Simulation-based Control of Reconfigurable Robotic Workcells: Interactive Planning and Execution of Processes in Cyber-Physical Systems

Marc Priggemeyer and Jürgen Roßmann
Institute for Man-Machine Interaction, RWTH Aachen University, Aachen, Germany

Abstract

Simulation-based Control offers a concrete realization for the concepts introduced by the definitions and standards concerned with Cyber-Physical Systems in manufacturing processes. The basis for this realization is a simulated virtual environment that is tightly coupled with all the physical assets involved in industrial applications. This paper provides an in-depth reconciliation of the ideas of the application of Virtual Testbeds in terms of Simulation-based Control with the historical layer scheme of the Purdue Enterprise Reference Architecture (PERA) and the very recent Reference Architectural Model Industry 4.0 (RAMI 4.0). The idea of Simulation-based Control is illustrated by the application of the outlined aspects to a reconfigurable robotic workcell for industrial production lines. Since a variety of internal tasks are involved in the execution of a pre-planned manufacturing process, details covering technical preconditions for system models and communication protocols are presented. In this context, the high-level development of control schemes in a virtual environment (VTB) allows an efficient way to define manufacturing processes that can be directly deployed to the processing hardware.

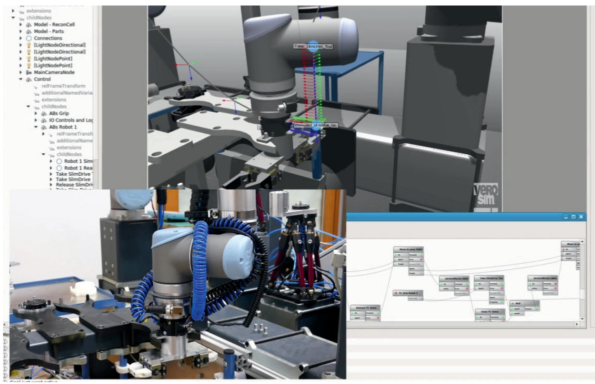

Figure 1 Execution of process steps in the simulated and in the real workcell (ReconCell)

1 Introduction

Due to shorter product life cycle times in few-of-a-kind productions, flexible ways to reconfigure manufacturing processes are required. In addition, the fast evolution of products requires easy reprogramming, flexible change of production lines and the possibility of non-expert intervention in the actual manufacturing automation to put the focus on product quality. Application of Cyber-Physical Systems to these use cases allow to cope with the aforementioned challenges.

Assembly processes get arbitrarily complex and require more complex manipulators and field devices for the process control. This requirement for smart devices also indicates the change from classical hierarchical communication structures to meshed communication networks. In such setups, single devices may run asynchronously but require serialization or parallelization techniques in the form of automatisms that schedule the execution of tasks, or programming paradigms for manual process design and planning. Field devices may also communicate synchronously, which imposes hard real-time constraints that have to be met on the lower communication layers. This intricacy goes beyond vendor specific solutions for multi robot setups, which are merely applicable in the case of a few cooperating machines. However, recent robot controllers provide programming language features for sophisticated motion primitives. While these solutions also provide the real-time interfaces to program and operate single manipulators, the capabilities for device orchestration in large facilities is rather limited.

To address this issue, we present an approach called Simulation-based Control (SbC) that introduces a transparent function layer that keeps an operator's focus on the simulation on component level. We adopt the vocabulary introduced by the Reference Architectural Model Industry 4.0 (RAMI 4.0 [1]) to achieve a straight-forward terminology. A virtual testing environment [2] is the core idea behind this technology to provide all tools for the development, planning and optimization of manufacturing systems before the actual transfer to the shopfloor. Virtual Testbeds (VTBs) implement this virtual environment using a simulation system, that the user can take full advantage of while complying with reality and the physical process (as shown in e.g. [3]). A VTB's focus lies on the application-oriented development of complex systems. It is then used for reproducible testing in an inherently safe environment that can be interactively modified and optimized until a desired mode of operation is achieved. Afterwards, the transfer of simulation results to the physical workcell is an important step towards the implementation of Cyber-Physical Sys-

ISBN 978-3-8007-4699-6

tems. This transfer is the central aspect to mirror the behaviour and state of actuators between reality and virtuality.

The simulation model specifies the foundation for the system description and its execution, since it comprises all the technical assets and the means by which they are interconnected. Descriptors for all services and domain specific functionalities offered by the assets and how these can be accessed. A precondition for the simulation is the precise mapping of the assets' functionality and their properties onto the different simulation domains (rigid-body dynamics, sensors, kinematics, ...), such that each can be commissioned in the virtual environment. Besides that, state information is introduced such that the assets can be tracked throughout their life cycle in the value chain and process specification. Exactly this process specification is what makes the simulation system essential. The interplay of components is characterized by fine-granular process steps that utilize the components in the virtual environment to form an exact replica of the physical setup. That way, the whole setup can be extensively tested and optimized before components are exchanged (switched) by their physical counterparts.

