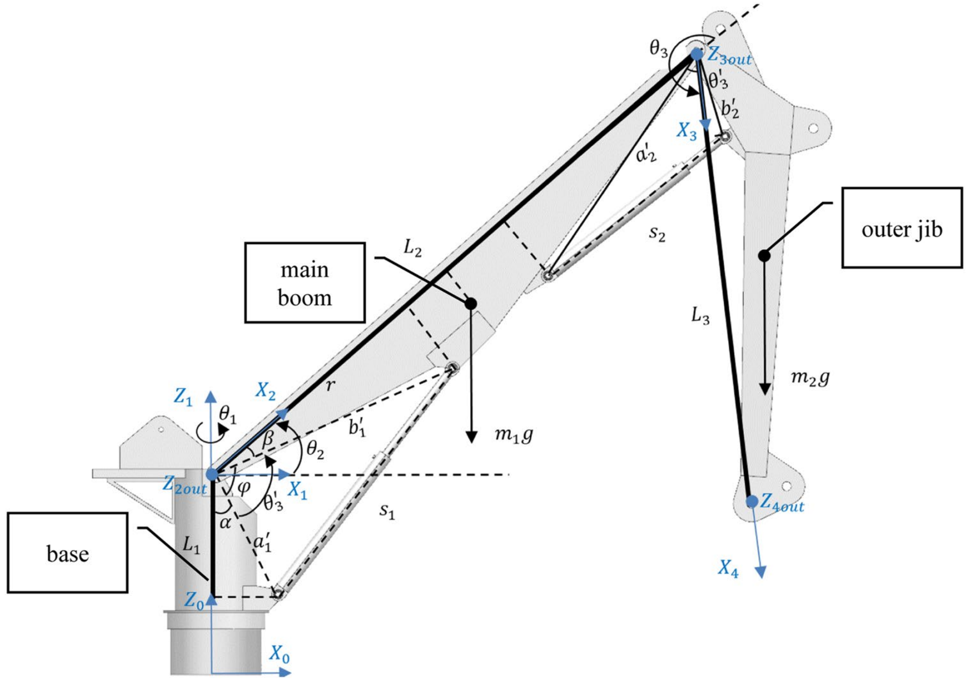


Fig. 6 Kinematic diagram of the 3-DoF knuckle boom crane

The transformation matrix $T(\theta)$ from the kinematic joints to the actuators of the crane, i.e., the motor and cylinders, is given by Eq. 5.

$$T(\theta) = \begin{bmatrix} i & 0 & 0 \\ 0 & \frac{a_1' b_1' \sin(\theta_2')}{s_1} & 0 \\ 0 & 0 & \frac{a_2' b_2' \sin(\theta_3')}{s_2} \end{bmatrix} \tag{5}$$

where i is the gear ratio of the slewing joint.

5 The integration layer: integration of the dynamic models and simulations

Provided with the models of the sub-systems of the KBC, the virtual crane simulator can be set up. As shown in Fig. 7, the VP simulator for the KBC includes a ship with a hydrodynamic model from the physics engine AgX which also handles the physics and multi-body dynamics of the crane. Models of the hydraulic power systems are implemented in 20-sim and used as FMUs in the simulator. The integration layer has access to the dynamic engines and FMU solvers. Both the physics engine and various models implemented as

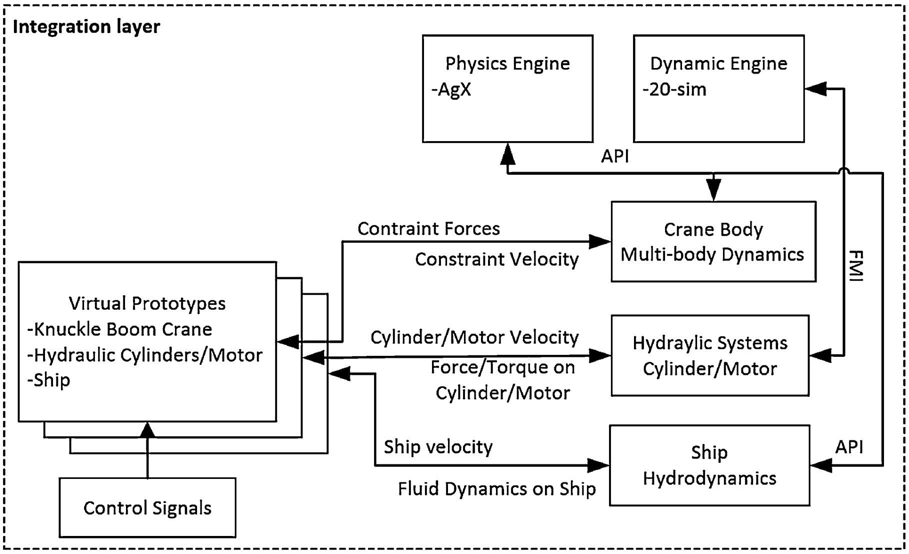


Fig. 7 The integration of the knuckle boom crane system modelling and simulations

