

# A HARDWARE-IN-THE-LOOP SIMULATOR FOR OFFSHORE MACHINERY CONTROL SYSTEM TESTING

Johnny Aarseth, Alf Helge Lien and Øyvind Bunes  
Deck Machinery Seismic & Subsea  
Rolls-Royce Marine  
PO Box 193, N-6069 Hareid, Norway  
Email: Johnny.Aarseth@Rolls-Royce.com  
Alf.Helge.Lien@Rolls-Royce.com  
Oyvind.Bunes@Rolls-Royce.com

Yingguang Chu and Vilmar Æsøy  
Department of Maritime Technology and Operations  
Aalesund University College  
PO Box 1517, N-6025 Ålesund, Norway  
Email: yich@hials.no  
ve@hials.no

## KEYWORDS

Offshore machinery, hardware-in-the-loop, modelling and simulation, Bond Graph.

## ABSTRACT

The paper presents a concept to develop a simulator for testing of the control systems of automated handling systems for offshore vessels. The concept is based on using Bond Graph for numeric modelling of physical systems, then implementing the model on a platform which can run and visualize simulations and communicate with the target control system in real time. A simulator is implemented and tested for a Launch and Recovery System (LARS) from Rolls-Royce Marine. The intention for this simulator is to provide a platform for development and testing of the control system without having a full scale physical system available, as that can be both practically difficult and an expensive way of performing testing. The simulator can also serve as a platform operational training as well as testing and monitoring the performance of the machinery system effectively, in particularly the characteristics of the hydraulic components during runtime.

## INTRODUCTION

Offshore operations involving complex and advanced machinery are never easy to perform due to the challenging and complex interacting environment. Norway is one of the leading players in the world in maritime industry. The rough conditions of the Norwegian Sea area forced the development of many specific task-oriented and innovative offshore solutions, such as anchor handling, pipe laying, deep sea drilling, launch and recovery of remotely operated tools and vehicles, etc. A great deal of resources and efforts are put into investigations, R&D, testing, training and maintenance of offshore machinery systems. For companies developing and supplying such equipment and services to the offshore industry, the degree of success is highly dependent on the reliability of the delivered systems in operation. This comprises both operability and safety functions, and hence the control systems need to go through extensive testing before being installed in new or existing systems. However, the possibility of testing a system in full scale at the work

site is often both limited and expensive. Thus, a lot of tests have to be performed on-board a vessel or a rig, which in such cases limiting the scope of work and time used for testing is important since the delivery time to market is another key factor of a successful product.

Hardware-in-the-loop (HIL) simulation is a form of real time simulation that used for developing and testing of complex embedded systems. Since the last two decades, HIL simulation has becoming increasingly prevalent in many industries driven by the complexity of advanced systems and the costs of both time and capital in building and testing on real systems. The general concept of HIL simulation is depicted in Figure 1. The physical system is replaced by a simulator by reproducing its behavioral characteristic and response to external commands by the use of a mathematic model. During testing the simulator, the target control system should not experience significant difference from being connected to the real system (DNV 2011). For systems that are intended to be visually observed, visualization of the simulator should also be included. With a 3D visualization window the simulator can also be used for operator training purpose, which is an indispensable part during product development in offshore industry as sea testing is both time consuming and expensive to carry out.

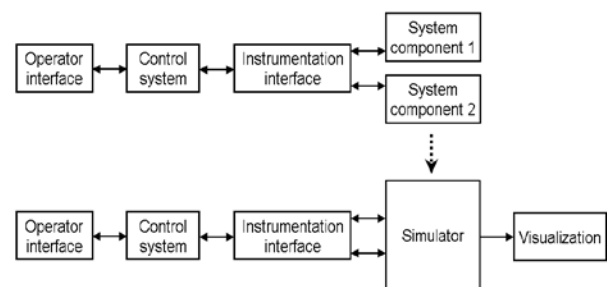


Figure 1: Concept of physical testing & HIL testing

HIL modeling methods and platforms vary from case to case depending on the physical systems and their control systems. Some examples of HIL simulator can be found for different applications including Hydro Power Plant Governor (Zaev, E 2012), Sequential Turbocharging System (Jianwei Du 2007), Engine Electronic Control

Unit (Cebi, A 2005). The project presented in this paper was carried out at the Aalesund University College in close cooperation with Rolls-Royce Marine. The objective is to develop a simulator for testing software and hardware in the control system for the Launch and Recovery System (LARS) as shown in Figure 2. The LARS is used on vessels during subsea operations for handling ROVs from deck to working depth. The system mainly consists of a power unit, a winch and an overboarding unit, which is represented by a traditional A-frame here.