Simulation-based Control implements this switch by introducing a set of major techniques, which will be further outlined in Section 3. In short, the objective is to simplify the creation of Cyber-Physical Systems as stipulated in Industry 4.0 [4]. During the planning and development phases, the aim is to have an application-oriented experience without the need of putting the topology of a setup in the foreground. It is much more important to create a system model with its functionality and outcome in mind. In [5] Schluse proposes a new structural element (called Experimentable Digtal Twin - EDT) for Simulation-based Systems Engineering that provides the tools covering the aforementioned need of developing applications on the system or even the component level. An EDT is a one-to-one replica of a real system, applicable and modifiable in the virtual environment that allows just that: virtual commissioning of individual components and complex systems comprising a set of components. With SbC we complement EDTs to make the physical commissioning possible.

We will also present a classification of Simulation-based Control given the historical Purdue Enterprise Reference Architecture (PERA) and the recent Reference Architectural Model Industry 4.0 (RAMI 4.0) to identify how SbC is applied in Cyber-Physical Systems in general. To that effect, Section 2 summarizes current research efforts and briefly recapitulates both, RAMI 4.0 and PERA. The concepts behind SbC are outlined in Section 3. That section includes the basic ideas how Cyber-Physical Systems can be created and how the transfer between virtuality and reality is achieved.

We will reconsider these results by taking into account an exemplary industrial robotic workcell (ReconCell) and how it is modelled, programmed and operated in the virtual and in the physical environment. A more detailed overview of what a ReconCell is can be found in [6]. Figure 1 depicts a sample sequence of a robot action in such a workcell. A standard 6-axis industrial robot, equipped with a pneumatic tool exchange system, is instructed to take a gripping tool from the tool tray. The coordination of gripping action and robot movement is implemented by a visually programmed sequence (lower right). The movement can be executed purely in simulation (upper part of Figure 1) or even be transferred to the real cell (lower left). Motion and gripping events are synchronized between the virtual and the physical environment.

2 State-of-the-Art

There is no clear definition of boundaries that form a Cyber-Physical System, but rather a set of focal points, which characterize recommendations for their realization [7]. One is e.g. the virtualization of technical assets, which provide a value by themselves due to their representation of data collections that are useful for further process optimizations. The orchestration of the services provided by these entities involves their management and interconnection to form business processes. A five layer architecture for the implementation of CPSs is given in [8] by defining a set of layers that should be covered by each manufacturing application to successfully create a business structure. The authors identify techniques within each of these levels to generate meaningful information to track a systems life-cycle and analytical state.

2.1 Control of Cyber-Physical Systems

In 2011, actual control methods for Cyber-Physical Systems were considered to still be in an elementary state [9]. Yet, in the meantime, the actual control of Cyber-Physical Systems has been subject to thorough research and a variety of solutions have been presented on different levels of automation. Only considering the publications of the last year, an extensive set of research results on the control of Cyber-Physical Systems can be gathered. That set includes several outcomes that include purely mathematical and system theoretical control schemes, approaches originating in information theory and also hybrid solutions that involve simulations to derive performance indicators.

In [10] a representative model of a set of agents is utilized to stabilize the individual agents. A supervisor controls the overall process by analysis of distinct outputs. It provides a system theoretical view on the subsystems and generates performance data for the whole process.

The authors of [11] combine discrete event and continuous time control in a petri net representation of a system model. A specialized language to formalize the problem description as a petri net combines a number of system models to be controlled by a centralized entity.

A method for distributed task ordering and scheduling is presented in [12]. The scheduling of tasks includes on-the-fly addition and removal of single tasks for sequential processes to be executed by actuators.

[13] outlines an agent-based modelling technique to formalize and simulate hybrid Cyber-Physical Systems, including the discrete event and continuous time modelling paradigms. The benefits of simulation of such systems are

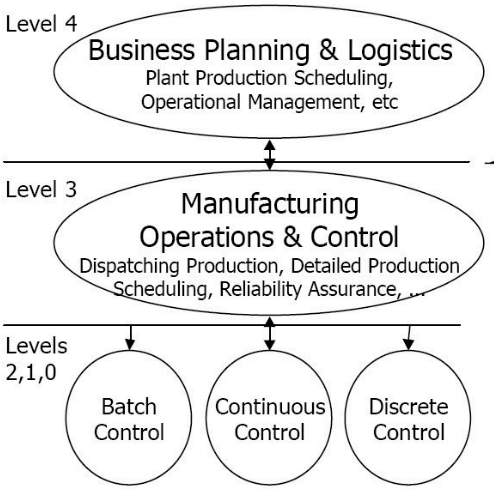

Figure 2 Purdue Enterprise Reference Architecture (based on [16] and [17])

presented by the authors, stating that it gives a deep insight into the system behaviour and allows for in-depth analysis of the system's performance and efficiency.

