Multi-physics Co-simulation of an Electric Reach Truck

Abid Abdul Azeez*, Viacheslav Zakharov*, Abdullokh Orifjonov*, Jesper Niemelä* and Tatiana Minav*

Tampere University, IHA - Innovative Hydraulics and Automation, Korkeakoulunkatu 6, 33720 Tampere, Finland*

E-Mail: abid.azeez@tuni.fi

On-board intelligent predictive maintenance tools to avoid sudden breakdown of components can increase the reliability of a system. It is often difficult to anticipate failure modes and to fabricate faults in an actual working system for the developing of condition monitoring. In this paper, a multi-physics co-simulation model of an electric reach truck is proposed as a tool for generating training data for AI based condition monitoring. The co-simulation is established using three different software platforms: Mevea for real-time multi-body dynamics and animation, AMESim for hydraulic components, and Simulink for control and electric drive model. The co-simulation model is successfully validated by comparing the simulation and measured data.

Keywords: Digitalization, condition monitoring, co-simulation, reach truck, induction motor, electric drive, vector control

Target audience: Mobile Hydraulics, Design Process, Condition Monitoring

1 Introduction and Background

Electric reach trucks are widely used in narrow aisle warehouses to carry and stack goods. Working hydraulics is an integral subsystem of the reach truck. However, hydraulic components (in addition to electric components) can be subject to failure due to wear, contamination of oil, or manufacturing errors. The typical hydraulic failure scenario of a reach truck includes leakages in various components such as hoses, pumps, cylinder, and valves. This affects the performance of the reach truck and can be detected by monitoring the pressures at actuator ports [1], vibration response [2], friction torque [3], force [4], the motor speed requirement that increases when there is an internal leakage in the system, and rising motor current due to abnormal friction [5]. Thus, an on-board preventive/predictive maintenance tool – that is implemented on a PLC or edge controller – to avoid sudden breakdown of components or excessive leakages that can increase the system reliability and reduce downtime by indicating faulty conditions to an operator or to a supervisory level cloud like SCADA should be implemented. Conventional condition monitoring uses sensors extensively [6] to monitor a system in real time utilizing data that contains information regarding several failure modes [7]. Majority of novel condition-monitoring methods are based on advanced control such as stochastic observers [8] and artificial intelligence (AI), such as neural networks (NN) and machine learning [9]. One of the most utilized advanced condition-monitoring methods is model-based condition monitoring. This class of condition monitoring utilizes dynamic mathematical models of components or whole systems, and data from sensors for determination of real-time system states. As an example, model-based condition monitoring of a hydraulic servo valve is represented in [10]. In this study, pressure, spool-position and coil-current sensor signals are used to determine several types of valve fault. Another example utilizes flow and pressure sensors in a hydraulic system and a mathematical estimator for calculating further system states, which are required for the valve-condition evaluation [11]. Such a method uses complex mathematical calculations. In [12], complex mathematical models are avoided by utilizing neural networks and/or fuzzy logic for condition monitoring; however, extensive training data is required in such a case.

Overall, modern AI-based condition monitoring methods are data hungry. Consequently, labelled data of healthy and faulty conditions is required extensively. It is often difficult to anticipate failure modes and to fabricate faults in an actual working system to generate data for pre-set conditions. To overcome this challenge, various simulation software has been explored widely. An example utilizing this approach is reported in [13], training data is obtained from the simulation model of an electro-hydraulic servo system based on linear regression, which was validated using a test rig. According to [13], that model-based data can be used for training AI-based condition monitoring.

Drawbacks of this approach are the need for a high-fidelity model definition, long simulation times and tuned simulation solver parameters.

Accurate and close to real condition simulation of hydraulic parts and systems can be realized by dedicated software such as Simcenter AMESim and MATLAB. AMESim is often used in hydraulic simulation tasks. Authors in [14] demonstrated utilization of the capability of AMESim simulating mechanics and hydraulics for hydraulic faults modelling . However, for certain domains such as control signal flow other software tools might be more convenient than AMEsim. Also co-simulation allows to combine the merits of different software tools. Such an example of combining AMESim and MATLAB is represented in [15].

Most of the existing papers describe condition monitoring of separate components or sub-system only. However, monitoring the whole system is not well investigated and it is required for further development of electrohydraulic systems and detection of multi domain faults. This paper builds on the methods established in [5] and investigates modelling capabilities for the establishment of multi-domain fault detection and classification methods. Exemplarily, this paper presents a tool based on multi-physics co-simulation that creates training data for condition monitoring of an electro-hydraulic reach truck. The reach truck model developed in this paper focuses on the main components of the reach truck and its lifting function. A co-simulation model of an electric reach truck for lift function is modelled using three complementing software tools: Mevea modeler for real-time multi-body dynamics simulation and animation, Simcenter AMESim for 1D simulation of the hydraulic system and MATLAB/Simulink for control and simulation of the Electric motor (EM).

The co-simulation model is validated by comparing the simulation and experimental data. The co-simulation model closely represents the actual working conditions of the electric reach truck and will provide valuable data for further investigation and implementation of condition monitoring. The remainder of the paper is organized as follows: The reach truck Co-simulation model is introduced in Section 2. Section 3 describes the utilized experimental setup for this study case in detail and data validation. Finally, Section 4 contains concluding remarks.

2 Methodology

Condition monitoring is to be implemented by utilizing data that is generated with the proposed tool. The co-simulation structure is depicted in **Figure 1**. Mevea software is assigned as the controller, while both MATLAB/Simulink and AMESim are agents.

