Interoperability Challenges and Opportunities in Vehicle-in-the-loop Testings: Insights from NUVE Lab's Hybrid Setup ⋆

Sarthak Acharya * **Aparajita Tripathy** ** **Juho Alatalo** *** **Pekka Seppänen** ** **Aki Lamponen** ** **Jukka Säkkinen** ** **Tero Päivärinta** *

* *M3S Research Unit, ITEE, University of Oulu, Finland (e-mail: firstname.lastname@oulu.fi).*
** *Oulu University of Applied Science (OAMK), Oulu, Finland (e-mail: firstname.lastname@oamk.fi)*
*** *IMS Research Unit, FTech, University of Oulu, Finland, (e-mail: firstname.lastname@oulu.fi)*

Abstract: Research and innovation in Vehicle-in-the-loop (VIL) testing is garnering more attention than ever. Integrating cyber-physical systems (CPS) into the VIL setups further enhances their functionality and hybridises the testing. Setting up any VIL infrastructure involves substantial investments and thus requires critical analysis of the resources to achieve the intended results. This study focuses on such a VIL testing infrastructure development at NUVE-Lab, aiming to provide state-of-the-art facilities for hybrid automotive testing. The facility includes physical components such as a heavy tractor (Valtra), dynamometers, an Actuators power need generation system (APGS) system, and battery emulators (BE), complemented by digital twins (DTs) of each physical machine, process, and environment to automate the testing facilities. This research examines various interoperability challenges within the current VIL framework. Three distinct testing scenarios are created to assess the overall functionalities of the hybrid setup: dynamometer-in-the-loop, APGS-in-the-loop, and BE-in-the-loop. Analyzing individual cases highlighted the need for different modeling and simulation (M/S) tools to develop digital twins. Among the tools, SIMULINK is used to build and refine the models of DTs, whereas MATLAB is used to develop control algorithms. The study also explores the adoption of Functional Mock-up Interface (FMI) standards to facilitate seamless interoperability among modeling and simulation tools. Additionally, the potential integration of the Eclipse Arrowhead framework (EAF), an IoT-edge-based automation tool, is discussed to enhance efficient data management, service interoperability, and the integration of various cyber-physical system components. In conclusion, this paper outlines the interconnection of the digital and physical platforms to evolve a hybrid VIL test laboratory, envisioning the future trajectory of the NUVE-Lab.

Keywords: Digital Twin (DT), Interoperability, Vehicle-in-the-loop (VIL), Eclipse Arrowhead Framework (EAF), MATLAB, SIMULINK.

1. INTRODUCTION

Vehicle-in-the-loop (VIL) is an automotive testing method that combines real machines with virtual simulations rather than conducting purely virtual tests. Advanced levels of testing in the design phase with reasonable costs and enhanced safety features have accelerated special research attention to this field in recent years Cheng et al. (2024). The focus of the traditional VIL setups was to incorporate different hardware and software in the loop for testing facilities. However, the inclusion of industry 4.0/5.0 complaint technologies has introduced hybrid VIL test setups. State-of-the-art testings and validations with hybrid setups are made possible by integrating cyber systems with physical processes, digital twins (DTs), communication (5G/6G), internet of things (IoTs), augmented reality (AR) and Virtual reality (VR) technologies, artificial intelligence (AI), machine learning (ML) algorithms, cloud and edge computing, and others Zhao et al. (2023). However, the addition of diverse technologies increases complexity and interoperability concerns for the interconnected systems Lv et al. (2024). One such cyber-physical-system-based hybrid VIL research infrastructure at the NUVE lab in Oulu, Finland, is discussed in this paper.

Cyber-Physical Systems (CPS) combine computational and engineering fields to develop systems that connect the

⋆ The financial support received from the European Union NextGenerationEU instrument, which is funded by the Research Council of Finland under grant number 352726.

physical and digital realms. These systems incorporate intricate networks, control systems, information fusion, and optimization techniques Su et al. (2022). These systems have the potential to revolutionize various domains by enabling advanced functionalities and automation. However, the implementation of CPS is accompanied by several challenges. One significant challenge is the diverse nature of technologies and standards, leading to interoperability issues that hinder seamless communication and integration among different subsystems within CPS Chaudhry et al. (2019). Among others, digital twin technologies (DTTs) have been observed as a core component of CPS for creating the virtual representation of physical objects or systems. The diverse nature of digital twin implementations, with various data structures and interfaces, hinders the interoperability between different digital twins Wang et al. (2023). Additionally, enabling real-time dynamic interactions between the simulation world and the physical world through digital twins introduces complexities in ensuring consistent and reliable data exchange, posing a challenge to achieving seamless interoperability. Therefore, IoT-based automation tools are effective for interoperability in such hybrid setups. Eclipse Arrowhead framework (EAF) is one such potential tool for interoperability solutions and is considered for NUVE's VIL setup.