2.2 Purdue Enterprise Refernece Architecture

The Purdue Enterprise Reference Architecture (PERA) was defined in the 1990s at the University of Purdue to provide guidelines for the establishment of business processes and declares 5 layers (0 - 4) on which automation of such processes can happen. It builds the foundation on which the IEC 62264 [14] and IEC 61512 [15] standard collections are defined, which impose strict rules on the integration of enterprise control systems. Mainly due to differences in terminology, technical languages and various organization cultures, the standard was defined to cope with these difficulties and to ensure comparability of operations while focusing on the organizational structure without the inclusion of specifics of different industrial branches [16].
The actuality and applicability of PERA is shown by its constant evolution (latest revision in 2013 [17]) and its inclusion and extension in even more recent developments like RAMI 4.0 [1]. Extensive work has already been put into the analysis of IEC 62264 to identify its value in the manufacturing industry (see e.g. [18], [16]).
DIN EN 62264 outlines hierarchies (e.g. equipment, functions) and models (e.g. data flow, object) especially for the upper levels (Level 3 & Level 4), but also illustrates how the interface between Level 3 and Level 2 can be utilized, i.e. how the production management influences the process supervision and how process information is fed back to the management level. The hierarchies for example specify how units, process cells and areas integrate into an enterprise, while for example the data flow model describes which information about a specific asset is available and how their actual values are formally defined.
On the other hand, DIN EN 61512 focuses on the three lower levels providing a detailed definition on what a process is and how batch control for processes has to be established. This part is of particular interest, especially when considering the following section, since the functions of Simulation-based Control directly influence the levels 0 - 2. These levels include information about the physical process and the involved workpieces and tools. They handle data acquisition for sensors and implement specific controllers for motion planning and tasks. In addition, process supervision for one distinct line of execution is also handled here (e.g. to ensure quality of service).

2.3 Reference Architectural Model Industry 4.0

In contrast to PERA, which spans only one dimension, RAMI 4.0 spans three dimensions (see Figure 3) and therefore takes into account more information when considering single assets that are involved in a CPS [1]. A variant of the aforementioned PERA represents one of the three dimensions. Hence, RAMI 4.0 also includes the enterprise hierarchies and models that were already state of the art in non Industry 4.0 contexts. The notion of spaces (namely *Cyber* and *Physical*) makes the difference when comparing Cyber-Physical Systems to classical systems, because it allows different representations of a single component: Even though a workpiece, a tool or even a facility might still only be what it was initially intended to be in the *Physical* space, it will now be managed as an technical asset that has meta-information in the *Cyber* space.
The other two dimensions in RAMI 4.0 require a minimal amount of meta-information for each asset. The architecture layers (*Business*, *Functional*, ... see Figure 3) define the functions and the specific data contained therein. In addition, the life cycle value stream is used to accurately describe an asset's state throughout its life cycle. This information includes its type and location, which both might change over time.
RAMI 4.0 specifies that a Cyber-Physical System provides manifests for the included assets. They include meta-information about the assets functional and purely informational properties. DIN SPEC 91345 [1] provides a much deeper insight into RAMI 4.0.

2.4 Experimentable Digital Twins

Schluse proposes in [5] Experimentable Digital Twins (EDTs) as a structuring element in a Virtual Testbed. We will use this formalism throughout the next sections, since it provides the necessary functionality and properties of a

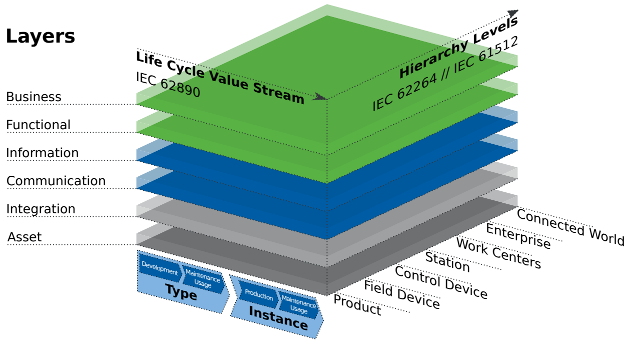

Figure 3 Reference Architectural Model Industry 4.0 (based on [1])

technical asset's representation in *Cyber* space. To briefly summarize, an EDT is a one-to-one replica of a real asset. That means, it (ideally) shares all the properties and functions that can also be accessed and utilized independently in a virtual, simulated environment. It therefore complies with the requirements introduced with RAMI 4.0 and also provides the extensive functions triggered by a simulator in a virtual environment that goes way beyond purely informational data.

3 Simulation-based Control

The term *transfer* will be used throughout this section to indicate the control by simulation of individual actuators and the execution facilities for complex processes. This means that a simulated instance's (i.e. the virtual replica's) state is imposed on the corresponding real asset.

In general, we presume that the transfer from *Cyber* space (i.e. virtual, simulated environment) to *Physical* space (i.e. reality) is an inherent feature in the rulesets and guidelines for the creation of CPSs. Hence, in every Cyber-Physical System we require, that it should not only possible to track the real technical asset's state in real-time, but also to transfer it's virtual state back to reality. Even though a number of aforementioned rulesets for the creation CPSs already exist (see Section 2), it is undefined how such a transfer of simulation results can be achieved in multi-domain simulation systems.