FMUs may be included and intertwined in the simulation, and could also be connected to each other. For example, the output velocity from the hydraulic FMU is used as the input to the constraint of the multi-body model, while the reaction force of the constraint is sent back to the FMU. The high fidelity physics engine AgX allows for accurate simulation of physical and mechanical laws, including gravity, collision handling, constraints, buoyancy, wind, etc. However, the use of the physics engine AgX for rigid body dynamics is optional as well as the use of the tool 20-sim for simulation of the hydraulic power systems. In another word, the hydrodynamic model for the ship and the multi-body dynamics model of the crane can be realized using other tools.

When a strong-coupled system, such as the hydraulic actuators attached to the crane body via constraints, is modelled separately and integrated again through co-simulation, the overall simulation accuracy and stability needs to be treated carefully. The trade-offs between simulation accuracy and stability must be examined purposefully. For example, the real-time performance of the simulation must be ensured during operation training practices. In this case, the modeling and simulation of dynamic systems could be simplified when many objects are included in the simulation scenario. In another case for system testing and analysis, detailed modeling of the dynamics must reflect the interested properties and behaviors of the physical systems. Hence, the real-time performance of the overall simulation becomes less crucial. In a word, the computation time-steps for co-simulations and data exchange for communication and visualization vary from case to case.

6 The visualization layer: data rendering

In the proposed VP system for maritime crane operations, a set of HTML5 web pages has been created to handle the user inputs and display 2D plots, 3D visuals and other types of data according to requirements. Data generated by the VP simulators is made accessible to the user in real-time. More specifically, a collection of available plots of simulation variables is presented to the user through the web browser or other tools for display. For example, complete time-series can also be downloaded as comma separated files (.csv) to be read in Matlab, as below Figs. 8 and 9.

Via a joystick, a random control signal is sent to the direction valve of the cylinder as the input. The behaviors of the hydraulic cylinder and the crane in the VP simulator are presented as shown in Fig. 8. The reference speed and displacement are read from the hydraulic FMU which is handled by 20-sim, and the actual speed and displacement are from the rigid body model handled by the physics engine AgX. When the gain is positive, the speed of the piston is positive and the displacement of the piston increases. This means in fact that the cylinder is being extended, and vice versa. The maximum displacement of the cylinder piston is set at 1.4 m plus 0.01 m deflection of the stopper. As a result, the displacement of the piston stops increasing at about 8 and