Mevea and AMESim are connected through a shared memory area and a functional mock-up interface (FMI). The purpose of the FMI standard is to allow simulation models to be shared across different software platforms [16]. Mevea and AMESim are connected through a functional mock- up unit (FMU) which is a zip archive that contains XML-files and compiled C-code. The C-code contains the FMI that is created in AMESim, before it is wrapped in the FMU and exported to Mevea. The FMU model serves as an agent software for the controller software to which the FMU is connected. Mevea and Simulink are connected through a TCP/IP socket interface. Mevea modeler provides interface as C files which are compiled in Matlab to mex-files. The mex-files work as Matlab functions which are used in Simulink blocks to transfer data between Simulink and Mevea. The solvers of Mevea, MATLAB/Simulink, and Amesim uses Runge Kutta method with a 0.2 ms, 0.1 ms and 0.05 ms timestep respectively for solving ordinary differential equations.

The Mevea multibody dynamics model is introduced in section 2.1. Section 2.2 presents the electric drive model realized in MATLAB/Simulink and the hydraulic AMESim model is introduced in section 2.3.

2.1 Multibody dynamics model

The multibody model of the reach truck is created in the Mevea Modeler software as shown in Figure 2. It contains ready-made components with a list of parameters that are used to define their properties such as friction coefficients, stiffness, or inertia. The components are used to define the kinematic chains that are actuated by the hydraulic actuators. The rigid body geometries and inertia data are imported from a CAD model of the reach truck. The reach truck consists of chassis, mast assembly with three-part mast, forks-carriage, forks, lifting chains, pulleys, and hydraulic cylinders.

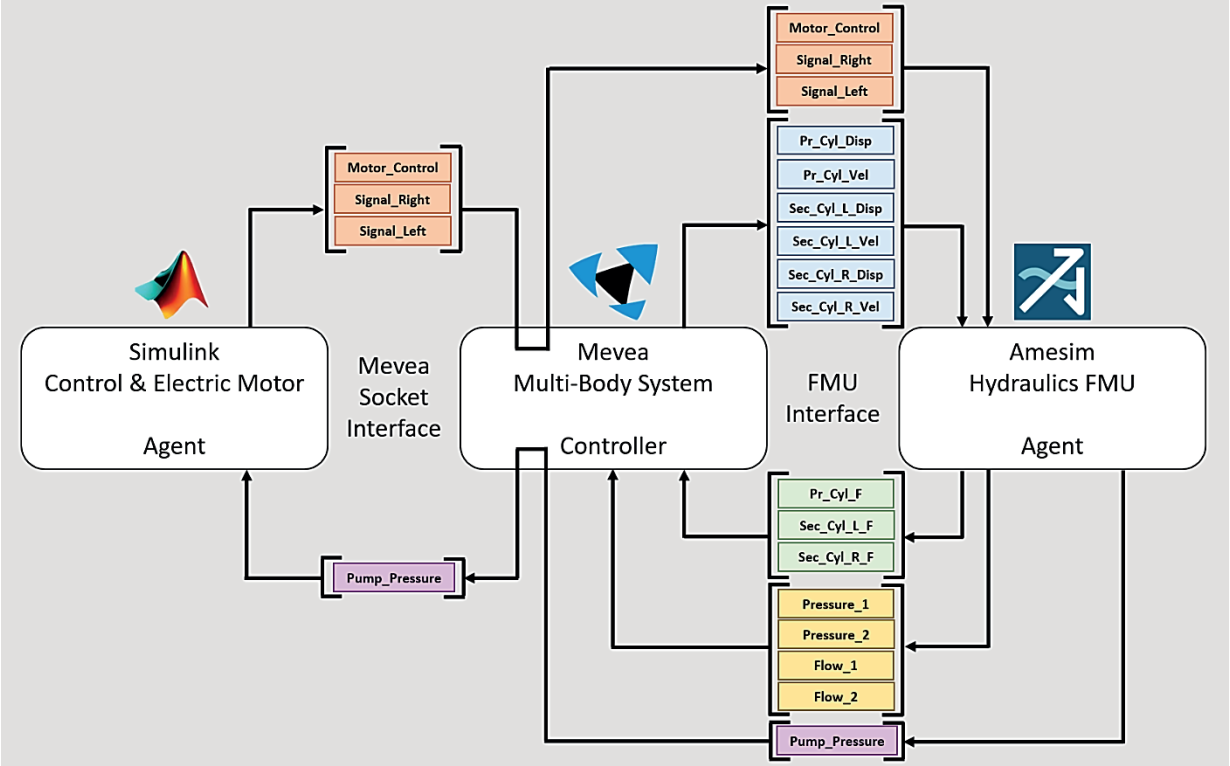


Figure 1: Co-simulation setup with Mevea as the controller; Simulink and AMESim are agents [11]