Simulation-based Control (SbC) focuses on a set of major methods to allow a flexible and efficient execution of processes in industrial applications: *Process Control & Real-Time Simulation*

Process control includes control strategies for individual sub-systems and complex facilities comprising a variety of actuators and sensors that are merged into one virtual environment. Real-Time Simulation allows for the acquisition of data of all individual assets that is afterwards combined in one simulation database. This database actively manages the meta data of all technical assets and thus provides access to the entities manifests, control objectives, goals and outcomes. Partitioning helps in the structural distribution of process sequences where direct interaction between any number of sub-systems is not necessary and only the outcomes of each sequence is aggregated to influence the process control on a larger scale. Simulation models are horizontally partitioned and distributed onto a set of distinct computers. The following subsections outline these summarized points further.

In general SbC leaves the focus on the engineering process of Cyber-Physical Systems. Since most thus involved steps are achievable in a Virtual Testbed, simulation models can be verified and optimized in a safe environment [19]. As previously mentioned, the resulting simulation database holds precise information about assets and the process that is to be executed. Figure 4 depicts an exemplary robotic workcell (ReconCell) that is operated by application of Simulation-based Control. It incorporates a number of Experimentable Digital Twins, which combined form another EDT representing the workcell. Some

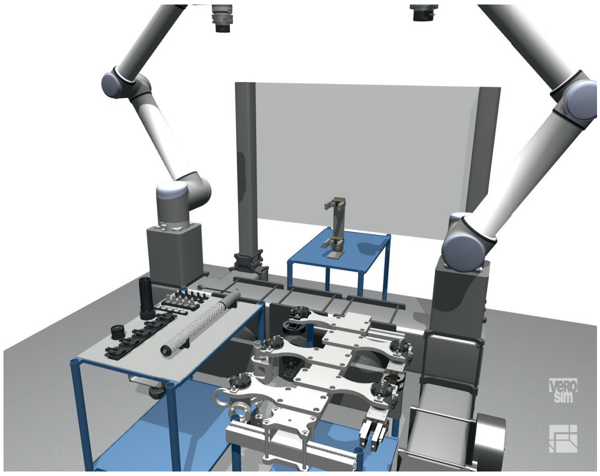

Figure 4 3D simulated ReconCell in a Virtual Testbed including actuators, tools and workpieces.

of these EDTs serve specific functions, like the actuators that allow the manipulation of other assets. The problem is, how the EDT triggers actual motion on the physical asset.

Simulation-based Control solves this by introducing solutions for the interfaces and constraints imposed by the devices and systems. As a result, setups, incorporating SbC as their central control concept, allow for the virtual development, optimization and deployment while linking specifications for hardware properties and requirements. Due to the tight coupling between spaces in Industry 4.0, the well established idea of plug and play devices still plays an important role here. The term is widely used synonymously with the expression ease-of-use and in the context of industrial automation by SbC implies the virtual commissioning and afterwards the transfer to reality with minimal effort. For the successful application, requirements need to be defined in the different spaces as well as on the layers of the reference model. Entities need to be defined and integrated in the VTB and all requirements on the assets, network topology, communication protocols and mechanical interfaces are modelled and setup in the virtual environment. The technical assets' respective representations in the physical space must conform to this.

3.1 Process Control

In ReconCell we aim for two approaches to the actual cell control. A high level approach, which is based on petri nets forming the ActionBlock paradigm [20] that allows for a visually appealing view on the process description. We model complex processes by adding single ActionBlocks to the workcell program. These ActionBlocks can either be single process steps (e.g. motion or gripping actions) or more elaborate process operations that define self-contained tasks (e.g. pick and place). An example ActionBlock network is depicted in Figure 5. This high-level process description provides additional information to the operator. First, since the process can be executed in the virtual environment, the process can be thoroughly reviewed before transferring the sequence to reality. Second, the operator can collect operational data, providing

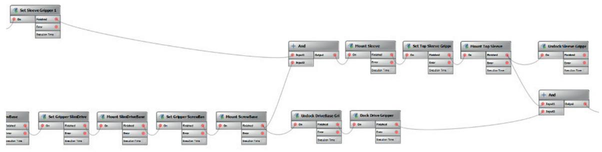

Figure 5 A sample ActionBlock network used for the high-level process control in ReconCell

the asset's state and performance data of a production line, throughout execution of the process.

The second approach, real-time simulation (see Section 3.2) provides the state information about the actuators and the services to control them directly. These services might have elementary interfaces, for example to pass setpoints, either cartesian or specific to the robot axes, that the real-time simulation will carry out and pass to the physical asset. In [21] we already presented some practical applications of SbC utilizing online path planning that directly executes on the real-time simulation system and that immediately reacts to the virtual environment (e.g. motion simulation).

The same principle is applied in ReconCell. We define different tasks that need to be carried out in the workcell. One of these tasks is for example the reconfiguration of a flexible fixture. Such a jig provides adapters for the tool exchange systems. It can thus be moved to different positions by a robot without manual intervention. This autonomous changeover process for a production line to support different or modified workpieces is exemplarily shown in Figure 6 as plots of a UR10's axis positions inside the considered ReconCell. The dashed lines represent the measured axis positions and the solid line the setpoints over time. In this case we did not take any measures to minimize the delay between measurement and command. Therefore, a delay of exactly one interpolation cycle is observable between the plots.