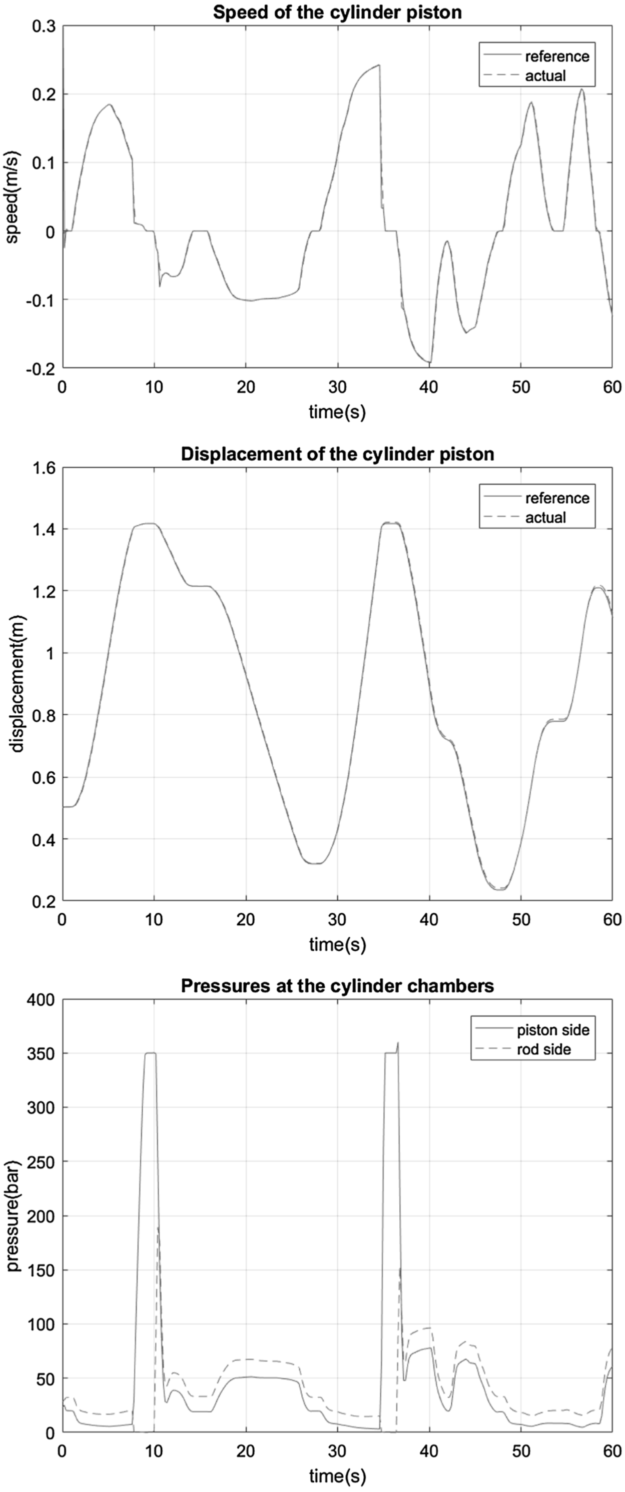


Fig. 8 The behaviors of the hydraulic cylinder during manual operations

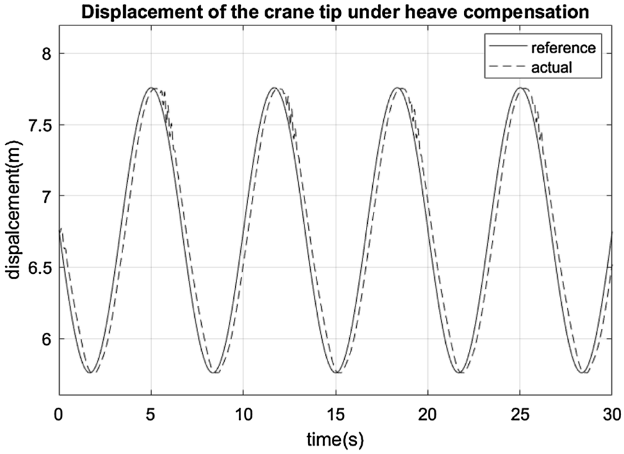


Fig. 9 The displacement of the crane tip during heave compensation operations

35 s even as the gain and speed are still in the positive direction. The pressures inside the two chambers of the hydraulic cylinder reflect the response of the system when the cylinder piston is researched to the end stopper. At about 8 and 35 s, the pressure at the piston side of the cylinder increases up to the maximum pressure, while the pressure at the payload side drops to zero (return tank pressure).

The displacement of the crane tip under active heave compensation operations is shown in Fig. 9. The control signal is the heave motion as the reference for the displacement of the crane tip, which is defined as a regular heave motion with 2 m wave height and 12 s period. Regular PID-controllers are used for the control of the hydraulic actuators. The actual displacement of the crane tip is obtained from the physics engine AgX. The effectiveness of the heave compensation algorithm is dependent on the performance of the hydraulic power systems as well as the structural properties of the crane. External disturbances from, such as the payload sway and the wind effects are not included in this case. Analyzing the response characteristics of the hydraulic system could improve the understanding of its behaviors during active heave compensation operations, but this exceeds the scope of the paper.

To show the modularity of the VP framework for visualization, a second implementation for rendering the 3D visuals is created using the Java-based cross-platform game engine jMonkeyEngine3. This implementation supports both the use of TCP/IP and WebSockets to communicate with the simulation server. Figure 10 shows the same scene in the browser using WebGL and on a desktop application using OpenGL, respectively. Changes to the simulation models in either of the implementations are propagated to all connected clients, which will have all views updated accordingly, thanks to the bi-directional capabilities of the real-time communication protocols. The approach of making the simulation accessible over the network provides more flexibility allowing multiple users to interact with the same simulation on different computers and letting the users control and visualize the simulation in whichever tool they might prefer. However, it may come with a considerable communication overhead. For a large simulation scenario with many visual objects, the amount of data needed to render the simulation with a pleasing frame rate can overwhelm either the internet connection or the client's processing. In such cases, the VP framework can be used as a library, where data is accessed through regular API calls, which does not have a communication overhead. Other simulation data, such as plotting data could still be accessed through the network.

