Seamlessly Interfacing Automation Systems with Simulations: A Case Study for FMI and OPC UA

Master Thesis Proposal

Student: Andrej Kurtović - 01428987
Main advisor: Ao.Univ.Prof. Dipl.-Ing. Dr.techn. Wolfgang Kastner
Co-advisor: Univ.Ass. Dipl.-Ing. Dipl.-Ing. Gernot Steindl, BSc
Date: 26.03.2021.

1. Motivation and Problem Statement

In the industry, and especially as a part of Industry 4.0 applications, simulations are an important part of the overall cyber-physical system (CPS) [WMO+16]. Simulations can be utilized during CPS operation, for example as a digital twin, to monitor, diagnose, predict, control, and reconfigure parts of the system [SSK+20]. On the other hand, simulations can be utilized during the CPS development phase, to test and verify the system's functionality. The testing and the evaluation can be carried out in real time, which ensures that the embedded control system is able to give the control input within the sampled period. Another option in CPS development is using real actuators instead of simulations, if the actuators are difficult to model [Bac05]. Utilizing hardware-in-the-loop presents an advantage, even during development, as some (finished) components can be tested, while the components which are still in development are replaced by their virtual counterparts. In order to obtain these benefits, the control logic of the simulation environments is often programmed completely independently from the actual working system, in a simulation environment. This leads to compatibility problems if the simulation has to interact with the real part of the system. Additionally, the process of converting the programs from the simulation environment to the deployment environment often introduces errors, and prolongs time-to-deployment of the system. Seamless interfacing attempts to solve these compatibility issues.

Automation systems, and cyber-physical systems in particular are often heterogeneous in nature. This presents both advantages and disadvantages. On one hand, having a sub-system specifically tailored to the given problem usually presents a considerable performance advantage. On the other hand, a heterogeneous system brings many challenges and obstacles with regards to system interoperability [JOM16]. One possibility for overcoming the interoperability issues is through a middleware system in combination with information modelling. An example for this would be OPC Unified Architecture (OPC UA).

OPC UA, as defined in the IEC 62541 standard [IEC62541], is often mentioned in the context of Industry 4.0 applications [PSM+17], as it offers a client-server machine-to-machine communication protocol with a Service Oriented Architecture (SOA), and it also provides an extensible way to define information models related to a physical process. In a deployed environment, an OPC UA server has access to process information from the connected machines, and provides these data to OPC UA clients over the network. Replacing the real hardware with a simulation would enable seamless integration of the simulation into the rest of the network. This would then provide the benefits of the simultaneous simulation and real system execution.

OPC UA simulation servers already exist, however they are limited to static, immutable OPC UA information models, meaning that the customization of the models is not possible. Commercial OPC UA servers which allow the modification of the information model also exist, but they are intended for testing the client-server communication[1], and the data they provide are usually randomly generated, or generated by some pre-defined functions, not having any relation to an actual physical process. An OPC UA server which could connect with a simulation environment, and expose an information model, which is semi-automatically linked to the simulation output, would greatly simplify the CPS development process, by reducing the time requirements, in addition to lowering the possibility of introducing errors.

This problem statement yields the following research question: *What is an appropriate workflow to (semi-)automatically connect an automation system described by an information model to an existing simulation model - enabling the automation system to access the simulation data during runtime?*

In order to evaluate the proposed workflow, OPC UA will be used along with some chosen simulation environment, to develop a proof-of-concept implementation.

2. Expected Results

The main goal of this thesis is to establish a workflow, that needs to be accomplished in order to connect an already existing simulation of a physical process, with an automation system. This will enable access to these simulation data for the rest of the automation system. The chosen workflow will be compared to other possible approaches. The first sub-goal is the automation of the workflow, and the second sub-goal is to use the open standards to achieve other goals.

The resulting workflow is expected to roughly be defined by the following steps (also see Figure 1):
1) Obtain the simulation interface model, as well as the automation system's information model
2) Semi-automatically determine the links between the variables/data of the two models
3) (Re-)start the mapping unit with the new information model. The mapping unit then provides the simulation data, as determined and regularly updated, by the simulation interface, to the rest of the automation system.