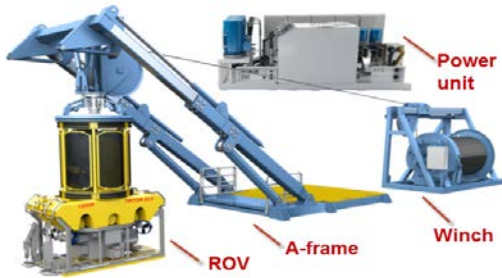


Figure 2: Rolls-Royce LARS for ROVs

The main contributions of this project work lie in two aspects. First, the modelling method based on Bond Graph (E. Pedersen 2008) can be used for modelling of complex systems covering multiple domains, including mechanical, electric, hydraulic and thermal domains. The 20-sim software provides several toolboxes for system modelling and simulation. 20-sim 4C is used for generating C code to run on the real time simulator. Second, the chosen Bachmann M1 controller provides commonly used interfaces for communication with the control systems. Thus, the original control system of the target machinery can be kept identical for the simulator and the real physical system. For most of other HIL simulators found in existing literature, the numeric models and modelling program use block diagrams built in Matlab. Compared to Matlab, 20-sim is a more open software, which allows users to build customized models depending on their needs. Previous researches in multi-domain system modelling and simulation using 20-sim based on BGM can be found in thermal-hydraulic system (Aridhi, E. et al. 2013), electro-mechanical system (Batlle, C. et al. 2008), and mechatronics system (Kayani, S.A. et al. 2007), etc.

The rest of the paper is organized as follows: the architecture of the simulator is described firstly; then the modelling and simulator platform setup is explained; and last, the results of several simulation and testing of the LARS simulator are presented.

## LARS HIL SIMULATOR ARCHITECTURE

An embedded hardware solution running the simulator and control system as a stand-alone unit is built to test if the proposed concept would work as intended. The simulator architecture is depicted as in Figure 3. A

numeric model of the LARS is developed and implemented on the Bachmann M1 controller for real time simulation. The simulator is then interfaced with the existing control system of the LARS to receive commands and provide feedbacks representing the response of the LARS.

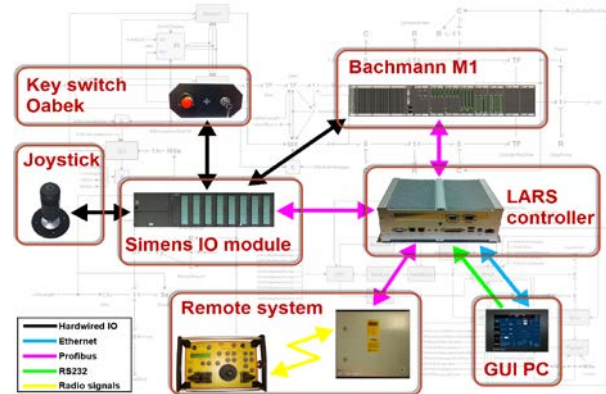


Figure 3: LARS HIL simulator architecture & communication topology

### 1. Hardware subsystems

To keep the interfaces between the control system and simulator as close as possible to the real system, all interfacing components used in the existing control system of the LARS are kept untouched (personal communication). This is done to maximize the validity of the results when using the simulator for control system testing. To reduce the physical size of the control system hardware, some signals normally transmitted by means of conventional IO were transferred to be transmitted over Profibus via Simens IO module. By this approach, the differences between the control system software used for the real system and the simulator came down to mapping of signals between conventional IO and the Profibus interface. As both mentioned technologies for signal transmitting were already used in the existing control system, this provided for a flexible solution for testing, as which signals are transmitted through the different communication methods could easily be changed.