A sensible use case where real-time control is helpful, would be an application requiring situational awareness. Continuous supervision of the environment's state is performed. Then, the control strategies might be adapted (e.g. the motion planning parameters) if the environment has changed and hardware or human resources are endangered. In ReconCell, the reconfiguration of a fixture, as proposed in the previous example, might block the path of a manipulator. Situational awareness requires reconsideration of the planned path to possibly find another solution that is collision free. Due to the fact that the simulation system keeps

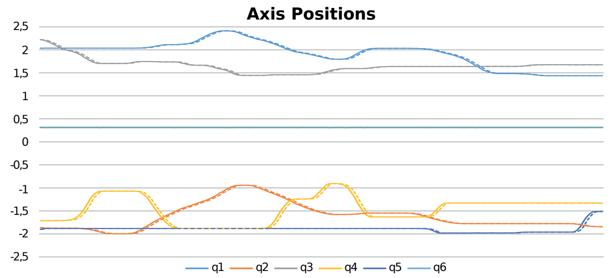

Figure 6 Sampled UR10 positions during a reconfiguration process in ReconCell (solid: commanded, dashed: measured)

track of the workcell status (poses of fixtures, progression of workpieces, positions of other manipulators), SbC outperforms the capabilities of conventional robot controllers that do not include this extensive information about the workcell.

3.2 Real-Time Simulation

In general, real-time systems are only practicable in a limited area of applications. Their design is usually very specific to the desired use case and they do not pose a viable general solution for a family of problems. In the case of Simulation-based Control, real-time, and therefore time deterministic, behaviour is intended and also necessary when it comes to the actual communication with a technical asset.

Vendor-specific solutions for external motion control of actuators are not beneficial for the direct use in the case of a reconfigurable workcell. At some point it might be necessary to exchange one robotic system for another. While it is generally easy to do this change of cell design in the virtual environment of a simulation system (the model is replaced with a mouse click), this might not be feasible in the physical setup: Is the network topology still applicable? Are communication protocols compatible? Are the timing constraints different?

Therefore, the demands on the simulation system include a specification of the supported protocols (including an actual functional implementation) and an abstraction layer for the (non) real-time communication with the technical assets. Thus, the VTB is applicable for the use case development, which includes the process description in form of robot programs, while the assets is still exchangeable.

We consider a multi-domain simulation system that is not only concerned with the control of systems, but also with the visualization, interaction, dynamics, sensors, etc. Figure 7 shows a computer network that we will take into account for further considerations. On the left side of Figure 7, one computer (the master node) loads the full simulation model that executes the VTB. In principle, the whole virtual environment could be simulated here with all aspects, but then the transfer to the physical assets would not be possible if they demand any timing constraints. Real-time constraints impose high demands on the simulation system that we resolve by distributing sub-models, that need to be executed in real-time, to dedicated computers (horizontal partitioning). We use a hub-and-spoke network topology that connects a number of computers via a L2 switch. The benefit of this topology is that it is compatible

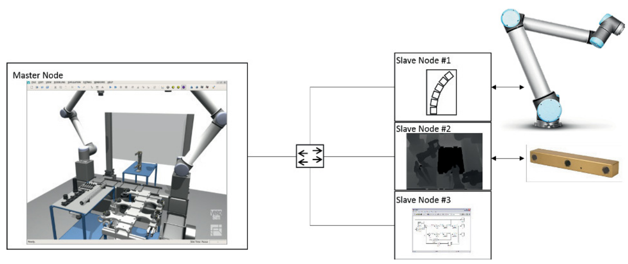

Figure 7 Sample SbC setup with master and slave nodes for the real-time control of actuators.

ISBN 978-3-8007-4699-6

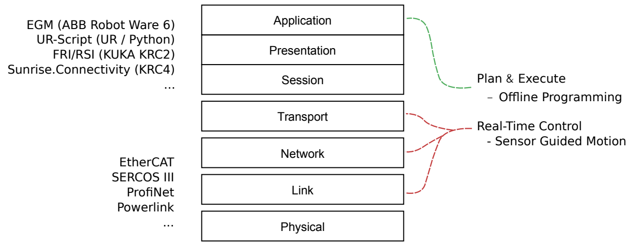

Figure 8 ISO/OSI Layers with real-time capable communication protocols on the Link and Presentation layers.

with most Industrial Ethernet protocol variants. The slave nodes on the right side of Figure 7 deliver the capacity to run single aspects of the simulation. This might be the data analysis for a specific sensor or a purely virtual asset, but it also includes the control functions and services that are described in Section 3.1. To achieve the partitioning, the simulation model of the technical asset, which e.g is to be controlled, is loaded into the simulation system on a dedicated computer. The master node only loads a shallow copy that passes service or property request to the respective slave.