The creation and optimization of digital twins (DTs) in VIL setups are dependent on various modeling and simulation (M/S) tools. The test-bench-based hybrid testing methods require the simulation of the environment in order to provide realistic inputs for the device under test. These simulation inputs need to be generated in real-time, which narrows down the choice of simulation tools and excludes all the software that needs heavy computing (such as FEM tools). To keep the number of simulation software reasonable, we are going to use Mevea and MATLAB/SIMULINK as the main tools in this research project. Mevea is a software for real-time simulation of mechanics, hydraulics, power transmission, and the operating environment (Mevea (2024))). MATLAB/SIMULINK is a software for modeling and simulating dynamic systems in multiple domains (Mathworks (2024b)).

In this paper, we aim to study the hybrid VIL setup in NUVE-Lab at the Oulu Applied Science University(OAMK), Oulu, Finland. The laboratory setup consists of a Valtra tractor, 4 dynamometers, an APGS system, and a battery emulator as physical components. On the counter, the digital models of the tractor are designed in the Mevea environment for simulation and testing Oulu Univesity of Applied Science (OAMK) (2023). A few digital counterparts are modeled using SIMULINK as well. One of the primary objectives is to study the interoperability challenges across various test scenarios. Next, to analyze the usability of existing Internet of Things (IoT) middleware and Open-source platforms as potential candidates to overcome the hurdles. Thus, two important research questions (RQs) are formulated:

RQ-1: *What are the essential components to be considered in setting up a hybrid VIL setup?*

RQ-2: *What are the key interoperability challenges in a hybrid VIL environment?*

The outline of the paper is as follows: background and key enablers for the hybrid VIL are mentioned in section 2. The research process and context followed are presented in section 3. Section 4 illustrates the key findings of the research work. Then, the results are discussed in section 5. Lastly, the conclusion and future scopes of the work are presented in section 6.

2. BACKGROUND

In this section, a summary of the recent works related to the vehicle-in-the-loop (VIL) testing is presented.

2.1 VIL Testbench Setup: SOTA and Concepts

The vehicle-in-the-loop technique is the combination of real-world vehicles with virtual environments to perform experiments in a cost-efficient way Park et al. (2020). The experimental setups for the VIL testing vary with infrastructures based on specific research objectives, testing precision, accuracy of experiments, available technologies, and resources. However, there is a core set of basic requirements to establish a functional VIL testing environment, including physical vehicles (hardware), simulation software, data acquisition systems, computing infrastructure, and safety measures Cheng et al. (2024). To validate the results, there are 3 basic testing approaches: simulation, hybrid (closed-field), and real (on-road) testing Solmaz et al. (2020). Simulation tests are the most economical and low in accuracy as they only involve digital twins and software. On the other hand, real tests include the physical vehicles in real-world scenarios, which are highly expensive but most accurate. Therefore, hybrid tests are often considered as optimal solution as they take both hardware and software in the loop to perform experiments.

2.2 Challenges in VIL setup

VIL testing setups have proven their worth for automotive research and development, but they face several challenges. One of the complexities in setting up a VIL laboratory involves a huge investment in infrastructure. Integration of various hardware and software platforms, maintenance and updates of the technologies, and scalability are among the top challenges. However, in this study, we only investigate and focus on the technical challenges associated with the implementation of VIL. The list of challenges is as follows:

- System Integration and Compatibility
- Connectivity and V2X Communication
- Real-time processing
- Data Acquisition and Handling
- Scalability and Flexibility
- High-Fidelity Sensor Simulations
- Mixed Reality and Enhanced Immersion
- Standardization and Bench-marking

Integration of advanced technologies such as radar target simulation and environment perception simulation in VIL testing poses technical challenges Maier et al. (2018). The integration phase is particularly complex due to the numerous components in modern vehicles Rossi et al. (2017). Additionally, the growing complexity of automotive cyber-physical systems and the verification challenges posed by