For communicating with an external control system the model must run on a real time operating system on a device equipped with the necessary interfaces. Bachmann M1 is a series of industrial controllers supported by 20-sim 4C offering a broad selection of CPU and IO modules, including conventional electrical IO and Profibus which are required for the simulator of the LARS HIL project.

The simulator is operated from the same two operator stations as the original control system: an operation panel and a radio remote system. The operation panel consists of a GUI PC and a prefabricated plate with emergency stop, alarm buzzer, key switch, and a

joystick. The graphical user interface used for the simulator is identical to what is used on the real LARS. The only difference between the GUI PC used for the simulator and the real LARS is that the GUI PC for the simulator also runs a Motion Reference Unit (MRU) simulator. The MRU simulator, mimicking the measurement of vessel motion data in 6 DOF, runs as a separate software application not affecting the system's graphical user interface. It is connected to the main controller using RS232 interface. The measured data of the ship motion is used by the control system when operating in active heave compensation mode.

The simulator communication topology and interface types between the hardware subsystems are illustrated as in Figure 3, including Conventional IO (black line), Ethernet (blue line), Profibus (pink line), RS232 (green line) and Radio signal (yellow line).

## 2. Software tools

The numeric model is built using Bond Graph method in the 20-sim software provided by the Controllab group. The 20-sim software includes a 3D animation feature which is used to build up a graphical model of the LARS for visualization based on shell models exported from 3D CAD software. 20-sim 4C is used to link data from the numeric model to the graphical model during runtime, providing for real time motion of the latter during simulation. It was also used to export the numeric model from 20-sim to the Bachmann M1 controller and to set up mapping between the model variables and the simulator IO interfaces. When using the Bachmann controller in an embedded solution, 20-sim 4C handles connections to the target, mapping of IOs, compilation of target specific code and logging and monitoring of system performance. It has the capability of displaying 3D animations created in 20-sim during runtime by fetching the required variables from the target when running the model.

## NUMERIC MODEL

The Bond Graph Method (BGM) is a modelling method to graphically describe the dynamics of a physical system by identifying the energy flow through the system. It can be used for modelling systems across different energy domains including the mechanical, electrical, hydraulic and thermal domains. The method uses a set of ideal elements to represent separate physical phenomenon's which can be combined to make models of complex systems. As the calculation power required in solving the equations of a model increases with its complexity, the level of detail must be adapted appropriately to the model's purpose to achieve a real time simulation without compromising the real world behaviour.

### 1. Modelling of the LARS

The Bond Graph model for the LARS is built up in main modules similar to its physical architecture including a

hydraulic power unit, winch and A-frame, as illustrated in Figure 1. Furthermore, these modules consist of sub-modules at varying levels, depending on the level of detail in which the component is modelled. The model includes the hydraulic and mechanical dynamics of the system, and in addition an interface to the control system is implemented. The hydraulic submodels are created based on the basic principles of fluid dynamics (ASSOFLUID 2007).

For the mechanical domain, models representing a ROV and the steel armoured umbilical connecting it to the winch were also created. The requirement for the model as regards to interfacing is defined by the interface of the control system software. As the size and complexity of the model sets, the requirement for processing power which affects the real-time performance, the level of detail for different parts of the model have been simplified to different extents.

The model is grouped in modules arranged similarly to the physical main components of the system, as illustrated in Figure 4.

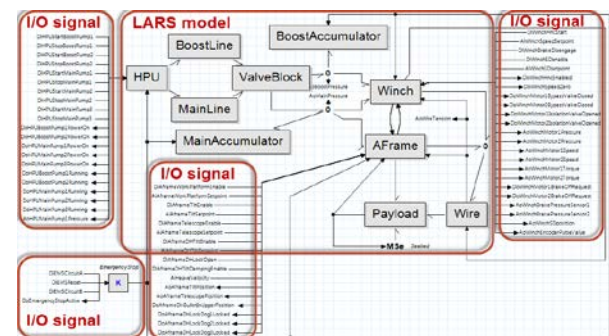


Figure 4: Bond Graph model overview

The HPU module provides hydraulic power to the system and feeds oil through pipe lines to a valve block, where the oil is distributed to the winch and A-frame. Accumulators are connected to the hydraulic system to cover rapid flow peaks, but are too small to store and release significant energy during active heave compensation. The winch and the payload are connected through an umbilical, and hence the winch handles the payload when not resting on other physical objects such as the seabed or vessel deck. The A-frame is used for handling the payload when docked in to the docking head lock, through which the umbilical between the winch and payload is routed. Position of the A-frame thereby affects the umbilical routing, contributing to handling the payload also when the latter is undocked. As a result, the A-frame has power bonds connected to both umbilical and payload. Vessel deck and seabed are included as vertical supports connected directly to the payload and releasing tension from the umbilical upon contact. Vessel motion affects the positioning of the system and hence the umbilical outlet and is included in the A-frame. All signals listed are included to support