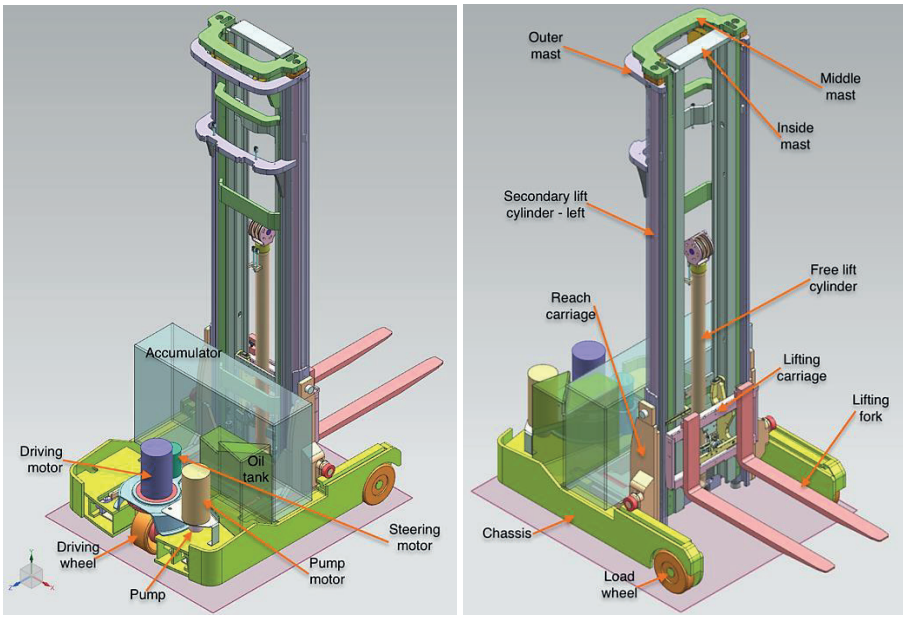


Figure 2: CAD assembly of the electric reach truck.

The kinematic chain of the reach truck is presented in **Figure 3**. Bodies and constraints define the kinematic chain which is actuated by components that contribute to load and are connected to the hydraulic model. Dummies define mass and inertia for hydraulic cylinders. A simple drivetrain is modelled for the reach truck as well, but it is not investigated in this paper. The lifting mechanism, which is the focus in this paper, consists of three-part mast – outer, middle, and inner mass as depicted in the Figure 2 above,– chainwheel, lifting carriage and forks and it is driven by three hydraulic cylinders. Rollers between inner and middle, and between middle and outer mast enable frictionless vertical movement of inner and middle masts, whereas the outer mast always remains fixed. The primary lifting cylinder drives the chainwheel in relation to the inner mast and two secondary lifting cylinders drive the middle-mast in relation to the outer mast. Chains are used to connect the inner mast to the carriage and the outer mast to the inner mast via pulleys. This has the effect of the primary cylinder indirectly driving the lifting carriage as well as the forks and the secondary cylinders indirectly driving the inner mast.

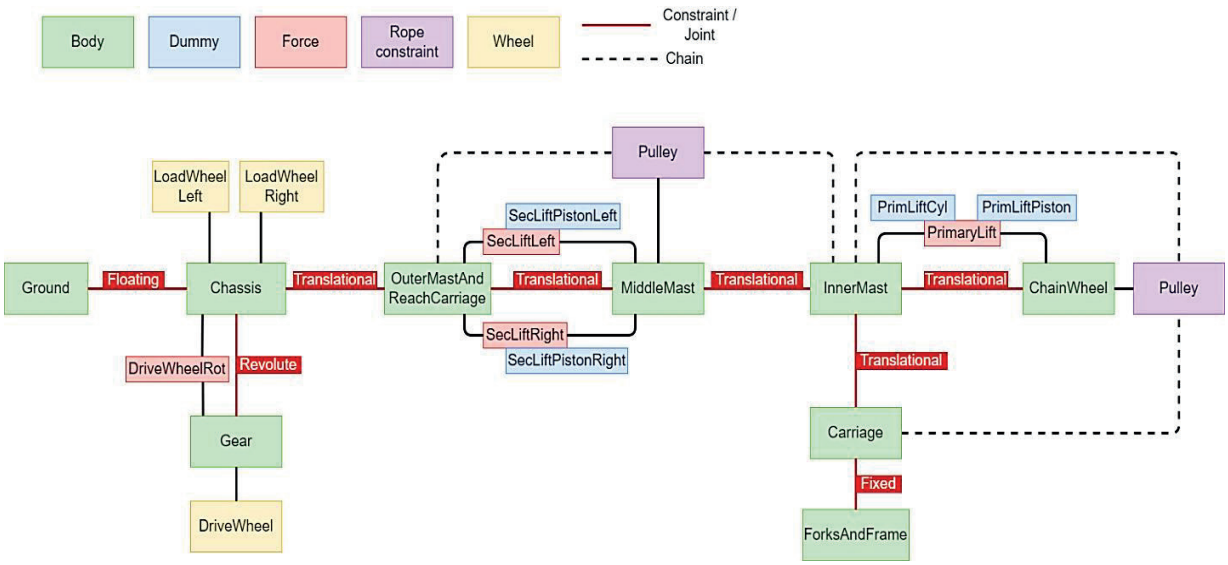


Figure 3: Kinematic chain of reach truck Mevea model

2.2 Electric drive model and control

The model of the EM as a part of electric drive and its control is created in MATLAB/Simulink software. The EM, a 14-kW induction machine, is used to drive the hydraulic pump at a variable speed. The EM is modeled via differential equations. A detailed description of utilized equations is represented in [17]. The EM is fed by the frequency converter, which transforms the AC from one frequency and amplitude to AC with other parameters. Such transformation is the main principle of electric motor control. The battery is not considered in this research and consequently the operation is simplified by excluding DC-AC conversion which is typical for Electric reach truck. Amplitude and frequency of the feeding currents and voltages are calculated at every moment with the help of a negative-feedback control system. Vector control is used as a control system for the induction motor. Frequency converter, induction motor, and control system are built via MATLAB/Simulink blocks. The structure of the model is depicted in **Figure 4**.