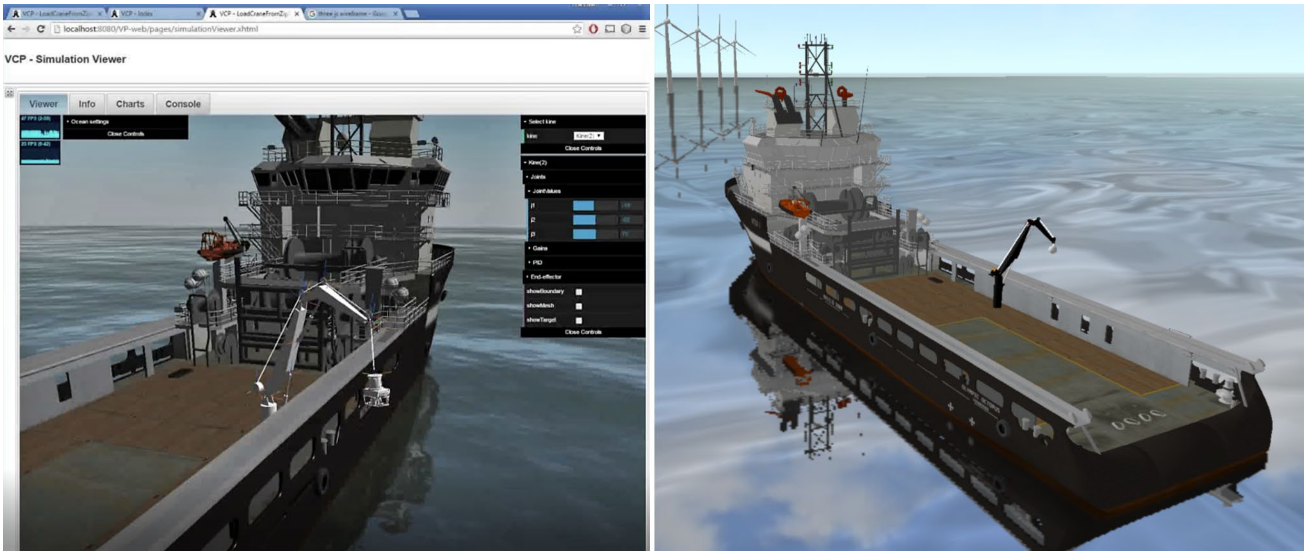


Fig. 10 The same scene rendered in WebGL and OpenGL simultaneously

7 Conclusions

The paper discussed the challenges of VP for multi-domain system design and operations. Modern engineering design is characterized by high customization, high complexity, high price, low production volume and short time for development. Over the last few decades, leading industries such as aerospace, automobile and shipbuilding have increasingly employed VP techniques. However, many challenges still exist in the development and implementation of these techniques. Taking current maritime crane systems as example, the following issues were highlighted.

- The development of integrated virtual environment for system design and verification is far behind the advance of computer technologies. A lot of manual work still exists for data exchanging and information sharing between different departments, which is time and cost consuming and error-prone.
- Current operation simulators are exclusively used for training purposes. Simulations of crane operations are dependent on ideal simplified mathematical models. The integration of physics and dynamics is challenging, especially, when real-time performance is required.
- There is no established standard for the interfacing of the heterogeneous simulation models of the multi-domain physical systems, where modeling and simulation usually performed using different software tools.

The software structure of the proposed VP framework is separated into three interrelated layers. The interfacing and communication of a multi-domain system models for co-simulation is based on the application of the FMI standard. Involved modules of the maritime crane system include mechanical design, hydraulic system modeling, and control algorithms for heave compensation, etc. As a case study for verification, the development of these sub-models was presented and implemented in the VP crane simulator. Real-time bi-directional communication between the visualization and integration layer is facilitated using the TCP/IP and/or WebSocket protocols. Visualization and user interface based on web technologies provides a flexible and lightweight approach for interaction and information sharing.