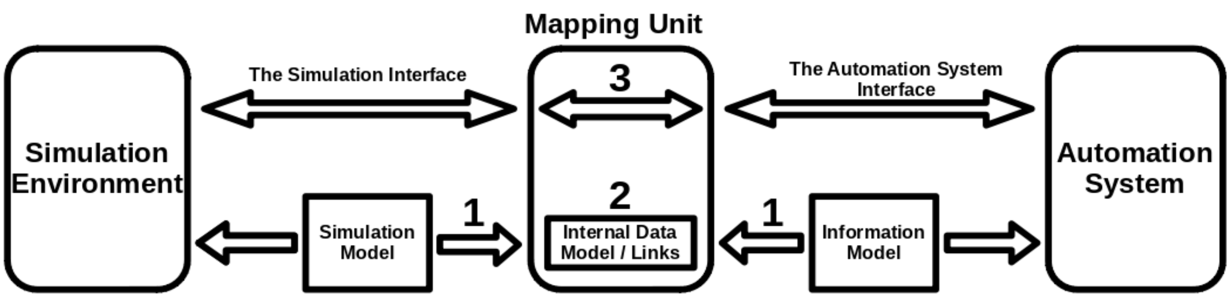

Figure 1: Graphical representation of the expected structure and workflow (numbered steps explained in paragraph above)

1 https://www.prosysopc.com/products/opc-ua-simulation-server/ ; accessed March 2021

However, the workflow shown in Figure 1 is not the only possible workflow. An additional possibility is to determine the information model based on the simulation model. The mapping unit would then define this mapping and provide an information model for the automation system, which the automation system developers would then adhere to. Another possibility is to move in the other direction, meaning that the mapping unit determines the simulation model, based on the information model. These, and other workflow possibilities should be evaluated and compared in the thesis. Finally, a proof-of-concept implementation using OPC UA will be used to evaluate the feasibility of the proposed workflow. Using Figure 1 as a reference, the mapping unit will be instantiated by an OPC UA server, and the automation system represents the OPC UA clients.

The process of mapping the simulation data to the data points of the automation system does not need to be fully automatic, however a certain level of support from the implemented program, or the procedure should be expected. It should be possible to modify the simulation and/or the information model dynamically, and ideally (but not necessarily) without recompilation. In theory, this would also enable the replacement of the simulation with the real components without the need for human intervention, provided that the real components use the same interface as the simulation. To this end, a standardized interface for the communication between the simulation and the automation system should be chosen.

3. Methodological Approach

The following methodological approach should result in fulfilling the goals of the thesis: firstly, a structured literature review will be conducted. Afterwards the workflow for a certain use case will be defined, and lastly, a proof-of-concept implementation will be constructed in order to evaluate the workflow. To this end, the following steps will be taken:

1) Literature review

    First, a review of the literature which is relevant to the topic will be carried out. The literature will provide insight into the current state-of-the-art, as well as the different approaches, possibilities, and methods which could be employed to achieve the goal of the thesis. Apart from this, the literature will also provide a theoretical insight into the problem at hand.

2) Defining the workflow

    In this step, the workflow to achieve the main goal of the thesis will be defined. The advantages and disadvantages of the possible workflows will be explored, after which a comparison of the workflows will follow. The chosen workflow will then be more thoroughly examined, and used for the proof-of-concept implementation (step 5).

3) Determining the mapping between the simulation model and the information model

    The next task will be the mapping between the simulation model and the information model. In order to define such a mapping, a good knowledge and understanding of the possibilities for the simulation and information models will need to be acquired first. Then, different constructs/data types in the simulation model will be mapped to their possible implementations in the information model, or vice-versa, depending on the chosen workflow. By reading both of the particular model files, the program/procedure should determine the most likely mapping between them, based on the information on the possible mappings. As designing a fully-automatic system for this task would be unfeasible, the program should offer the possibility of human intervention, by enabling mapping modification. This will ensure that the program's "understanding" of the mapping between both of the models is correct and complete.

4) Technology review

Different possibilities for the open-source libraries for achieving the goal of this thesis exist. A survey and analysis of possible libraries and frameworks will be done in order to ensure good understanding of the possibilities, as well as possible solutions. Firstly, a simulation interface, as well as the library which will enable communicating over this interface with the simulation will be chosen. The primary candidate here is the Functional Mock-up Interface (FMI)[2], with the FMI Library[3] which is written in the C language. Second, for the automation system, an OPC UA[4] stack will be chosen, to enable establishing the OPC UA server, as well as implementing the provided information model. The most likely candidate for this is the open62541 library[5], which is also written in the C language. Lastly, a library or a program will be chosen to interpret the models, and provide this information to the OPC UA server. This will probably be done by a custom-made XML parser, although some programs already exist.