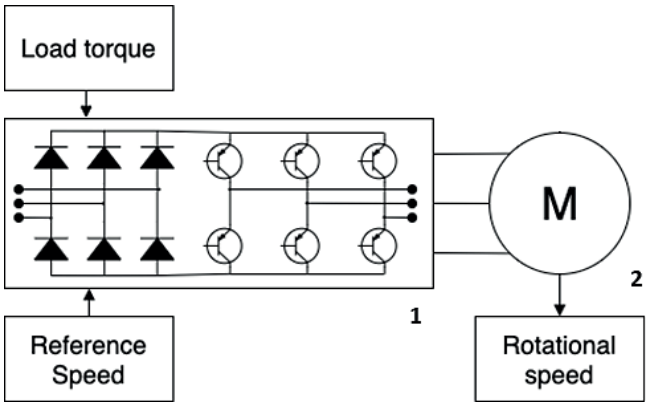


Figure 4: Structure of electric drive model (1-Inverter,2-Electric Motor).

The model of the electric drive has two input variables, a load torque coming from the hydraulic AMESim model and a reference speed as a control input for the inverter. The output of the model is the simulated EM speed. Both the digital co-simulation model and the physical reach truck are controlled with the same on-board joystick. The physical reach truck is operated, and the flow, pressure, height, and joystick control values are measured with a data acquisition system presented in Chapter 3. The same measured joystick control values run the digital co-simulation through the Simulink as illustrated in **Figure 5**. Joystick control value is interpreted to electric motor speed in rpm reference value and valve reference values for the hydraulic model in the controller block.

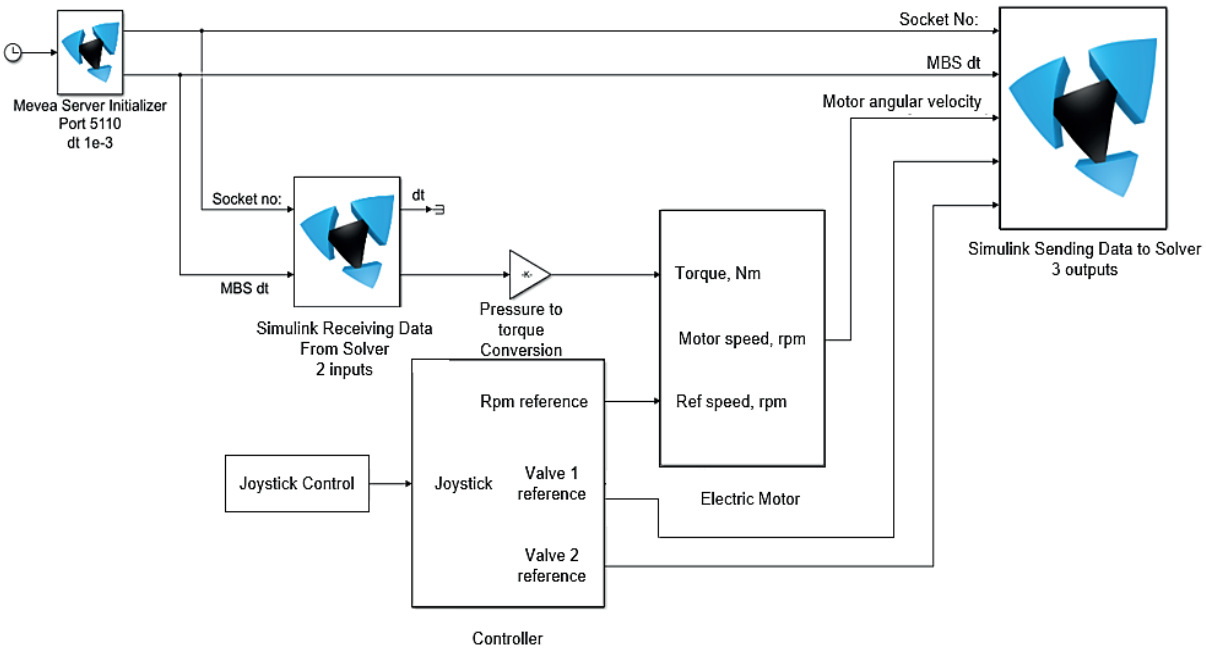


Figure 5: Simulink model (electric drive control and co-simulation control) and Mevea Solver multi-body simulation model run simultaneously.

2.3 Hydraulic Model

Simcenter AMESim is used to develop a one dimensional hydraulic simulation model by the means of pre-defined hydraulic components. The simulation model developed in AMESim is illustrated in **Figure 6**; it represents the actual hydraulic setup on the reach truck. The system consists of three cylinders; one primary cylinder (PC) and two secondary cylinders (SC). Cylinder diameters and kinematics are such that the primary cylinder is driven on lower pressure than the secondary cylinders. Therefore, the primary cylinder is extending to its end position before the secondary cylinders start moving; for retraction/lowering, the sequence is reversed.