In Figure 7 we consider the previously described ReconCell setup with two robotic manipulators. The master node provides a 3D view on the virtual environment, while slave #1 takes over real-time control.

Figure 8 depicts the ISO/OSI layers with annotations on how different real-time protocols integrate with two programming options. The Plan & Execute scheme is important for non-real-time control of robotic workcells. We use the ActionBlock paradigm to visually program the robotic workcell on a high level of abstraction. In ReconCell, if a gripping tool is replaced by another, the process description does not need to be changed. The asset is therefore replaced in the virtual environment as well as in reality and the process can still be executed. This example does not take into account the mechanical interfaces for the tool exchange system and the workpiece: it is assumed that the gripper is replaced by one with the same tool exchange adapter and the same fingers such that it is still useful in the application scenario.

In contrast to the non real-time part, the actuators are directly controlled on a low level using one of the real-time communication protocols. The upper right part of Figure 7 shows one partition that executes part of the simulation model in real-time. We utilize the operating system QNX [22] to impose our real-time constraints on the virtual environment that, in case of the actuator control, accepts set points as inputs and generates position updates, which are forwarded to the actuator. If the actuator asset is exchanged, this part of the simulation model also changes and a different kinematic model and communication protocol is used, but the high level program for the workcell is still applicable.

As a result, the cell in the virtual environment always replicates the cell in reality. Figure 9 illustrates this with a side-by-side view on a ReconCell both in the virtual and physical environment.

3.3 PERA Classification

PERA provides design rules for the instantiation of business processes in a general form [23] but in the context of industrial automation allows to classify the basic control concepts of SbC intersecting the lower PERA levels (especially in terms of levels 0.5 -2, see Figure 10). Of great importance for SbC is the independence of specialized hardware on the lowest level, which avoids vendor lock-in and allows for the flexibility that is necessary to control processes of different kinds. These processes might also be defined outside of an industrial context [24] but will not be further discussed here. Above that, on Level 1, a decentralized control instance fuses data obtained from intelligent devices to form complex control schemes, which are to be executed either in virtuality or in reality. These schemes are either as low-level as basic motion planning and as high-level as the planning of collision free paths based on the current environment state. For a better impression on the impact of Simulation-based Control, it's intersecting property on PERA [17] and the extension to interfaces with business processes behind the control of a smart factory is outlined (see also Figure 10) as follows:

All workpieces, tools, actuators and structural elements are situated on the lowest level (Level 1). They are all integrated in the physical process that is to be controlled by simulation. For SbC to operate correctly, a one-to-one correspondence between the physical parts and their virtual counterparts is given. During process control however, the simulation system interacts with the physical parts via interfaces: either hardware or software. Even though all the meta-information is available, Level 0.5 is a necessary addition to achieve that. These physical interfaces are defined and manifested in the models utilized by Simulation-based Control. Calibration data is available for a direct mapping between reality and virtuality. Thus, control processes are executed with high accuracy.

Assets on the sensing and parametrization level in the Purdue scheme adapt to local changes and deviations due to inaccuracies caused by (e.g.) the imperfect placement of workpieces. Here, we do not consider those displacements, which are static and could be coped with on Level 0 by the application of calibration data. SbC rather provides dynamic displacements in an algorithmic way by analysis of sensor data. In addition to that, parameters, e.g. for the motion planning, are defined on this layer and provided on a per simulation model basis for the respective actuator.

Process control and supervision includes the overall ad-

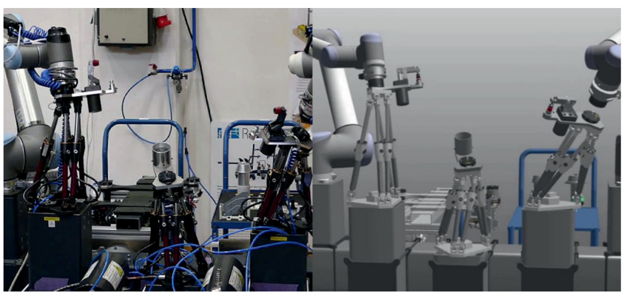

Figure 9 Side-by-side view of a ReconCell (left: system in *Physical* space, right: system in *Cyber* space).

ISBN 978-3-8007-4699-6

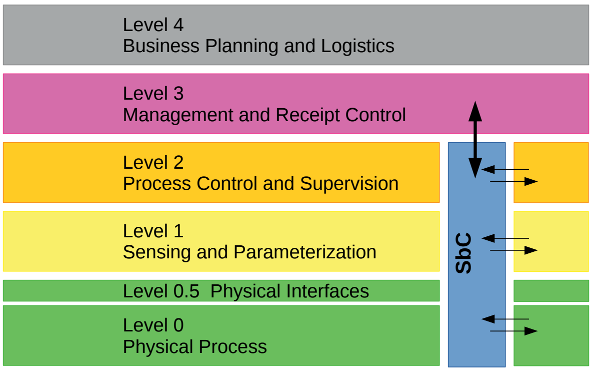

Figure 10 Simulation-based Control intersecting the Purdue level scheme (based on [16] and [17])

ministration of all involved assets in a single robotic workcell. This includes the synchronization of tools and assets in the workcell to securely operate a set of actuators in parallel. Therefore, Simulation-based Control adapts the planned paths based on the status information about a workcell.