the interface required by the control system. A signal based emergency stop circuit is connected to the main components activating certain safety functions.

For each submodel (block) of the LARS, the components are modelled as bond graph or equation submodel describing the energy flow through the system. Details of the submodels are not shown due to the limitation of the paper size.

## 2. 3D visualization

For visual presentation of the LARS, a built-in 3D animation toolbox is provided in 20-sim. 3D animation toolbox is a 3D visualization tool which can be used to create simple three dimensional figures or import detailed shell models in the STL format, which is supported by most 3D CAD tools. The 3D animation concept is built up using reference frames to achieve the movements of physical parts in relations with each other in a hierarchy nestled to a common base reference frame. All imported and created parts used for visualization has a dedicated reference frame as illustrated in Figure 5. Each frame can be assigned with fixed or variable positioning and alignment to its parent frame. Variable positioning or alignment can be linked to any parameter or variable in the numeric model of the simulator during runtime.

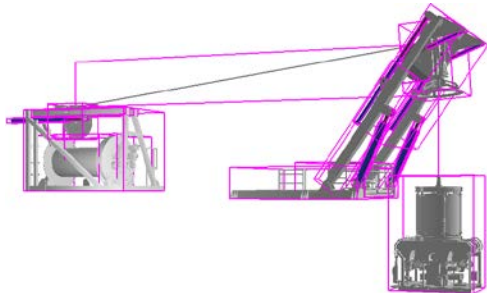


Figure 5: LARS reference frame setup

## SIMULATOR SETUP

As the setup is interfaced with the original control system of the LARS, it provides for flexibility as to physical realization of the interfaces and can vary from including just a few of the main parts communicating on ProfiBus and Ethernet to a full scale control system with all IOs connected. The components are fit into a compact cabinet with joystick and GUI PC mounted on the front door (Figure 6). For testing modifications of the LARS as to mechanical and hydraulic design and how this affects the system including the control system, a simulator in its simplest form with only the least required interfacing is needed. The same goes for testing the control system software, unless testing is related to handling of errors in the electrical hardware. For testing electrical hardware the physical interfaces can be set up to include the necessary components.

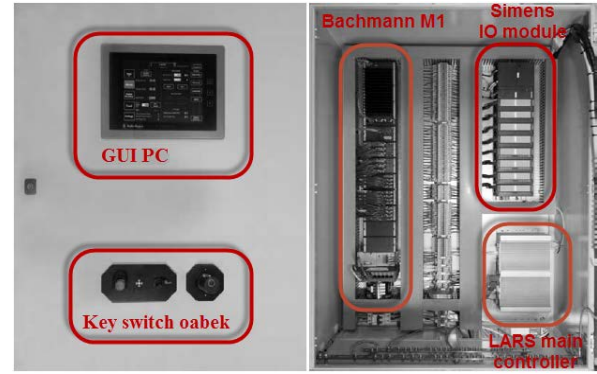


Figure 6: Simulator set up.

## MODEL BEHAVIOR AND SIMULATION

To verify that the simulator performed as intended, various tests related to behavior and performance were executed. The main task for the LARS is handling ROV between vessel deck and sea surface. During launch and recovery when the ROV is locked into the docking head, the system is normally in Latched mode. This is a mode where the control system runs the winch based on the tension measurement in the umbilical to maintain a constant tension of approximately  $2T_e$ . This allows for the operator to manoeuvre the A-frame without considering the winch simultaneously to avoid slack or over-tensioning the umbilical. The umbilical tension has to be below  $2T_e$  to be allowed to set the winch to Latched mode. The tests presented in this chapter were performed using full hydraulic power with all pumps running, while the tilt cylinders of the A-frame were used to move the A-frame in and out.