The prime mover is the EM that is modelled in MATLAB/Simulink. EM is connected to a fixed-displacement pump with an orifice in parallel, which models internal leakage gaps. For the cylinder extension movements, the hydraulically actuated 3/3-directional valve is switched to connect the pump to the cylinder, and pump control is used; for the retraction movement, on the other hand, the pump remains still, and the directional valve is opened continuously for valve-control of the movement. A pressure gauge is assuring that the pressure drop across the directional valve remains constant during lowering, which makes the valve control load independent. The check valves prevent the pressure gauge from influencing the extension movement, which is pump controlled. To achieve control of the lifting and lowering speed, the operator signal, coming from a joystick, is used to control the electric motor speed for extension and the valve opening ratio for retraction. Furthermore, a relief valve limits the maximum pump pressure is modelled as well.
To generate co-simulation capabilities, a functional mock-up interface (FMI) is created in AMESim to send and receive information between the simulation models that were built using different software. FMI is a standardized way of defining a model and its interface in one container. It includes an XML file which contains a description of the model, and the model itself is defined in C language, compiled for the target computer.

The FMI with eleven inputs and nine output variables is illustrated in **Figure 7**. The input variables of the FMI are generated by AMEsim and are fed to Mevea. The output variables are generated by Mevea and Simulink respectively and are fed to AMEsim. Motor_Control, Control_Right and Control_Left are the output variables generated by Simulink while the remaining output variables are generated in Mevea. The following section will introduce the experimental setup utilized for validation of the proposed co-simulation model.

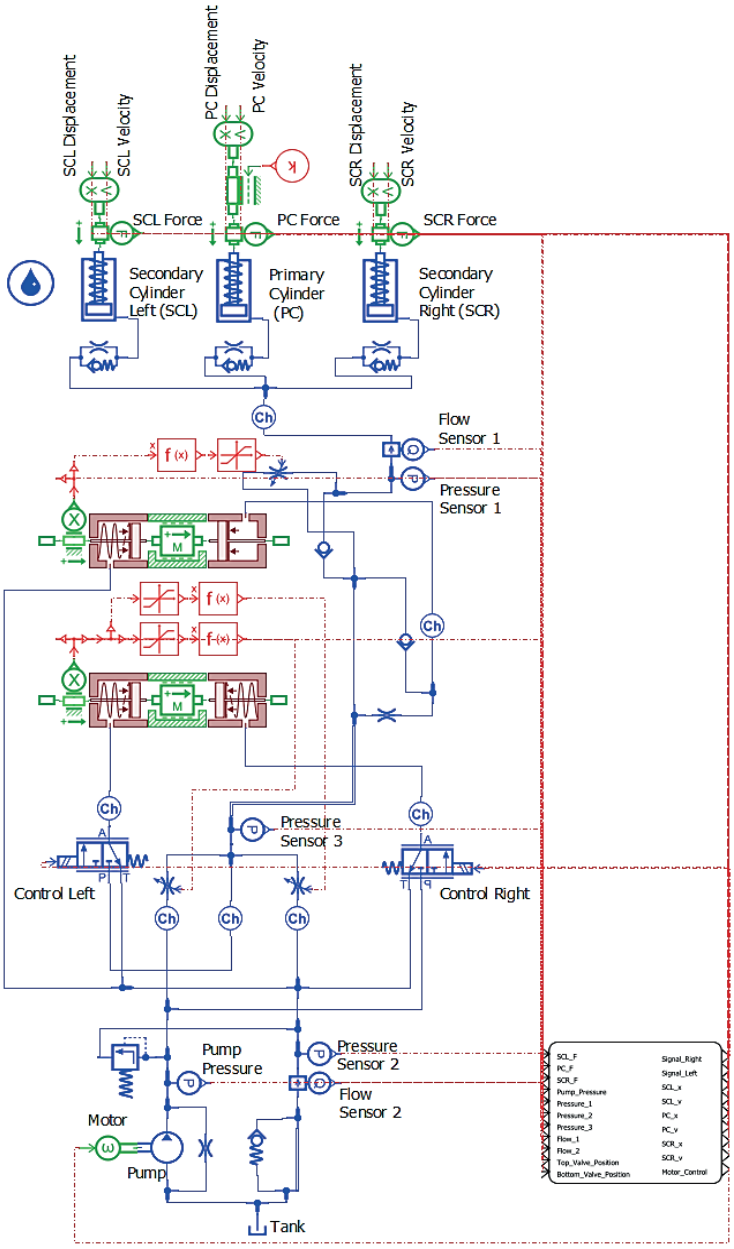


Figure 6: Simcenter AMESim Hydraulic Simulation Model.

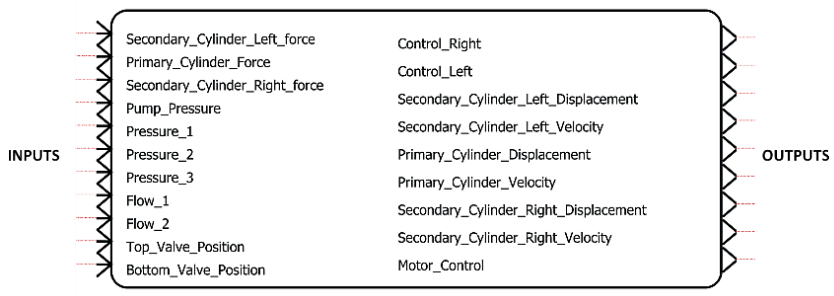


Figure 7: Functional Mockup Interface Unit (FMI). "Inputs" refer to the data gathered from Amesim model, while "Outputs" refer to the fed into the Amesim model.

3 Experimental Setup and Data Validation

The experimental setup consists of the original electric reach truck, added sensors, and a data acquisition system. **Figure 8** shows the photograph of the electric reach truck and the data acquisition system. The control and data acquisition schematics of the experimental setup are shown in **Figure 9**. The sensors added to the system include a height encoder, one pressure sensor, and two flow sensors. The height encoder is manufactured by SICK and has a

measuring range of 0-10 meters. The pressure sensor is manufactured by Trafag and has a measuring range up to 400 bar. The flow sensor is manufactured by Kracht and has a measuring range between 0 – 80 L/min each.