Since all of these interferences on the lower levels also influences the organizational part of the process control, the SbC intersection in Figure 10 acts as a layer on itself offering an information flow to and from the Management and Receipt Control level of PERA (Level 3). Important parameters and performance indicators are thus gathered and forwarded to the upper levels. This is helpful to evaluate the SbC performance.

3.4 RAMI 4.0 Classification

Since PERA specifies the hierarchies and models for an actual enterprise and therefore the processes performed in the integrated facilities, it builds the basis for the *Physical* space in Industry 4.0. Here, all the technical assets are actually present as physical and tangible instances. Mechanical and communication interfaces impose constraints that cannot be dismissed if the process implementation is supposed to be effective. Like with PERA, we therefore rely on precise systematic simulation models for a virtual environment to successfully control the physical system. Thus, we still rely on the SbC level, that we introduced in Section 3.3, and its integration and information flow in the hierarchy levels of RAMI 4.0 is exactly the same as depicted in Figure 10.

In contrast to the one-dimensional hierarchy of PERA, that we had to disrupt to allow an information flow between non-adjacent levels, RAMI 4.0 provides two additional dimensions that we utilize. We can leave the hierarchy levels intact, while still allowing to overcome the interface barriers that were originally imposed between any two non-adjacent levels (compare SbC interfaces in Figure 10). We use these two dimensions as the basis for the *Cyber* space.

First, according to DIN SPEC 91345, the architectural layers specify the architecture of the CPS according to their functions and their meta-data. The *Business* layer is not directly involved with Simulation-based Control, yet SbC provides an interface to that layer by providing (e.g.) performance indicators that can be included in business models. Still, since RAMI 4.0 charges the business layer to perform orchestration task on services in the *Functional* layer, it plays a role in the previously outlined partitioning (Section 3.2) of a simulation model.

We extensively used the term *simulation model* in this section. In our case, it is a (static) structured document comprising the composition of assets. It mirrors the hardware setup, i.e. it includes all assets and components that are essential for the successful application of the CPS to its respective task. Therefore, part of the model is situated in the *Integration* layer. The CAD data is accessible here for any function that requires a component's geometric information (e.g. a path planning algorithm). Additional further semantics like kinematic models, dynamic parameters or real-time constraints (besides others), which are formalized in the simulation model, are located in the *Information* layer. We do not use the *simulation model* synonymous with the term *manifest*, it is rather one of the basic sets of information that is used during the creation of the manifest, since the model does not yet include all of the functions and services offered by the simulation system.

Process Control & *Real-Time Simulation* are all functions or services, which are also formally specified in the *Functional* layer. Their implementations (e.g. concrete controllers or communication protocols) are situated in the *Information* layer.

Second, the Life Cycle & Value stream, according to DIN SPEC 91345, represents the CPS state over time. Especially in the case of a ReconCell, which includes reconfigurable elements, the change of the CPS state is an important operational data element, as it provides useful information for cell or process optimizations and the adaptation of motion parameters. This fact especially concerns the *Information* layer, since this is the location we keep asset properties over time.

4 Conclusion

In this paper we link the current development of the Reference Architectural Model Industry 4.0 to Simulation-based Control. It is a technology that allows to transparently transfer simulation results from the virtual environment to the real system. During the development of a Cyber-Physical System, the focuses of attention should be on the system functionality and validity. Therefore, simulation offers the tools to virtually commission the CPS in a safe virtual environment, to test, verify and optimize until a sophisticated solution for a problem description is found.

The goal of Simulation-based Control then is to simplify the interfaces between virtuality and reality. The integration of specifications, properties, requirements and interfaces into the simulation model provides the necessary information to use the virtual system's state to actually control the physical components.

We used the standards for PERA and RAMI 4.0 to classify Simulation-based Control in the context of Industry 4.0 and answered how SbC can be applied to operate CPSs and therefore, how the switch between virtuality and reality can be realized. Especially the hierarchy levels are of importance, since SbC focuses on the process execution. Nev-

ISBN 978-3-8007-4699-6

ertheless, the other two dimensions in RAMI 4.0 allow to define the functions and state information over time that are necessary to realize a process in a CPS. Thus, Simulation-based Control encloses the methods for the replication of an asset's state in the physical environment.

ACKNOWLEDGEMENT

This project has received funding from the European Union's Horizon 2020 research and innovation program under grant aggreement No 680431 (ReconCell).

5 Literature

[1] D. SPEC, "DIN SPEC 91345:2016-04 - Referenzarchitekturmodell Industrie 4.0," 2016.

[2] J. Rossmann *et al.*, "A new approach to 3d simulation technology as enabling technology for erobotics," in *Proceedings of the 1st International Simulation Tools Conference & EXPO (SIMEX)*, 2013, pp. 39–46.