Prior to the test the A-frame was placed over the vessel side in overboarding position 4, which means 4 m from zero position in positive direction. The overboarding zero position is defined as the position of the docking head lock when the A-frame telescope structure and the docking head is aligned at 90 degrees vertical. Towards the vessel centre is defined as negative direction while over the vessel side is defined as positive direction as illustrated in Figure 7.

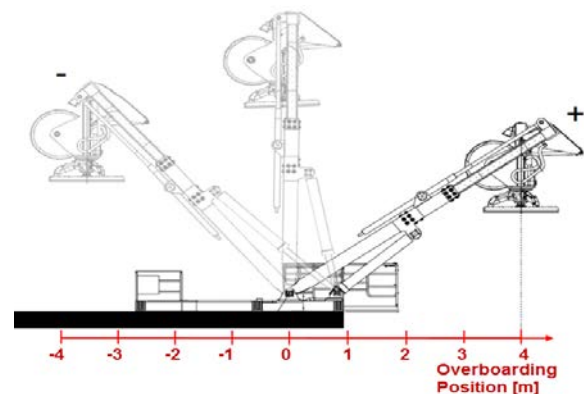


Figure 7: A-frame docking head overboarding position



The test started with the winch in normal mode, which means its speed is controlled by a joystick. Next, the winch was run inwards until the payload was latched into the docking head lock. Once latched, the winch was run out in order to transfer the weight from the winch (umbilical) to the docking head lock (A-frame). When the winch loads drops below 2 Te, latched mode was enabled and the A-frame tilt was operated to move inwards until the entire payload was over the vessel deck. The winch was then set back to normal mode and the payload was lifted off the docking head lock. The lock was then opened and the winch was run outwards until the payload was landed on deck. Plots of cylinder pressure, umbilical tension, winch speed and ROV position are shown and explained in next Figure 8.

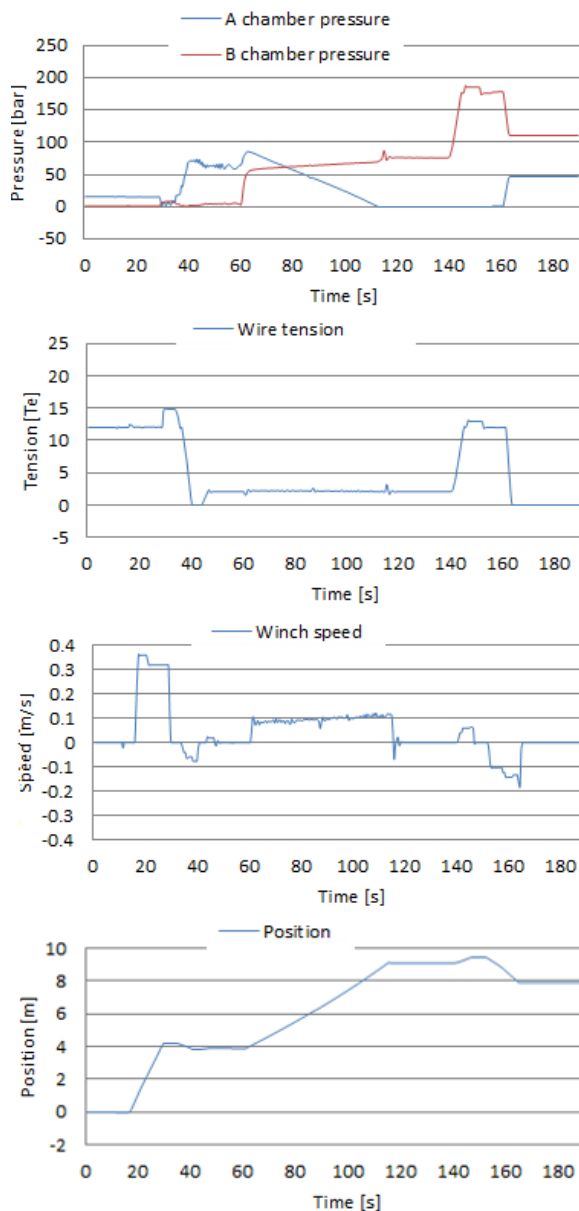


Figure 8: Plotting of cylinder pressure, umbilical tension, winch speed and load position during simulation