The data acquisition is based on a National Instruments CompactRIO 9045 alongside two NI 9229 C series modules with 4 channels each that are capable of simultaneous sampling at a rate of 50 kHz. The data acquisition hardware is programmed using LabVIEW software. A LabVIEW Field-Programmable Gateway Arrays (FPGA) module is used to take full advantage of the data acquisition module. The height sensor, two flow sensors, joystick control and pressure sensor value are recorded using the data acquisition system. The experiment is carried out under no load condition and with a sampling rate of 1 kHz. The duty cycle is lifting the forks of the reach truck and returning to home position at various velocities.

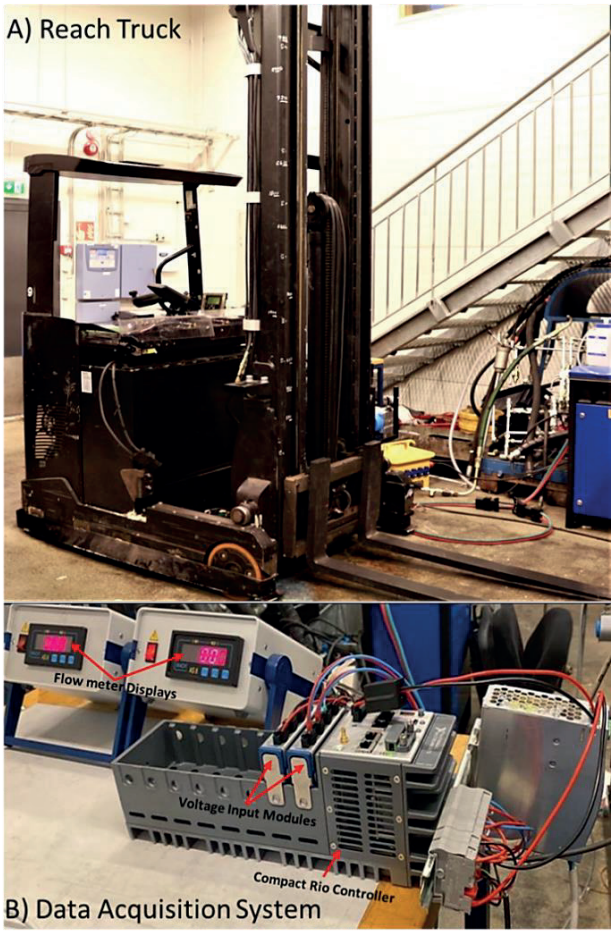


Figure 8: A) The electric reach truck, B) The data acquisition system

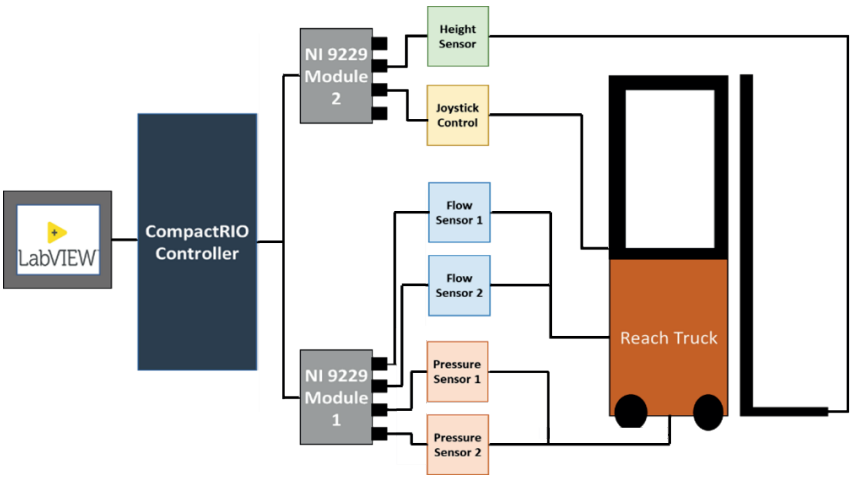


Figure 9: Structure of data acquisition.

The data validation is carried out by comparing the measured results with simulation for three different scenarios. Maximum, medium and varying velocity scenarios are utilized in this research. The height, flow, pressure, and measured joystick control parameters are plotted for each scenario. A positive amplitude in joystick control indicates lifting phase, and negative amplitude indicates lowering phase. The simulated and measured results for maximum velocity during lifting and lowering are illustrated in **Figure 10**. The simulated and measured results are well aligned for the height and flow. Noticeable oscillation in the pressure value before the lifting phase starts can be observed. However, the value stabilizes during the operating cycle. The oscillation at the start is due to the limitations in simulation settings which does not allow zero gauge pressure in the cylinder chamber. Overall, the simulation model performs well and reacts perfectly with the measured joystick control values.

