

# Virtual Reality Commissioning in Production Systems Preparation

M Dahl<sup>1,2</sup>, A Albo<sup>1</sup>, J Eriksson<sup>1</sup>, J Pettersson<sup>1</sup>, P Falkman<sup>1</sup>

**Abstract**— Virtual commissioning (VC) is a method used to virtually visualize and test production systems, control logic and material flows. The focus of this paper is to further extend this concept using Virtual Reality (VR). The introduction of VR in VC enhances the concept by adding a more realistic visualization and movement tracking which extends the possibilities of its validation. The changes to the validation aspect are mainly due to the fact that it is now possible to interact with the running virtual production in a realistic and intuitive way. The interaction gives designers and operators a new possibility to go from being observers to actors in the design phase. They are now able to validate the production system, test security protocols and validate the human interaction with the system, using VR.

**Index Terms**— virtual commissioning, virtual reality, automation, digital manufacturing

## I. INTRODUCTION

Virtual commissioning (VC) is a concept concerning the visualization, programming and validation of a production system in a virtual environment, as a preparational step for the construction of new manufacturing plants or changes to an existing one [1], [2].

Using VC for validation has been shown to be beneficial for both broad focus (material flow) and narrow focus (PLC-code) [3], [4] and can also save a significant amount of time and money [5]. These savings are achieved since errors in the planned production cell are discovered in an earlier stage of the development process [6], [7]. A system where VC has been performed may, therefore, have a higher software quality [8].

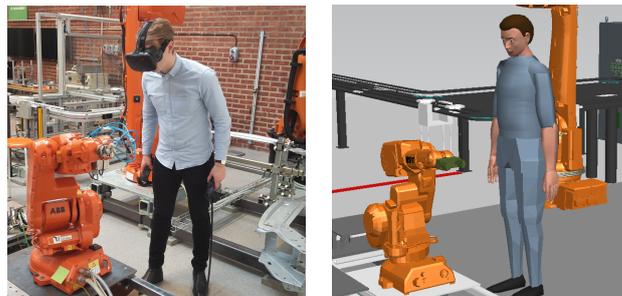
Another advantage of performing VC is that it enables a safe way of testing the integration of new technology and software with the (virtual) production system, since there is no risk of physical destruction in the virtual model and there exists an unlimited number of prototype parts in it [9], [10].

While the industrial usage of VC is increasing [11], another field that has seen rapid growth in interest lately is Virtual Reality (VR), largely due to its reduced cost [12].

\*This work has been carried out at the Wingquist Laboratory VINN Excellence Centre within the Production Area of Advance at Chalmers. It has been supported by VIRTCOM-Virtual preparation and commissioning of production systems including PLC logics, reference number 2014-01408, Vinnova, FFI within Sustainable production technology.

<sup>1</sup>Martin Dahl, Anton Albo, Johan Eriksson, Julius Pettersson and Petter Falkman are with the Department of Signals and Systems, Chalmers University of Technology, 412 96 Göteborg, Sweden

<sup>2</sup>Corresponding author: martin.dahl@chalmers.se



(a) The user wearing a HMD, inspecting a robot in a real production cell. (b) The virtual human object corresponding to the user.

Fig. 1: Comparison between the real and the virtual scenario where a user is validating a robot.

There is no clear definition of VR and researchers in the field tend to, according to [13], define VR in slightly different ways. The definition used in this paper signifies that VR is a technology used to provide the user with an experience of the real world in a virtual environment by using visual, audible, and haptic feedback.

In [12], it is suggested that investing in VR is valuable for manufacturing companies because of the gains in cost competitiveness. Automotive manufacturers have, for example, successfully used VR for evaluating the driver visibility of their vehicles as well as analyzing colors and ergonomics in an early development stage [14].

A natural extension to VC is, therefore, to be able to step in to a virtual model of a production system and experience it as if it already exists. This can be achieved by implementing VR into existing tool chains for VC, which enables a user to not only observe the production system but also actively interact with it. This provides additional opportunities for validation and user training:

- Safety equipment and related control logic can be evaluated and tested by interacting with virtual devices in the virtual world (e.g. breaking light curtains or opening safety gates).
- Operators can start training by interacting with a live production system before it is built (e.g. using a virtual HMI or inspecting virtual parts from natural viewing angles).
- Early evaluation and optimization of operator ergonomics.

This work describes a method where VR is added as an extension to existing VC technology in order to achieve a simulation environment where a user can

interact with the virtual production. Similar work on combining VR and production preparation include [15], [16], [17]. However, in these, the environment has been static and no interaction with a running simulation has been possible.

This paper is organized as follows: a brief background regarding VC and VR is given in Section II. In Section III, the experimental setup is described, detailing the virtual manufacturing model, the control system with a simulation unit device, and at last the HTC-Vive. Section IV describes the implementation needed in order to add VR support to the existing VC tool chains, followed by the results in Section V. Finally, concluding remarks are provided in Section VI.