The VP system will change the current maritime crane design process allowing for combined collaborations between customers, designers, engineers, operators. The work efficiency of product development, system testing and analysis, and operation safety will also be improved. The results as a general knowledge resource could also be transferred to other multi-domain dynamic systems. The main novelty of the proposed VP system is the openness and flexibility for modeling, simulation and visualization. Knowledge within different disciplines can be reused efficiently facilitated by the FMI standard which is currently supported by most of the available modeling and simulation tools. The VP crane simulator brings physics and dynamics into the simulations of complex operation applications, hence, the results become more realistic in reflecting the behaviors of the physical world.

There are still several challenges to make the VP system more generic and robust. The future work of this study includes improving and standardizing a more generic software architecture of the VP system regarding the flexibility of rendering different simulation scenarios. Open, flexible and tool independent component model libraries with different complexities will be established for different simulation purposes, including the mechanical parts, hydraulic systems and control algorithms and compensation functions. Implementation editor or graphical user interface will be developed to make the use of the VP system more efficient, especially for the users who lack the knowledge of programming or modeling of dynamic systems. Finally, yet importantly, the benchmarking for model simplification and simulation performance should be established depending on different simulation purposes. Specifically, the simulation accuracy and stability of using FMI co-simulations needs to be investigated. For stiff systems, the stability of the simulation is crucial when tightly coupled parts are separated. The simulation time-steps must be decided cautiously to achieve least real-time simulation. Clever master algorithms need to be introduced when necessary to improve the simulation efficiency without losing much accuracy.

Acknowledgements The authors thank Prof. Hans Georg Schaathun for his advice on the software architecture, and Yuxiang Deng and Birger Skogeng Pedersen for their support on the implementation of crane designer tool to the VP framework. The project is financially sponsored by the Norwegian National Research Foundation (Innovation Projects for the Industrial Sector, MAROFF 235309, and Centre for Research-based Innovation, SFI 237929).

References

1. Bak MK (2014) Model based design of electro-hydraulic motion control systems for offshore pipe handling equipment
2. Johansen TA, Fossen TI, Sagatun SI, Nielsen FG (2003) Wave synchronizing crane control during water entry in offshore moonpool operations—experimental results. J Ocean Eng 28(4):720–728
3. Skaare B, Egeland O (2006) Parallel force/position crane control in marine operations. J Ocean Eng 31(3):599–613
4. Neupert J, Mahl T, Haessig B, Sawodny O, Schneider K (2008). A heave compensation approach for offshore cranes. American Control Conference, pp 538–543
5. van Albada SB, van Albada GD, Hildre HP, Zhang H (2013). A novel approach to anti-Sway control for marine shipboard cranes. ECMS 249–256