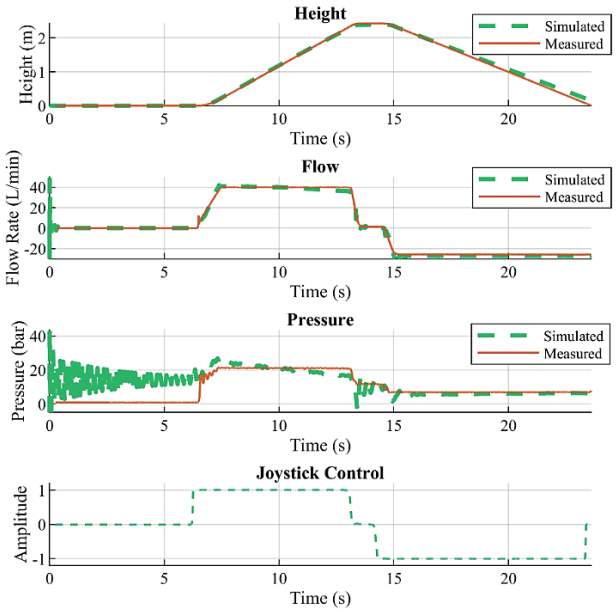


Figure 10: Measured and simulated results comparison for maximum velocity scenario

The simulated and measured results for medium velocity during lifting and lowering are illustrated in **Figure 11**. A slight deviation towards the end of the lowering phase for the height can be observed. Overall, the simulated and measured results are perfectly aligned for the height, flow, and pressure. The relationship between the joystick control amplitude and the flow rate is nonlinear and hence accurately tuning the co-simulation model would require plenty of data for various joystick control amplitude. Consequently, a slight acceptable deviation is present in the height.

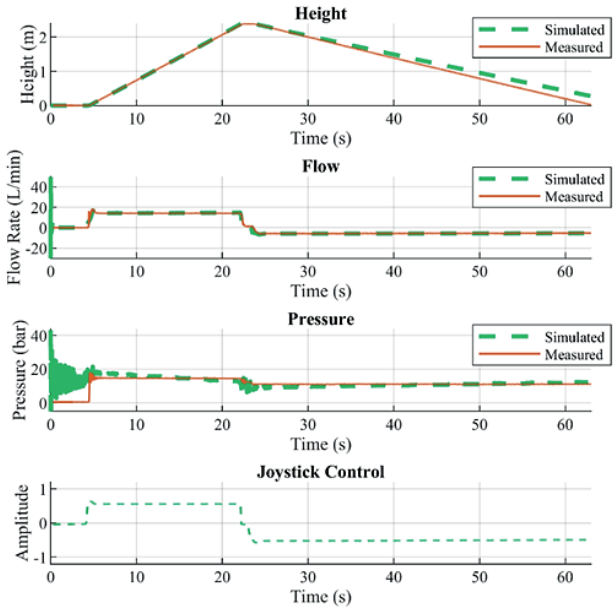


Figure 11: Measured and simulated results comparison for medium velocity scenario

The simulated and measured results for varying velocity during lifting and lowering are illustrated in **Figure 12**. The varying velocity is an excellent scenario as presentation of actual operating conditions, the operator will not operate the reach truck using constant velocity. Identical to previous tested scenarios, the measured and simulated results are aligned to each other for height, flow, and pressure signals. Only a slight deviation that is caused due to the nonlinear relationship between the joystick control amplitude and flow rate can be observed in Figure 12.

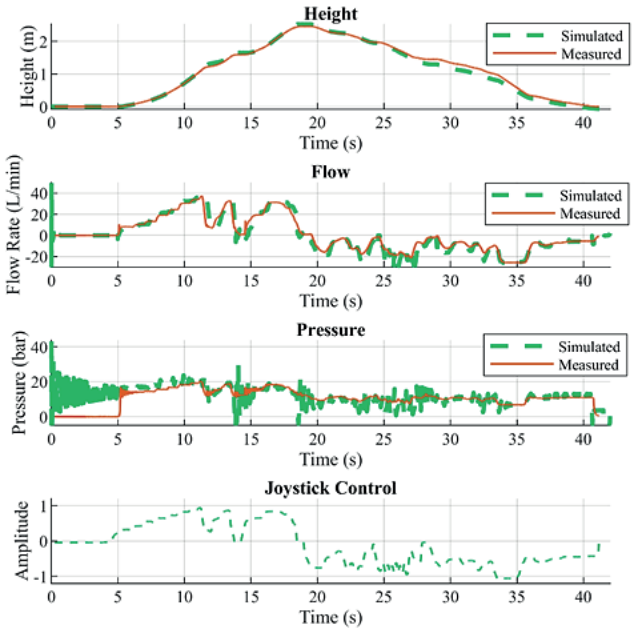


Figure 12: Measured and simulated results comparison for varying velocity scenario

The co-simulation model can hence be used as a tool to generate data for various failure modes by changing parameters [5]. The generated data will be used to train a classifier that can differentiate between the type of failure modes with the aid of data pre-processing as well as AI algorithms.

4 Conclusion

On-board intelligent predictive maintenance tool to avoid sudden breakdown of components would increase the system reliability and reduce downtime. It is often challenging to anticipate failure modes and to fabricate faults in an actual working system to generate valuable data for preset conditions for novel condition monitoring AI-based methods. The presented work propose a tool based on co-simulation multi-physics model for creating a training data for novel condition monitoring AI-based methods. Electric reach truck is used as study case. The co-simulation model was created by utilizing three software: Mevea Modeler for real-time multi-body dynamics, AMESim for hydraulic components and MATLAB/Simulink for electric drive and its control. The model is validated by comparing the simulation and experimental data. The simulation results are perfectly aligned for most of the operating cycle and hence can be used as a suitable tool for generating training data for AI based condition monitoring.

5 Acknowledgements

This work was enabled by the financial support of Academy of Finland (project ESTV) and internal funding from the Department of Automation Technology and Mechanical Engineering, IHA group at Tampere University, Finland.