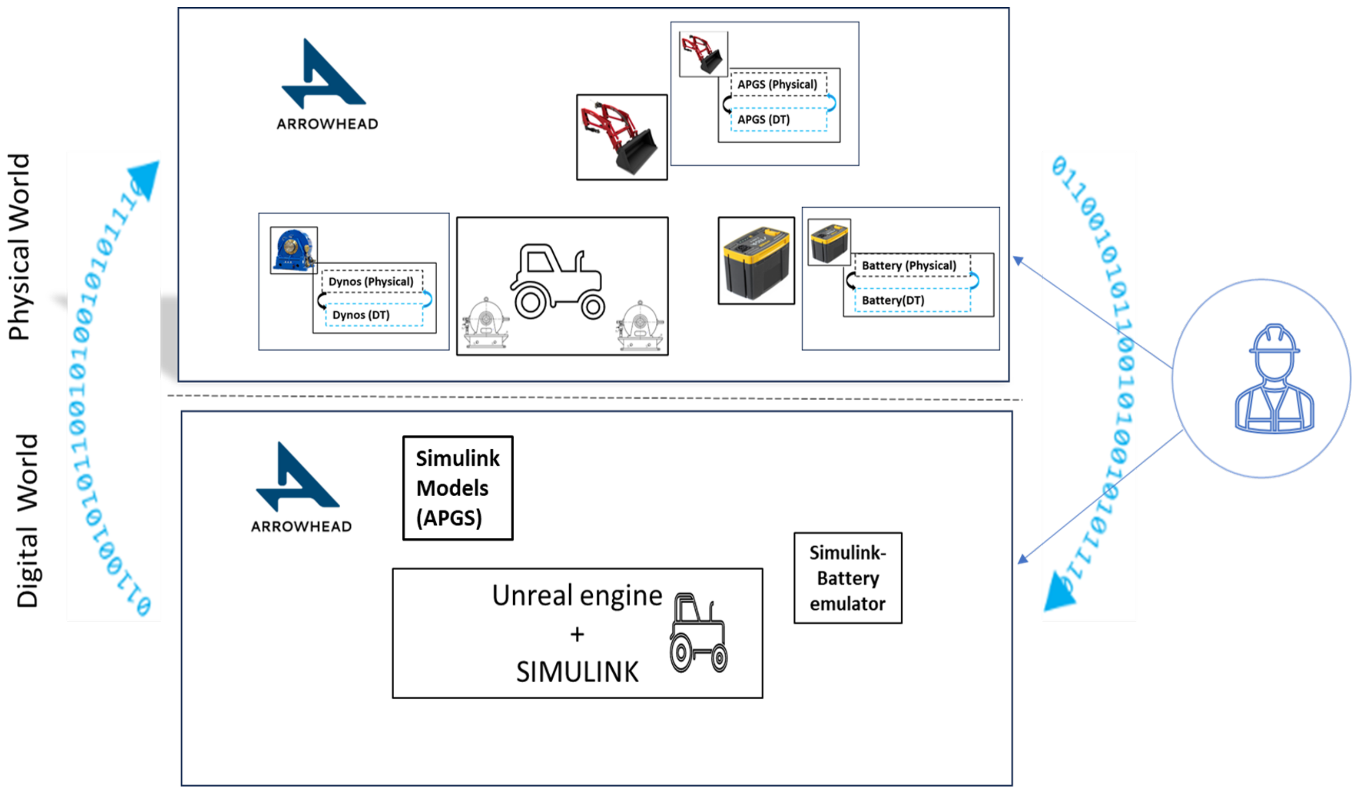

Fig. 1. Overview of the Vehicle-in-the-loop Testing in NUVE-Lab.

distributed software in vehicles create difficulties Raghupatruni et al. (2019). There is also a pressing need for simulations that accurately reflect the complexity of real-world testing environments Babić et al. (2020). Moreover, thorough testing for autonomous vehicles remains a major challenge Chen et al. (2020).

2.3 Role of Simulation and Modelling Platforms

Modeling, in general, requires a lot of modeling expertise and is labor intensive, although in this project, we are modifying the existing Valtra tractor model introduced by Jaiswal et al. (2019). The real-time demand of the simulation models also restricts the choices made in modeling and reduces the details that can be applied to the model.

In this research, a setup will be made where the hydraulic system of the tractor is modeled in Simscape, and the front loader is modeled in Mevea. The simulation is performed as a co-simulation where the Simscape model is exported as a functional mock-up unit (FMU)to the Mevea environment (FMI (2024)). The use of the FMU in co-simulation requires a lot of adaptation of the models and can emerge with various issues due to the varying modeling principles in different software.

2.4 Role of CPS, DTs and IoT Platforms

IoT platforms and frameworks play a significant role in enabling Vehicle-in-the-Loop (VIL) testing by offering the necessary infrastructure for connecting and monitoring all the components used in the testing procedure. These platforms allow the seamless integration of IoT devices, sensors, and communication technologies within the VIL setup, enabling the exchange of real-time data and control functions. There are numerous IoT frameworks and platforms currently being used by industries, including the Eclipse Arrowhead Framework, AUTOSAR, BaSys, FIWARE, OCF, IoTivity, and more (Paniagua and Delsing (2020)). The Eclipse Arrowhead framework has been selected for the GORI project to develop networked connectivity between different VIL setups using local cloud automation.

2.5 Industry Complaint Open-source IoT Framework

The Eclipse Arrowhead Framework (EAF) is an open-source industrial IoT framework that provides interoperability solutions in Industry 4.0.Delsing (2017). This framework is built on the principles of SOA (Service Oriented Architecture) and leverages the concept of local clouds. Here, a local cloud is a network of interconnected systems and services that function within a limited environment, usually within a particular organization, stakeholder, or region. This architecture emphasizes Standardized communication, late binding, loose coupling, cyber-security, scalability, Dynamic service Discovery, and multi-stakeholder integration. The framework enables real-time communication between different systems irrespective of the technology being used within a local cloud or between systems registered in different local clouds. The framework is compatible with several communication protocols, including HTTP, COAP, MQTT, and OPC UA, as well as transport protocols such as TCP, UDP, and DTLS/TLS,Acharya et al. (2023). Three mandatory core systems are provided to facilitate interaction between systems. The mandatory core systems are:

(1) the *Service Registry* system, which records the services currently being offered,
(2) the *Authorization* system that controls system-to-system authorization at a detailed level for secure service exchange,
(3) the *Orchestrator* system that enables the consumer application to discover the required service endpoint at run time.