Initially, the system was run without any input for about 20 seconds. Then the joystick was used to hoist the ROV as seen in the winch speed and position plots. The ROV was lifted about 4m before it comes into contact with the docking head at approximately 30s. This contact will then try to move the A-frame inwards, thus the tilt cylinders went from being compressed to being stretched as seen in the pressure plot from one of the cylinders. The hydraulic pressure rose in chamber B and fell in chamber A until the docking head lock was closed and the winch lowered the ROV onto the lock. For the tilt cylinder, this leads to the pressure of chamber A settling at a higher level than when only supporting the weight of the A-frame only. The peak in tension around 30s was caused by the ROV coming in contact with the A-frame. The umbilical tension fell rapidly to zero when the ROV was landed on the docking head lock as the weight of the ROV was transferred to the A-frame. When initiating Latched mode, the winch is automatically paying in slowly until the tension in the umbilical reaches approximately 2Te as can be seen in the winch speed and umbilical tension plots. When moving the A-frame inwards, the pressures in both tilt cylinder chambers increased then the A chamber pressure slowly decreased towards zero while the B chamber pressure slowly increased until the A-frame movement stopped at about 120s. The tension plot verifies the control system's constant tension controller was able to maintain an umbilical tension of 2Te while operating the A-frame. From 140 to 150s, the winch was hoisted to lift the ROV off the docking head lock. With the ROV being suspended in the umbilical, the lock was opened and the payload slowly lowered to the vessel deck.

One of the selling points of this LARS is the Active Heave Compensation (AHC) function which expands the operation window in rough weather conditions and thus reduces the potential cost of operation downtime. To test if the model would be able to run in AHC mode within the given range several tests were performed under different wave conditions.

The following pictures show the ROV position during AHC mode with for two different wave motion conditions. In case one (Figure 9), the wave height and period are 2.4m and 5s, respectively. AHC mode started after about 10s and after another 5-10s the heave compensation is fully operational and the ROV position is steady. In case two (Figure 10), the wave height and period are increased to 15m and 10s, respectively. According to the plotting results, this is beyond the capacity of the system and it thus becomes important that the system do not fail, but maintains to compensate within its capability.