5) Implementing a Proof-Of-Concept

Lastly, a proof-of-concept implementation will be implemented, following the defined workflow of the thesis, to show the feasibility of the workflow. An OPC UA server will be designed based on the mapping information obtained from the previous steps. The OPC UA server could either be constructed by the mapping itself - by generation of the C code, which has to be compiled - or by re-configuring the server and its information model during runtime, if it is provided by the chosen library. The second option is considered to be more favorable. The OPC UA server will then provide the information, as defined by the information model, and as provided by the simulation model, to possible OPC UA clients.

4 State-of-the-Art

The future of the CPS modeling and simulation is discussed in [WMO+16]. The authors argue that the cyber-physical production systems, which are integral to the future factory environment (smart factories) should be scalable and modular. The advantages of these approach include greater ease of integrating, adapting, and replacing individual production units, as required, based on the unpredictable market demands, or in order to maintain operability in the case of disruptions and failures. The authors pinpoint the concurrent combination of the physical world, and their digital counterparts as crucial in achieving these goals. Thus, the importance of simulation tools increases, as they support the engineering, re-engineering, and decision-making processes, as well as enable evaluation of external and internal changes, and the impacts they cause on the simulated system. Finally, in order to address the challenges of developing systems which can model and simulate components of future factories, the authors present a framework for modeling and simulation of CPS-based factories.

[NFCM19] presents a modular simulation approach with the FMI for use in creating a digital twin, which is an integrated simulation of the system, or some component(s), which mirrors the corresponding real life component/system (i.e. the twin), using internal models of the real component/system, as well as available sensor data. This paper uses a black box approach for the individual modules within the main simulation models, which simulate different behaviors of the system. These individual models are activated only when needed, and work in addition to the main simulation model. This achieves additional flexibility of the simulation.

2 https://fmi-standard.org/ ; accessed March 2021
3 https://github.com/modelon-community/fmi-library ; accessed March 2021
4 https://opcfoundation.org/about/opc-technologies/opc-ua/ ; accessed March 2021
5 https://open62541.org/ ; accessed March 2021

In the paper [GHIU17], the authors consider the advantages and disadvantages of OPC UA information models. The paper presents an overview of different ways and projects where OPC UA information models were used, and how they impacted the corresponding system. One of the key advantages identified by the authors, is the possibility for stronger decoupling of the client and the server in applications, while simultaneously making applications more flexible. In particular, the following use cases for OPC UA are considered by the authors: modular automation, co-simulation, and integration of field level devices with high data throughput. The authors also identify some of the disadvantages of the OPC UA, such as: limitations with regards to object orientation, server aggregation, and information model revisioning. Finally, the authors concluded that the OPC UA, and its information models are suitable for new applications in the area of digitalization in process industries, by allowing the construction of flexible and smart applications.

The possibility of integrating a dynamic simulator and advanced process control using the OPC UA standard has been considered in [LSS19]. The large amount of data generated by the manufacturing industries is difficult to exchange among the various manufacturing systems, applications, and components, due to the variety of the underlying systems. A middleware such as OPC UA could be used to exploit the existing data, and achieve higher efficiency by bridging the communication gap between the systems. OPC UA presents a scalable, platform independent, user friendly, and secure approach. The paper adopts a simplified version of the Tennessee Eastman - which is a well-known industry problem, as a use-case on which the simulation and control modeling is performed. The authors have also identified some of the difficulties when using OPC UA, such as: the set of services in an OPC UA server which is used by the client are defined in the OPC UA standard and are not changeable by the user, and the security tiers are difficult if not impossible to control, after the application platform is chosen. The authors also point out that performance measurement of the OPC UA connection is not present, and that the level of reliability and the quality of connection are hard to understand.

In their paper [JSW18], the authors explore the challenges, and their possible solutions, regarding the development of IoT system simulations. In particular, the paper presents an approach for dynamic co-simulation of IoT systems using a multi-agent-system. This means that different IoT components are simulated in separate simulation tools. Bringing different simulation tools together presents a great advantage, as the different IoT components are often produced by different manufacturers, and may have different requirements; thus they need to be simulated in different simulation tools. The approach is dynamic because the individual simulations - represented by agents, can enter the simulations that are already in progress, dynamically during runtime, which is referred to "Plug-and-Simulate". The authors evaluate the presented approach by constructing a prototype, which then successfully solved the challenges of heterogeneity and dynamicity in the simulation of IoT systems.