In addition to the core systems, there are client systems that essentially function as application systems, either as providers or consumers, that seek to establish communication with one another. Each core system and provider system offers a set of services that are registered with the *Service-Registry* and includes a specific set of interfaces, metadata, and service paths. Any system that wishes to utilize a service must request the service address from the *Orchestrator* system during runtime. The Orchestrator verifies with the Service Registry if the service is now accessible and then consults the Authorization system to determine if the exchange of the specific service is approved between the two systems. After the verification process is completed successfully, the *Orchestrator* provides the relevant service endpoint in response. The consumer system directly contacts the service-provider system. The framework also employs the Gatekeeper and Gateway core systems to provide inter-cloud communication.

2.6 NUVE-Lab Vision

In NUVE-Lab, the ongoing research is focused on building a hybrid VIL testing platform that can accommodate cutting-edge automotive research and development. The current laboratory setup is shown in Fig. 1. Further optimization of digital twins by accurately replicating the physical entities will enhance applications such as predictive maintenance, real-time monitoring, fault diagnosis, etc. Different test scenarios will be developed further by integrating sensors, actuators, programmable logic controllers (PLCs), and Lidars. The longer-term goal is to enable testing across a wide range of vehicles, both with physical and digital prototypes. This will leverage facilities to test all types of prototypes before the realization of products.

3. RESEARCH PROCESS AND CONTEXT

This section explains the research process followed to analyze the integration of physical and software components of the current VIL setup at the NUVE lab. To carry out a systematical investigation of the interconnectivity and interoperability concerns, three test scenarios are developed (as shown in Fig. 2. In the first scenario (**T1**), dynamometer is attached to the vehicle to perform tests, testing setup is shown in Fig. 2. The second case (**T2**) is for testing hydraulic systems by attaching an actuator's Power Need Generation System (APGS) to the vehicle. In the third setup (**T3**), a battery emulator is connected to the vehicle. All the test scenarios are explained in section 4.

3.1 Challenge Identification

To investigate the interoperability challenges, all the connections are labeled (A, B, C, ..., J) in each test scenario.