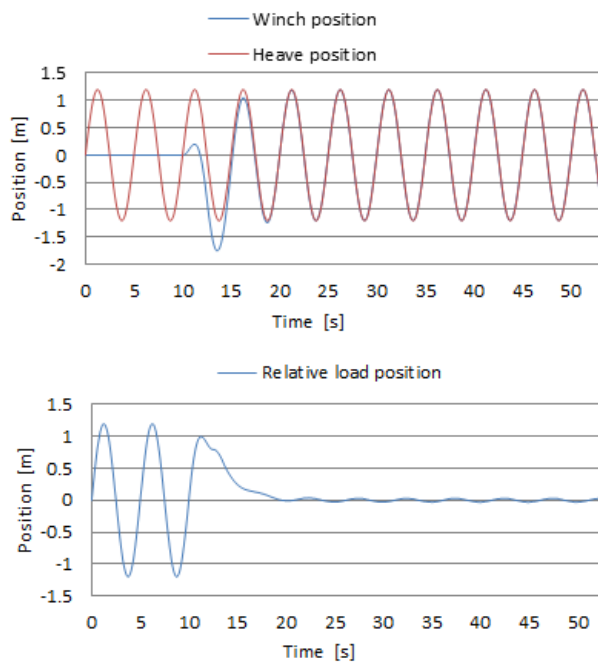


Figure 9: Load position under AHC mode with 2.4m wave height and 5s period

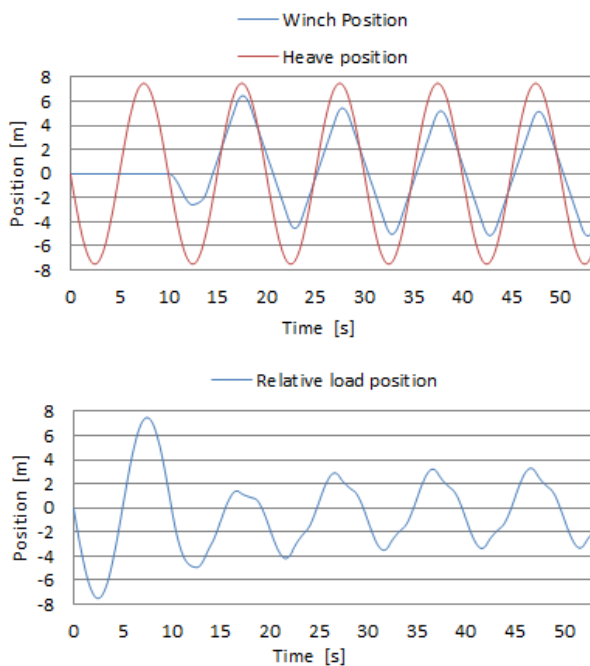


Figure 10: Load position under AHC mode with 15m wave height and 10s period

The control system warned the user about the inability to fully compensate, and the residual motion of the ROV is significant. To verify the control system still works as intended even without being able to compensate the heave, a plot of the winch speed is shown (Figure 11). The speed required for full compensation would be close to 6 m/s, while the actual maximum winch speed is 2.4 m/s.

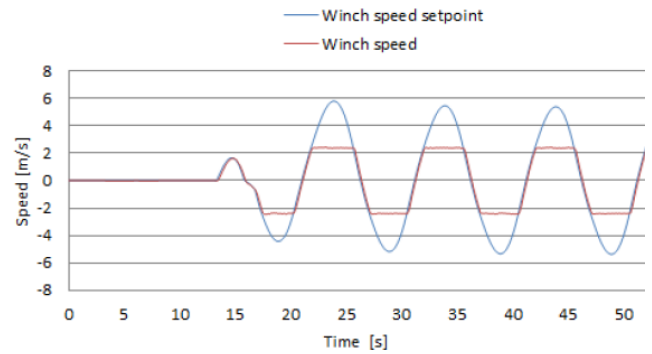


Figure 11: Winch speed under AHC mode with 15m wave height and 10s period

Besides the plots above, all the parameters and variables in the numeric model can be plotted and connected to the 3D model for animation. The model includes the main components of the LARS thus the characteristic response of the system, in particularly the hydraulic performance, can be monitored during and after runtime. Within the field of training, this setup is well suited as it provides for holding courses for troubleshooting as well as operations (Figure 12).

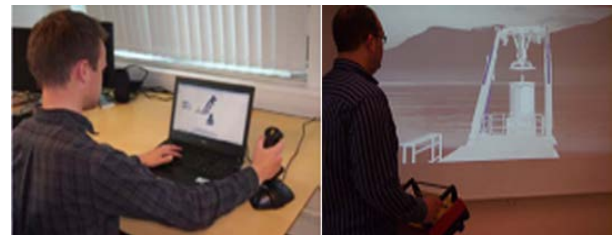


Figure 12: Real time simulation with 3D visualization using joystick, keyboard and radio remote

## CONCLUSION

Results from testing the simulator shows the data acquired during simulation runtime corresponds well to expected values related to the performance of a real LARS. This indicates the simulator is well suited for testing of control systems as the response of the system is important when testing control loops and also error handling. The concept developed is modularized similarly to the real system, and utilizes a flexible interface between the simulator and the control system with respect to scalability and interface technology. This means similar simulators can be developed for other systems using the same concept following the same methods as used in this project.

Despite the capabilities of 20-sim as regards to displaying 3D-animation being very good for development, testing and demonstrations, it has some limitations in terms of visual quality. If a simulator should be presented and sold as a commercial product, the visualization should be made more realistic with possibilities to add effects like for instance different

weather scenarios. Especially for an operator training simulator the looks are important to create an environment which feels as close to reality as possible. This can be solved by letting a more advanced graphical motor render the graphics. All parameters, variables and signals in a running model are available for monitoring both in Bachmann Solution Center and 20-sim 4C which communicates with the controller through Ethernet. Hence, setting up an UDP stream from the M1 controller's Ethernet interface with data required by a prospective 3<sup>rd</sup> party graphical motor can be a way to solve the link to higher quality graphics.