The concept of automatic integration of simulation systems with OPC UA has been explored in [RR20]. A data mapping between the simulation meta data model, and the OPC UA information model is defined in both direction, and the communication is ensured with message passing between the OPC UA server and the simulation software. The simulation software chosen is VEROSIM - an extensible simulation framework. The mapping is used to automatically generate OPC UA address spaces from the simulation models, and the inverse mapping (simulation model to OPC UA information model) is used to generate simulation model structure from queries to existing OPC UA servers. Large number of concurrent simulated objects is possible, with individual and independent OPC UA servers and clients. Thus the approach of this paper is especially useful for the distributed simulation. Finally, the approach is tested in an automotive production line simulation, where the results show that the implementation can handle significant network load.

The most important difference to the proposed master thesis is the fact that this paper presents a distributed solution with individual OPC UA servers, implemented for the VEROSIM simulation framework. The intended goal of the proposed master thesis is the creation of a single, configurable OPC UA server, which is capable of communication with various simulation environments, based on standardized interfaces.

5 Relevance to the Curricula of Computer Engineering

The proposed thesis is concerned with the simulations of components in industrial CPS, and in particular as part of the Industry 4.0, which is one of the key points of computer engineering. The machine-to-machine communication, related protocols, and automation within system engineering are all topics which concern computer engineering. Additionally, being able to automatically generate and (possibly seamlessly) incorporate different parts of systems, and to precisely define their interaction greatly contributes to ease of system design and reduces the chance of human errors. Such automation, precise component interactions and functionality are all core ideas within computer engineering.

In particular, the topic of this thesis is concerned with the ideas which are encountered in the following subjects:

191.104 Information Technology in Automation
183.168/9 Dezentrale Automation

6 References

[WMO+16] - WEYER, Stephan, et al. Future modeling and simulation of CPS-based factories: an example from the automotive industry. *Ifac-Papersonline*, 2016, 49.31: 97-102.

[SSK+20] - STEINDL, Gernot, et al. Generic Digital Twin Architecture for Industrial Energy Systems. *Applied Sciences*, 2020, 10.24: 8903.

[Bac05] - BACIC, Marko. On hardware-in-the-loop simulation. In: *Proceedings of the 44th IEEE Conference on Decision and Control*. IEEE, 2005. p. 3194-3198.

[JOM16] - JIRKOVSKÝ, Václav; OBITKO, Marek; MAŘÍK, Vladimír. Understanding data heterogeneity in the context of cyber-physical systems integration. *IEEE Transactions on Industrial Informatics*, 2016, 13.2: 660-667.

[IEC62541] - International Electrotechnical Commission 62541 - OPC Unified Architecture, 2015

[PSM+17] - PETHIG, F., et al. Industrie 4.0 Communication Guideline Based on OPC UA. *VDMA, Fraunhofer IOSBN-INA, Tech. Rep.*, 2017.

[NFCM19] - NEGRI, Elisa, et al. FMU-supported simulation for CPS digital twin. *Procedia manufacturing*, 2019, 28. Jg., S. 201-206.

[GHIU17] - GRAUBE, Markus, et al. Information models in OPC UA and their advantages and disadvantages. In: *2017 22nd IEEE International Conference on Emerging Technologies and Factory Automation (ETFA)*. IEEE, 2017. S. 1-8.

[LSS19] - LATIF, Hasan; SHAO, Guodong; STARLY, Binil. Integrating A Dynamic Simulator and Advanced Process Control using the OPC-UA Standard. *Procedia Manufacturing*, 2019, 34. Jg., S. 813-819.

[JSW18] - JUNG, Tobias; SHAH, Payal; WEYRICH, Michael. Dynamic co-simulation of internet-of-things-components using a multi-agent-system. *Procedia CIRP*, 2018, 72: 874-879.

[RR20] - REITZ, Jan; ROßMANN, Jürgen. Automatic Integration of Simulated Systems into OPC UA Networks. In: *2020 IEEE 16th International Conference on Automation Science and Engineering (CASE)*. IEEE, 2020. S. 697-702.