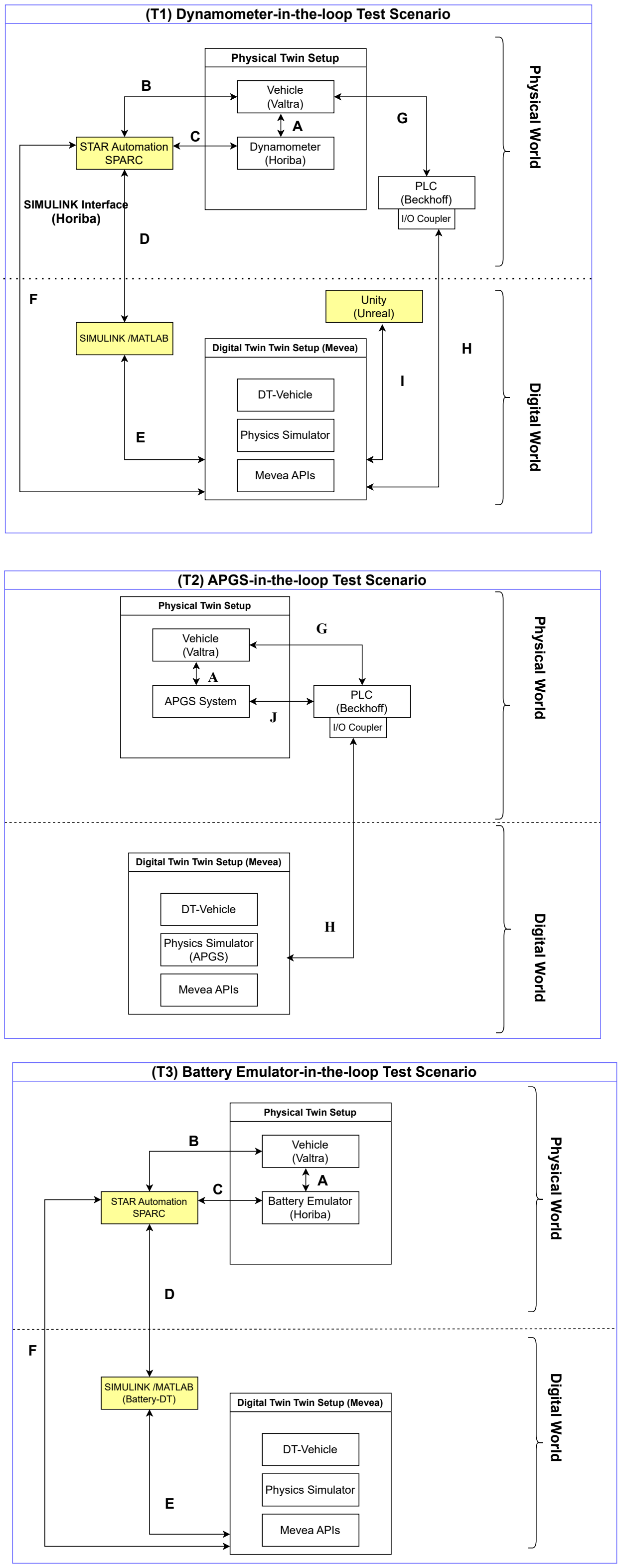

Fig. 2. Test Scenarios in NUVE's VIL Setup: (T1) Dynamometer Testing with the Vehicle; (T2) Hydraulics System testing with the Vehicle; (T3) Battery Emulator testing with the Vehicle.

Table 1. Challenges Identified for all the 3 Test Scenarios

Connection	Type	Identified Challenge(s) Dynos-in-the-loop (T1)	APGS-in-the-loop (T2)	BE-in-the-loop (T3)
A	P-P	- Mechanical setup and configurations - Version Maintenance and Upgradations - Software Integration	- Electrical setups - Control system integration - Version Maintenance	- Electrical Compatibility - Interface Management
B	P-D	- Protocol Mismatch - API Compatibility -Maintenance and Upgradations	Not Available	- Protocol Mismatch - API Compatibility -Maintenance and Upgradations
C	P-D	- Communication Protocols - Latency	Not Available	- Protocol consistency - Control system-software integration - Latency
D	D-D	- Data Exchange - Protocol Mismatch - Version compatibility - Latency (better than connection 'F')	Not Available	- Data Exchange - Protocol Mismatch - Version compatibility - Latency
E	D-D	- API compatibility - Model Accuracy - Version compatibility - Data Format and representation	Not Available	- Creation of DTs - API compatibility - Model Accuracy - Version compatibility - Data Format and representation
F	D-D	- Protocol Compatibility - API compatibility - Latency (UDP loop) - Model Accuracy	Not Available	- Protocol Compatibility - API compatibility - Data format Compatibility - Latency - Model Accuracy
G	P-P	- Protocol Mismatches (CAN to Mod-bus/ Ethernet/IP) - Latency - I/O Compatibility -Integration with other systems	- Protocol Mismatches (CAN to Mod-bus/ Ethernet/IP) - Latency - I/O Compatibility -Integration with other systems	Not Available
H	P-D	- Data Exchange and Latency - Digital Twin Accuracy - Scalability	- Creation of Digital Twins (DTs) - Data communication - Integration with other systems - Refinement of DTs	Not Available
I	D-D	- Model Accuracy - Data Format Compatibility - Integration of Physics Simulators	Not Available	Not Available
J	P-P	Not Available	- Protocol consistency - Control system-software integration - Latency	Not Available

Three main categories of the connections are shown in Figure 2: physical to physical (P-P), physical to digital (P-D), and digital to digital (D-D). Identified challenges are mentioned in section 4.

4. RESULTS

4.1 Dynamometer-in-the-Loop Testing (T1)

In this test setup, the dynamometers are tested with the vehicle, as shown in Fig. 2. All the connections and iden-tified challenges are mentioned in table 1. Dynamometers connected with the vehicle (Valtra tractor) are controlled by STARS automation software and exchange data with the digital twin setup. Digital twins are created in the Mevea tool and updated in real-time using SIMULINK models. Some of the significant interoperability challenges faced in this VIL testing are protocol compatibility, la-tency, API compatibility, and software integration.

4.2 APGS-in-the-loop Testing (T2)

In the second test setup, an actuator power need generation (APGS) system was tested in the VIL. The test setup is shown in Fig. 2, and the identified challenges are mentioned in table 1. This test setup is optimized in terms of the number of connections (compared to T1). One of the crucial parts is to create digital twins with granular information from their physical counterparts. Other challenges include protocol mismatching, input-output (I/O) compatibility, latency and etc.

4.3 Battery Emulator-in-the-Loop Testing (T3)

The third setup is to test the battery in the hybrid VIL setup. The experimental setup is presented in Fig. 2. Most of the connections in this setup have familiarity with the first test-case (T1). The identified challenges are listed in table 1. Some essential interoperability barriers in this testing are achieving real-time data and feeding into the twin of the battery. Software integration with the control systems is very crucial for this test setup and can possibly need middleware solutions in the future.