Abbreviations

AI	Artificial Intelligence
AC	Alternating Current
CAD	Computer-Aided Design
DC	Direct Current
EM	Electric motor
FMI	Functional Mock-up Interface
FPGA	Field-Programmable Gate Array
IM	Induction motor
NN	Neural Network
PLC	Programmable Logic Controller

SCADA	Supervisory Control and Data Acquisition
XML	Extensible Markup Language
PC	Primary Cylinder
SC	Secondary Cylinder

References

[1] T. Krogerus, H. Sairiala, M. Saarinen, and K. T. Koskinen, "*Fault classification based on self-organizing maps in water hydraulic forklift,*" Proc. Tenth Scand. Int. Conf. Fluid Power, SICFP'07, no. January 2007, 2007.

[2] R. E. Link, A. C. H. Tan, P. S. K. Chua, and G. H. Lim, "*Condition Monitoring of a Water Hydraulic Cylinder by Vibration Analysis,*" J. Test. Eval. - J *TEST EVAL*, vol. 28, 2000, doi: 10.1520/JTE12142J.

[3] M. Ramachandran and Z. Siddique, "*A Data-Driven, Statistical Feature Based, Neural Network Method for Rotary Seal Prognostics,*" J. Nondestruct. Eval. Diagnostics Progn. Eng. Syst., vol. 2, p. 1, 2019, doi: 10.1115/1.4043191.

[4] M. Ramachandran, J. Keegan, and Z. Siddique, "*A hybrid pso-svm based method for degradation process prediction of reciprocating seal,*" Proc. Annu. Conf. Progn. Heal. Manag. Soc. PH*M*, vol. 11, no. 1, pp. 1–9, 2019, doi: 10.36001/phmconf.2019.v11i1.852.

[5] A. Abdul Azeez, X. Han, V. Zakharov, and T. Minav, "*AI-based condition monitoring of hydraulic valves in zonal hydraulics using simulated electric motor signals*," ASME/BATH 2021 Symposium on Fluid Power and Motion Control. Virtual, Online. October 19–21, 2021. V001T01A016. ASME. doi: 10.1115/FPMC2021-68615.

[6] A. Alenany, A. M. Helmi, and B. M. Nasef, "*Comprehensive Analysis for Sensor-based Hydraulic System Condition Monitoring,*" Int. J. Adv. Comput. Sci. Appl., vol. 12, no. 6, pp. 133–140, 2021, doi: 10.14569/IJACSA.2021.0120615.

[7] C. Dyer, "*Condition monitoring in hydraulic systems*," 1986.

[8] P. Garimella and B. Yao, "*Model based fault detection of an electro-hydraulic cylinder,*" Proc. Am. Control Conf., vol. 1, no. July, pp. 484–489, 2005, doi: 10.1109/acc.2005.1469982.

[9] B. Xu, J. Shen, S. Liu, Q. Su, and J. Zhang, "*Research and Development of Electro-hydraulic Control Valves Oriented to Industry 4.0: A Review,*" Chinese J. Mech. Eng. (English Ed., vol. 33, no. 1, 2020, doi: 10.1186/s10033-020-00446-2.

[10] J. R. Branco Ramos Filho and V. J. De Negri, "*Model-Based Fault Detection for Hydraulic Servoproportional Valves,*" Proc. from 13th Scand. Int. Conf. Fluid Power, June 3-5, 2013, Linköping, Sweden, vol. 92, no. April 2016, pp. 389–398, 2013, doi: 10.3384/ecp1392a38.

[11] A. Steinboeck, W. Kemmetmüller, C. Lassl, and A. Kugi, "*Model-based condition monitoring of an electro-hydraulic valve,*" J. Dyn. Syst. Meas. Control. Trans. ASME, vol. 135, no. 6, pp. 1–9, 2013, doi: 10.1115/1.4024800.

[12] K. Kudelina, B. Asad, T. Vaimann, A. Rassõlkin, A. Kallaste, and H. Van Khang, "*Methods of condition monitoring and fault detection for electrical machines,*" Energies, vol. 14, no. 22, pp. 1–20, 2021, doi: 10.3390/en14227459.

[13] L. Zhou, X. H. Jiang, and Y. Liu, "*Fault Modeling for Electro-Hydraulic Servo System Based on Hybrid System,*" Adv. Mater. Res., vol. 472–475, pp. 169–173, Feb. 2012, doi: 10.4028/www.scientific.net/AMR.472-475.169.

[14] S. Liu and G. Zhang, "*The fault simulation methods of hydraulic system based on AMESim,*" no. Icmse, pp. 1139–1143, 2015, doi: 10.2991/icmse-15.2015.207.

[15] W. Sun, D. Jian, Y. Yuan, and Y. Yuan, "*Fault simulation of electro-hydraulic servo system for fault self-healing based on immune principle,*" *Proc. - 2016 9th Int. Symp. Comput. Intell. Des. Isc. 2016*, vol. 2, pp. 136–139, 2016, doi: 10.1109/ISCID.2016.2040.

[16] L. I. Hatledal, H. Zhang, A. Styve, and G. Hovland, "*FMU-proxy: A Framework for Distributed Access to Functional Mock-up Units,*" Proc. 13th Int. Model. Conf. Regensburg, Ger. March 4–6, 2019, vol. 157, pp. 79–86, 2019, doi: 10.3384/ecp1915779.

[17] J. Pyrhonen, T. Jokinen, and V. Hrabovcova, "*Design of Rotating Electrical Machines"*, WILEY, 2013.