6. Chu Y, Sanfilippo F, Æs V, Zhang H (2014) An effective heave compensation and anti-sway control approach for offshore hydraulic crane operations. IEEE ICMA, Tianjin, pp 1282–1287
7. Offshore Simulation Center. http://www.osc.no. Jun 2017
8. Chu Y, Hatledal LI, Sanfilippo F, Æs V, Zhang H, Schaathun HG (2015) Virtual prototyping system for maritime crane design and operation based on functional mock-up interface. In: Oceans '15 MTS/IEEE. Genova, Italy, pp 1–4
9. Bordegoni M, Rizzi C. Innovation in product design: from CAD to virtual prototyping, Springer, London
10. Bertsch C, Ahle E, Schulmeister U (2014). The functional mockup interface—seen from an industrial perspective. In: 10th International Modelica Conference, Lund, pp 27–33
11. Chu Y, Æsøy V, Ehlers S, Zhang H (2015) Integrated multi-domain system modelling and simulation for offshore crane operations. J Ship Technol Res Schiffstechnik 62(1):36–46
12. Chin CS (2012) Dynamic positioning simulation, thrust optimization design and control of a drill ship under disturbances and faulty thruster. Simulation 88(11):1338–1349
13. Terashima K, Ito R, Noda Y, Masui Y, Iwasa T (2010) Innovative integrated simulator for agile control design on shipboard crane considering ship and load sway. IEEE CCA, Yokohama, 1293–1300
14. Ku N, Ha S, Roh MI (2014) Crane modeling and simulation in offshore structure building industry. J Comput Theory Eng 6(3):278–284
15. Lee KY, Cha JH, Park KP (2010) Dynamic response of a floating crane in waves by considering the nonlinear effect of hydrostatic force. J Ship Technol Res Schiffstechnik 57(1):64–73
16. Li P, Wang C (2009) Design and implementation of visual simulation system for shipborne crane control. CCDC'09. Guilin, Guangxi, Chinese, pp 5482–5486
17. Thekkedan MD, Chin CS, Woo WL (2015) Virtual reality simulation of fuzzy-logic control during underwater dynamic positioning. J Mar Sci Appl 14(1):14–24
18. Functional Mockup Interface. https://www.fmi-standard.org/. Jun. 2017
19. Viel A (2014) Implementing stabilized co-simulation of strongly coupled systems using the Functional Mock-up Interface 2.0. In: Modelica'2014. Lund, Sweden, pp 213–223
20. Neema H, Gohl J, Lattmann Z, Sztipanovits J, Karsai G, Neema S, Sureshkumar C (2014) Model-based integration platform for fmi co-simulation and heterogeneous simulations of cyber-physical systems. In: Modelica'2014. Lund, Sweden, pp 235–245
21. Erdelyi H, Prescott W, Donders S, Anthonis J (2012). FMI implementation in LMS Virtual. Lab Motion and application to a vehicle dynamics case. In: Modelica'2012, Lund, 2012, pp 759–764
22. Henningsson M, Åkesson J, Tummescheit H (2014). An FMI-based tool for robust design of dynamical systems. In: Modelica'2014, Lund, pp 35–42
23. Andreasson J, Elmqvist H, Griffin J, Henriksson D, Mattson SE, Olßon H (2014). Consistent simulation environment with FMI based tool chain. In: Modelica'12, Munich, pp 1277–1283
24. Zorriassatine F, Wykes C, Parkin R, Gindy N (2003) A survey of virtual prototyping techniques for mechanical product development. Proc Inst Mech Eng Part B J Eng Manuf 217(4):513–530
25. Ha S, Kim L, Park S, Jun CS, Rho H (2009) Virtual prototyping enhanced by a haptic interface. J CIRP Ann Manuf Technol 58(1):135–138
26. Chu Y, Zhang H, Wang W (2016) Enhancement of virtual simulator for marine crane operations via haptic device with force feedback. In: International conference on human haptic sensing and touch enabled computer applications, Springer, pp 327–337
27. Bennett GR (1997) The application of virtual prototyping in the development of complex aerospace products. J Airc Eng Aerosp Technol 69(1):19–25
28. Lau HYK, Mak KL, Lu MTH (2003) A virtual design platform for interactive product design and visualization. J Mater Process Technol 139(1):402–407
29. Chin CS, Lum SH (2011) Rapid modeling and control systems prototyping of a marine robotic vehicle with model uncertainties using xPC Target system. Ocean Eng 38(17):2128–2141
30. Prats M, Pérez J, Fernández JJ, Sanz PJ (2012) An open source tool for simulation and supervision of underwater intervention missions. In: 2012 IEEE/RSJ International Conference on Intelligent Robots and Systems, Vilamoura, pp 2577–2582
31. Slomka F, Kollmann S, Moser S, Kempf K (2011). A multidisciplinary design methodology for cyber–physical systems. In: ACESMB 2011, Wellington, pp 23–37
32. Hatledal LI, Schaathun HG, Zhang H (2015) A software architecture for simulation and visualization based on the functional mock-up interface and web technologies. In: SIMS 56. Linköping, Sweden
33. Hameed IA, Bye RT, Osen OL, Pedersen BS, Schaathun HG (2016) Intelligent computer-automated crane design using an online crane prototyping tool. In: Proceedings of the 30th European Conference on Modeling and Simulation (ECMS 2016), pp 564–573
34. Chu Y, Deng Y, Skogeng PB, Zhang H (2016) Parameterization and visualization of marine crane concept design. In: ASME 2016 35th International Conference on Ocean, Offshore and Arctic Engineering. American Society of Mechanical Engineers
35. Chu Y, Æsøy V, Zhang H, Bunes O (2014) Modelling and simulation of an offshore hydraulic crane. In: ECMS 2014. Brecia, Italy, pp 87–93
36. Karnopp DC, Margolis DL, Rosenberg RC (2012) System dynamics: modeling, simulation, and control of mechatronic systems. Wiley, New Jersey
37. Craig JJ (1989) Introduction to robotics. Addison-Wesky, New York
38. Chu Y, Æsøy V (2015) A multi-body dynamic model based on bond graph for maritime hydraulic crane operation. In: OMAE 2015, St. John's, Newfoundland, Canada, No. 41616