5. DISCUSSIONS

5.1 Dynamometer-in-the-Loop Testing (T1)

In the dynamometer-in-the-loop test case, the Valta tractor mounted on dynamometers interacts with Mevea software, where the multi-body dynamics model of the tractor and the virtual environment are modeled (as shown in Fig. 2). The purpose is to evaluate the performance and the behavior of the tractor under realistic driving conditions without the need for a physical test track. For example, in one case, the Mevea software (DT setup) receives the wheel

Table 2. Analysis of each connection and Possible Arrowhead Solutions

Connection	Type (Test Case)	EAF Mapping	Feasibility
A, J	P-P (T1, T2, T3)	Creating individual Arrowhead application systems for vehicles and dynamometers that can connect to sensors and actuators of the hardware. These systems can then provide services that collect, store, and manage these sensors and actuator's data.	The sensors and actuators' information from the hardware can be easily extracted at the next connection levels. Hence, using Arrowhead at this point is not recommended.
B	P-D	Creating individual Arrowhead application systems for the vehicle and STAR automation that can exchange information with each other and control the actuators in the vehicle through Arrowhead services.	Using Arrowhead at this point is not recommended to avoid latency.
C	P-D	Creating individual Arrowhead application systems for the dynamometers and STAR automation that can exchange information with each other and control the actuators in the vehicle through Arrowhead services.	This will create unnecessary latency. Hence, using Arrowhead at this point is not recommended.
D	D-D	Creating individual Arrowhead application systems for STAR Automation and SIMULINK/MATLAB, where information exchange between the two systems takes place via the Arrowhead service exchange process.	Arrowhead can be useful if it can avoid additional latency.
E	D-D	Creating individual Arrowhead application systems for SIMULINK/MATLAB and Mevea. The SIMULINK system can send the input(torque) values to the Mevea through the Arrowhead service exchange process. After testing the DT against the input value at Mevea, the Mevea system can send the outputs (speed of the tires) to the SIMULINK system.	Arrowhead can be used in this scenario as it can be useful to store the test data in a standardized way (senML) into the Data Manager (DM), provided it avoids latency.
F	D-D	Creating individual Arrowhead application systems for STAR automation and Mevea that can exchange information like torque and speed securely via the Arrowhead service exchange process.	Arrowhead can be used to establish direct communication between the two entities if there are no SIMULINK models in the loop for testing.
G	P-P	Creating individual Arrowhead application system for the vehicle and PLC that can connect to sensors and actuators of the Vehicle to the PLC I/O ports. These systems can then provide services that collect, store, and manage these sensors and actuators' data.	The sensor and actuator information from the vehicle can be easily extracted directly from the PLC. Hence, using Arrowhead at this point is not recommended.
H	P-D	Creating individual Arrowhead application systems for the PLC and Mevea. The PLC system can provide services to read the input and output signals from the PLC and send that information to the Mevea system for testing. The Mevea system can send the output values back to the PLC via the Arrowhead service exchange process.	Arrowhead can be utilized here in order to exchange data in a secure way and also to store the data in the Data Manager (DM). Changes to the hardware (like sensors and actuators) can be handled at run time through the automatic discovery of new services, resulting in reduced engineering effort.
I	D-D	Creating individual Arrowhead application systems for Unity and Mevea and exchange information between the two systems using Arrowhead service exchange process and TCP protocol.	Arrowhead can be utilized. Latency needs to be tested. (Best suited when Mevea and Unity operate in different networks or locations.)

speed data from the dynamometers and uses it to calculate the motion of the tractor's simulation model in the virtual environment. The Mevea software also calculates the driving resistance that the tractor model encounters in the virtual environment based on the position and velocity of the simulation model and the characteristics of the environment. Therefore, interoperability of data between the physical and digital parts is a must.

The communication between Mevea software and the Horiba system (STARS automation) has to be routed through Matlab/SIMULINK because, at the moment, Horiba and Mevea have no ability to communicate directly with each other. This might cause extra latency, but the current setup appears to be a working solution based on the preliminary communication tests done in NUVE-Lab.

5.2 APGS-in-the-loop Testing (T2)

In the APGS-in-the-loop testing scene, the hydraulic system of the Valtra tractor is connected to the APGS system, which can control the hydraulic load of the main hydraulic line and the pressure of the load-sensing line. The performance of the tractor's hydraulic system can be evaluated without the need for any auxiliary equipment, such as a front loader. Alternatively, we can also test how the tractor reacts to the auxiliary equipment without actually mounting the equipment to the tractor. The Mevea software is used to build a multi-body dynamics model of the front loader and the model of the hydraulic system of the Valtra tractor. The modeling can also be done by using Simscape (Mathworks (2024a)), which is an extension for the SIMULINK. The simulation model of the front loader experiences different kinds of loads generated in the virtual environment, such as lifting. The front loader movement is controlled by giving control commands in the virtual environment.

The simulation model provides the pressure and flow information of the hydraulic fluid to the APGS system, which in turn generates similar pressure and flow in the real world in order to load the tractor's hydraulic system accordingly. The speed of the tractor's hydraulic pump can be measured and sent to the simulation model in order to create a feedback loop. This way, a realistic system that mimics the behavior of the tractor and the front loader in different operating conditions can be created.

The FMU adaptation requires resolving several compatibility issues, such as data types, input/output variables, initialization methods, and solver settings. The collaboration with the software supplier is working exemplary, and the raised questions have been resolved without delays, although the fully working test model is still under construction.