## ACCKNOLEDGEMENT

The project was carried out as the authors' thesis assignment of their master study program at Aalesund University College (AAUC) in the cooperation with Rolls-Royce Marine. The authors wish to express their gratitude to the Controllab and Bachmann group for their time and support in performing the usability test. Special thanks go to Prof. Houxiang Zhang from Aalesund University College for his contributions to the paper.

## REFERENCES

- Aridhi, E.; Abbes, M.; Mami, A. 2013. "Pseudo bond graph model of a thermo-hydraulic system". In *Proceedings of 2013 5th International Conference on Modeling, Simulation and Applied Optimization* (Hammamet, Apr.28-30), IEEE, Hammamet, 1-5.
- ASSOFLUID, Italian Association of Manufacturing and Trading Companies in Fluid Power Equipment and Components. 2007. *Hydraulics in industrial and mobile applications*.
- Battle, C.; Dòria-Cerezo, A. 2008. "Bond graph models of electromechanical systems. The AC generator case". In *Proceedings of 2008 International Symposium on Industrial Electronics* (Cambridge, Jun. 30-Jul. 02), IEEE, Cambridge, 1064-1069.
- Cebi, A.; Guvenc, L.; Demirci, M.; Karadeniz, C.K.; Kanar, K.; and Guraslan, E. 2005. "A Low Cost, Portable Engine Electronic Control Unit Hardware-in-the-Loop Test System" In *Proceedings of the 2005 International Symposium on Industrial Electronic* (Dubrovnik, Jun.20-23), IEEE, Dubrovnik, 293-298.
- DNV. 2011. *Hardware in the Loop Testing*, Standard for certification No.2.24. DET NORSKE VERIT AS, Høvik. (Jul).
- E. Pedersen; and H. Engja. 2008. "Mathematical Modeling and Simulation of Physical Systems". Lecture notes in course TMR4257 Modeling, Simulation and Analysis of Dynamic Systems, Department of Marine Technology, Norwegian University of Science and Technology. Trondheim. (Aug).
- Jianwei Du; Yinyan Wang; Chuanlei Yang; and Hechun Wang. 2007. "Hardware-in-the-loop Simulation Approach to Testing Controller of Sequential Turbocharging System", In *Proceedings of the 2007 International Conference on Automation and Logistics* (Jinan, Aug. 18-21), IEEE, Jinan, 2426-2431.
- Kayani, S.A.; Malik, M.A. 2007. "Automated Design of Mechatronic Systems using Bond-Graph Modeling and Simulation and Genetic Programming". In *Proceedings of 2007 International Bhurban Conference on Applied Sciences & Technology* (Islamabad, Jan. 8-11), IEEE, 104-110.
- Zaev, E.; Tuneski, A.; Babunski, D.; Trajkovski, L.; Nospal, A.; and Rath, G. 2012. "Hydro power plant governor testing using hardware-in-the-loop simulation" In *Proceedings of the 2012 Mediterranean Conference on Embedded Computing* (Bar, Jun. 19-21), IEEE, Bar, 271-274.

## AUTHOR BIOGRAPHIES

**Johnny Aarseth** studied Product and System Design (M.Sc) at Aalesund University College. He now works as Design Manager at Rolls-Royce Marine AS.

**Alf Helge Lien** studied Product and System Design (M.Sc) at Aalesund University College. He works as Senior Project Engineer Automation at Rolls-Royce Marine AS.

**Øyvind Bunes** received his M.Sc in Marine Technology from the Norwegian University of Science and Technology in 1996. Until 2002 he was employed as a researcher at MARINTEK until he joined ODIM, later Rolls-Royce, in 2002. He now serves as Senior Principal Engineer at Rolls-Royce Marine AS. He is also an Associate Fellow in the Rolls-Royce Engineering Fellowship and an Assistant Professor at Aalesund University College.

**Yingguang Chu** received his master degree in Product and System Design from Aalesund University College in 2013. He continues as a research assistant and Ph.D. candidate at Aalesund University College.

**Vilmar Æsøy** received his Ph.D. in Mechanical Engineering in 1996 from Norwegian University of Science and Technology. From 1997 to 2002 he worked as reasearcher in Aker Maritime and R&D manager in Rolls-Royce Marine As. Since 2002 he works as an Associate Professor at Aalesund University College.