[3] J. Rossmann, T. Steil, and M. Springer, "Validating the camera and light simulation of a virtual space robotics testbed by means of physical mockup data," in *International symposium on artificial intelligence, robotics and automation in space (i-SAIRAS)*, 2012, pp. 1–6.

[4] Plattform Industrie 4.0. (2017, 6) Industrie 4.0 plug-and-produce for adaptable factories: Example use case definition, models, and implementation. https://www.zvei.org/presse-medien/publikationen/industrie-40-plug-and-produce-for-adaptable-factories-example-use-case-definition-models-and-implementation/ (visited on 2018-03-29).

[5] M. Schluse, M. Priggemeyer, L. Atorf, and J. Romann, "Experimentable digital twins - streamlining simulation-based systems engineering for industry 4.0," *IEEE Transactions on Industrial Informatics*, vol. PP, no. 99, pp. 1–1, 2018.

[6] T. Gaspar, B. Ridge, R. Bevec, M. Bem, I. Kovač, A. Ude, and Ž. Gosar, "Rapid hardware and software reconfiguration in a robotic workcell," in *Advanced Robotics (ICAR), 2017 18th International Conference on*. IEEE, 2017, pp. 229–236.

[7] H. Kagermann, J. Helbig, A. Hellinger, and W. Wahlster, *Umsetzungsempfehlungen für das Zukunftsprojekt Industrie 4.0: Deutschlands Zukunft als Produktionsstandort sichern; Abschlussbericht des Arbeitskreises Industrie 4.0*. Forschungsunion, 2013.

[8] J. Lee, B. Bagheri, and H.-A. Kao, "A cyber-physical systems architecture for industry 4.0-based manufacturing systems," *Manufacturing Letters*, vol. 3, pp. 18–23, 2015.

[9] J. Shi, J. Wan, H. Yan, and H. Suo, "A survey of cyber-physical systems," in *Wireless Communications and Signal Processing (WCSP), 2011 International Conference on*. IEEE, 2011, pp. 1–6.

[10] K. Sakurama, "Control of large-scale cyber-physical systems with agents having various dynamics," *IEEE Transactions on Big Data*, 2017.

[11] A. O. Kilyen and T. S. Letia, "Platform for unified enhanced time petri net models," in *Intelligent Computer Communication and Processing (ICCP), 2017 13th IEEE International Conference on*. IEEE, 2017, pp. 223–230.

[12] T. Semwal, S. S. Jha, and S. B. Nair, "On ordering multi-robot task executions within a cyber physical system," *ACM Transactions on Autonomous and Adaptive Systems (TAAS)*, vol. 12, no. 4, p. 20, 2017.

[13] P. Novak, P. Kadera, and M. Wimmer, "Agent-based modeling and simulation of hybrid cyber-physical systems," in *Cybernetics (CYBCONF), 2017 3rd IEEE International Conference on*. IEEE, 2017, pp. 1–8.

[14] D. IEC, "61512 (2000) chargenorientierte fahrweise."

[15] M. Adams, W. Kühn, T. Stör, and M. Zelm, "Din en 62264," *Die neue Norm zur Interoperabilität von Produktion und Unternehmensführung–Teil*, vol. 1, pp. 52–57, 2007.

[16] D. Brandl, "What is isa-95? industrial best practices of manufacturing information technologies with isa-95 models," *Acedido a*, vol. 15, 2008.

[17] STANDARD, "Enterprise-control system integration part 1: Models and terminology (iec 62264-1:2013)," 2013.

[18] D. He, A. Lobov, and J. M. Lastra, "Isa-95 tool for enterprise modeling," in *ICONS: The Seventh International Conference on Systems, Saint Gilles, Reunion, France*, 2012, pp. 83–87.

[19] I. R. Wischnewski, D. Initiative zur rechnerintegrierten Fertigung RIF eV, and I. J. Roßmann, "3-d modelling and simulation of tool change systems for the virtual production," in *Proceedings of the 20th IASTED International Conference*, vol. 670, no. 077, p. 87.

[20] D. Losch and J. Roßmann, "Visual programming and development of manufacturing processes based on hierarchical petri nets," in *Soft Computing & Machine Intelligence (ISCMI), 2016 3rd International Conference on*. IEEE, 2016, pp. 154–158.

[21] J. Rossmann, M. Dimartino, M. Priggemeyer, and R. Waspe, "Practical applications of simulation-based control," in *Advanced Intelligent Mechatronics (AIM), 2016 IEEE International Conference on*. IEEE, 2016, pp. 1376–1381.

[22] D. Hildebrand, "An architectural overview of qnx." in *USENIX Workshop on Microkernels and Other Kernel Architectures*, 1992, pp. 113–126.

[23] C. Y. Baldwin and K. B. Clark, *Design rules: The power of modularity*. MIT press, 2000, vol. 1.

[24] J. Rossmann, C. Schlette, and M. Springer, "Kinematic robot control for a planetary landing mockup," in *Proceedings of the Twentieth IASTED International Conference on Applied Simulation and Modelling ASM (accepted)*, 2012.