The APGS system communicates with Mevea through Bechhoff/EtherCAT. This has been tested, and the tentative results show that the solution works. This raises the question of whether it would be possible for Mevea to communicate with the Horiba system via Ether-CAT. This would eliminate the need for the SIMULINK between Mevea and Horiba and would possibly increase the communication speed.

5.3 Battery Emulator-in-the-Loop Testing (T3)

In this testing case, the battery emulator is yet to be fully integrated with the hybrid VIL setup. To ensure seamless communication between the physical setup and digital twin setup, compatibility across protocols and data consistency is required. Another important aspect at this moment is to create an accurate digital twin that can simulate unforeseen conditions. To work on the mentioned interoperability challenges, the adaptation of IoT middleware solutions to the current VIL setup is needed. One such framework is analyzed below.

5.4 Logical Reflections on AH Potential to NUVE-VIL context

Utilizing the industry 4.0 IoT frameworks in VIL testing offers several benefits. The benefits encompass ensuring data quality management, efficient data processing, and enhanced data security practices Javed et al. (2019)Mishra et al. (2015). The EAF follows the principles of Industry 4.0, offering the same advantages outlined earlier. The basic steps in utilizing EAF in any connection are to create an Arrowhead application system for each end of the connection and then establish communication through the service exchange process of EAF explained earlier in section 2.5. In order to gain a deeper understanding of how EAF might be employed, Table 2 presents a comprehensive list of the connections between the various components and systems in NUVE Lab, as well as how EAF can optimize the data exchange process.

One of the primary benefits of Arrowhead is its ability to automatically discover new services during execution. This feature is particularly valuable in the continuously evolving manufacturing landscape, characterized by frequent hardware and software changes. Automatic service discovery techniques minimize the effort needed to synchronize changes in the physical world with the digital realm. For instance, changes in the physical world, like switching between sensors or PLCs, can be easily handled during execution Tripathy et al. (2022). This will significantly reduce the effort involved in re-engineering.

5.5 Threads to Validity

The hybrid VIL setup discussed in the paper is an ongoing research work at the NUVE lab at OAMK, Oulu.

Internal validity This study is based on the practical work at the laboratory. Therefore, a possible threat to the study findings includes missing the relevant data. We employed a thorough literature survey on the existing literature to analyze our findings. In *external Validity*, concerns the generalizability of the findings and diversification of technologies across different setups. To mitigate this all the available industrial reports and gray literature were refereed to validate the findings.

6. CONCLUSION AND FUTURE WORK

In this paper, a hybrid VIL setup of the NUVE lab is demonstrated. This study is focused on building a hybrid VIL environment to assess advanced automotive testing capabilities. Interoperability is observed as a key factor in building such setups while integrating CPS, DTs, software components, and other industry-compliant technologies. The study highlighted the requirements and challenges for transitioning from traditional VIL to a hybrid testing platform. The physical infrastructure is integrated with cyber-physical- systems to facilitate the advanced-level testing. Three test scenarios, including dynamometer-in-the-loop, APGS-in-the-loop, and Battery emulator-in-the-loop, have been designed to investigate and explore the interoperability challenges. The role of modeling and simulation (M/S) platforms in developing and refining digital twins is reviewed. In addition, the potential of the Arrowhead framework to facilitate interoperability in a hybrid VIL setup is examined. These insights from the NUVE lab also lay a foundation for creating more innovative hybrid VIL setups in the future.

This is an ongoing research project, and future work will focus on finding engineering solutions to existing interoperability concerns. The Arrowhead framework will further be tested to enhance interoperability among digital twins, data management, scalability, and collaboration. Furthermore, the creation and refinement of digital twins to achieve better accuracy in mimicking physical components and processes will be focused. The ultimate vision of this hybrid VIL laboratory is to facilitate state-of-the-art automotive testing and innovation.

ACKNOWLEDGEMENTS

The research work is carried out under the GO-RI project sponsored by the Academy of Finland. This is a collaborative work between the University of Oulu and Oulu Applied Science University (OAMK), Finland. The authors would like to thank NUVE-Lab and the team for setting up the testing infrastructure. The authors would like to acknowledge the financial support received from the European Union – NextGenerationEU instrument which is funded by the Research Council of Finland under grant number 352726.

REFERENCES

Acharya, S., Wintercorn, O., Tripathy, A., Hanif, M., Van Deventer, J., and Päivärinta, T. (2023). Twins interoperability through service oriented architecture: A use-case of industry 4.0. In *Proceedings of the Annual Symposium of Computer Science 2023 co-located with The International Conference on Evaluation and Assessment in Software Engineering (EASE 2023)*. R. Piskac c/o Redaktion Sun SITE, Informatik V, RWTH Aachen.

Babić, A., Vasiljević, G., and Mišković, N. (2020). Vehicle-in-the-loop framework for testing long-term autonomy in a heterogeneous marine robot swarm. *IEEE Robotics and Automation Letters*, 5(3), 4439–4446.

Chaudhry, N., Yousaf, M.M., and Khan, M.T. (2019). Security assessment of data management systems for cyber physical system applications. *Journal of Software Evolution and Process*. doi:10.1002/smr.2241.

Chen, Y., Chen, S., Xiao, T., Zhang, S., Hou, Q., and Zheng, N. (2020). Mixed test environment-based vehicle-in-the-loop validation-a new testing approach for autonomous vehicles. In *2020 IEEE intelligent vehicles symposium (IV)*, 1283–1289. IEEE.

Cheng, J., Wang, Z., Zhao, X., Xu, Z., Ding, M., and Takeda, K. (2024). A survey on testbench-based vehicle-in-the-loop simulation testing for autonomous vehicles: Architecture, principle, and equipment. *Advanced Intelligent Systems*, 2300778.

Delsing, J. (2017). *IoT Automation: Arrowhead Framework*. CRC Press.

FMI (2024). Functional mock-up interface specification, version 3.0.1. `https://fmi-standard.org/docs/3.0.1/`. [Online; accessed 17-May-2024].

Jaiswal, S., Korkealaakso, P., Åman, R., Sopanen, J., and Mikkola, A. (2019). Deformable terrain model for the real-time multibody simulation of a tractor with a hydraulically driven front-loader. *IEEE Access*, 7, 172694–172708. doi:10.1109/ACCESS.2019.2956164.

Javed, A., Yousefnezhad, N., Robert, J., Heljanko, K., and Främling, K. (2019). Access time improvement framework for standardized iot gateways. In *2019 IEEE International Conference on Pervasive Computing and Communications Workshops (PerCom Workshops)*, 220–226. IEEE.

Lv, S., Qin, Y., Gan, W., Xu, Z., and Shi, L. (2024). A systematic literature review of vehicle-to-everything in communication, computation and service scenarios. *International Journal of General Systems*, 1–31.

Maier, F.M., Makkapati, V.P., and Horn, M. (2018). Environment perception simulation for radar stimulation in automated driving function testing. *Elektrotech. Informationstechnik*, 135(4-5), 309–315.

Mathworks (2024a). Simscape. `https://se.mathworks.com/products/simscape.html`. [Online; accessed 16-May-2024].

Mathworks (2024b). Simulink. `https://se.mathworks.com/products/simulink.html`. [Online; accessed 16-May-2024].

Mevea (2024). Mevea simulation software. `https://mevea.com/solutions/software/`. [Online; accessed 16-May-2024].

Mishra, N., Lin, C.C., and Chang, H.T. (2015). A cognitive adopted framework for iot big-data management and knowledge discovery prospective. *International Journal of Distributed Sensor Networks*, 11(10), 718390.

Oulu Univesity of Applied Science (OAMK) (2023). Nuve-lab. `https://oamk.fi/nuve-lab/`. [Online; accessed 04-April-2024].

Paniagua, C. and Delsing, J. (2020). Industrial frameworks for internet of things: A survey. *IEEE Systems Journal*, 15(1), 1149–1159.

Park, C., Chung, S., and Lee, H. (2020). Vehicle-in-the-loop in global coordinates for advanced driver assistance system. *Applied Sciences*, 10(8), 2645.

Raghupatruni, I., Goeppel, T., Atak, M., Bou, J., and Huber, T. (2019). Empirical testing of automotive cyber-physical systems with credible software-in-the-loop environments. In *2019 IEEE International Conference on Connected Vehicles and Expo (ICCVE)*, 1–6. IEEE.

Rossi, R., Galko, C., Narasimman, H., and Savatier, X. (2017). Vehicle hardware-in-the-loop system for adas virtual testing. *Towar. a Common Software/Hardware Methodol. Futur. Adv. Driv. Assist. Syst. DESERVE Approach*, 251–267.

Solmaz, S., Rudigier, M., and Mischinger, M. (2020). A vehicle-in-the-loop methodology for evaluating automated driving functions in virtual traffic. In *2020 IEEE Intelligent Vehicles Symposium (IV)*, 1465–1471. IEEE.

Su, H., Xiao, B., Zhou, M., Qi, W., Sandoval, J., and Kim, S.T. (2022). Theory, applications, and challenges of cyber-physical systems 2021. *Complexity*. doi: 10.1155/2022/9861298.

Tripathy, A., van Deventer, J., Paniagua, C., and Delsing, J. (2022). Opc ua service discovery and binding in a service-oriented architecture. In *IEEE 5th International Conference on Industrial Cyber-Physical Systems*.

Wang, X., Hu, X., Ren, Z., and Tian, T. (2023). Knowledge-graph based multi-domain model integration method for digital-twin workshops. doi:10.21203/rs.3.rs-2630784/v1.

Zhao, X., Gao, Y., Jin, S., Xu, Z., Liu, Z., Fan, W., and Liu, P. (2023). Development of a cyber-physical-system perspective based simulation platform for optimizing connected automated vehicles dedicated lanes. *Expert Systems with Applications*, 213, 118972.