## II. BACKGROUND

This section presents a brief overview of, and background to, VC and VR technology.

### A. Virtual commissioning

As described in [18], there exist four basic configurations in which development, testing, and integration of an automation system can occur. Traditionally, the physical production system has been tested and integrated using the real control system. Starting with a control system that is prepared to some degree, the system is integrated into the production system and tested. This activity may take weeks. But the real control system can also be coupled with a simulation model of the production system, which is usually called a hardware-in-the-loop (HIL) setup. This setup is what VC commonly refers to [2] and it is also how the term is used in this work. The inverse, reality-in-the-loop commissioning, would be the physical production system controlled by a simulated controller. This can have benefits for example when debugging a control system. The last combination of the two concerns the simulation model together with the simulated control system – this is called offline programming or constructive commissioning [2]. When designing a new control system, this is the natural starting point.

The main motivation for using VC, as defined in the previous paragraph, is to reduce the testing and integration time during the developmental phase. This is achieved by testing and integrating the control system before the physical production system is completely installed. The expectation is by using a simulation model of the production system, undesirable behavior can be detected well ahead of the physical installation. In fact, conducting a VC enables tests that would be prohibitively expensive or even impossible to run on a physical system. Additionally, having the simulation model makes it possible to test changes to the production system while it is running and being able to incorporate last minute changes without worrying about their impact on the system. [6], [19], [2]

Because VC, as defined in this work, uses the real control system, the simulation models need to be specified at

the level of sensors and actuators [2]. This is possible in today's VC enabled simulation software. To control the virtual environment with external logic, one uses either a soft-PLC (constructive commissioning) that operates virtually on the simulation computer, or an external physical PLC [20] that communicates through some kind of signal distribution terminal to the computer. Well established techniques that are used within the field are OPC DA, or OPC UA, which with a standardized protocol transmit signals between the different clients over Ethernet [21].

Current methodology applies VC as the last step in the automation engineering phase, to reduce the physical commissioning time of the control system. However, VC can provide value during the entire automation engineering process, as shown in [22], where a framework called *Integrated Virtual Preparation and Commissioning* is introduced. A formal model of the control logic, combined with having the same virtual models of the production system shared between the preparation, control system implementation, and VC phases, enable early validation of high level control as well as the possibility to perform early optimization based sequencing.

### B. Virtual reality

A head mounted display (HMD) is a device used for the visualization of a VR environment. This device contains a display which provides the user with the visual content as well as multiple sensors used to measure the user's head motions and feed these back into the visual content. A user that is wearing a HMD can be seen in Fig. 1a.

The use of VR is rapidly increasing as well as spreading to new areas. One explanation for this is the fact that the cost of VR-equipment has fallen drastically [12], which in turn has increased the interest for VR-technology in the industry. Companies in the automotive industry have for example used VR as a tool in the early stage of the development process to successfully investigate design parameters such as colors and sizes of different parts. It is also used for validating the ergonomics of the fitter who will be assembling vehicles [14]. Moreover, VR is used in areas like gaming, research, product development, manufacturing, operator training, and education [23], [24].

In [25], it is shown how VR can be used as an educational tool by describing how VR was successfully used to prepare surgeons for advanced operations without the risk of harming a real person. Another example of VR in the industrial environment is a project by Boeing, where HMDs were used to visualize parts of their airplane manufacturing in a static manner [15]. This setup is used as a platform for testing and showcasing new parts and products.

## III. EXPERIMENTAL SETUP

In this section the setup of the different components used in the experiment is described.

### A. Virtual manufacturing model

The production system used in the experiment is an existing robot cell within the Production System Laboratory (PSL) at Chalmers University of Technology, Sweden. The cell is made up of four ABB-robots, one KUKA-robot and a conveyor which transports pallets to and from the robots. A virtual model of the cell has been created in Process Simulate, a software provided by Siemens. See Figure 1 for an illustration of both the virtual model and the real robot cell.

The virtual model of the robot cell includes a human object in order to test sensors, buttons, and safety equipment such as emergency stops and light curtains. The human object is the entry point for extending VC with VR. This is done by using "Jack" v.8.4, another Siemens software which is used to run various tests and operations on the human objects that are present in Process Simulate [26]. Jack has support for development of external modules to extend the functionality of the program. The module called "Jack Collaboration" provides the necessary connections between Jack and Process Simulate in a server-client kind of relationship.

The virtual model of the robot cell contains all I/O:s needed to control it using the real control system, which make it possible to use the virtual commissioning facilities available in Process Simulate to control the system using an external PLC.

### B. Control system with a simulation unit device

The control system hardware used in the experimental setup is a Siemens SIMATIC PLC CPU 1517F-3 PN/DP with integrated safety together with the engineering software tool TIA Portal V13 SP1 containing both STEP 7 for programming and WinCC for HMI-applications.

Siemens has developed a communication solution in the shape of a simulation unit called SIMBA [27], which works as a hardware module that depicts a virtual equivalent of the nodes, that otherwise would have been connected to the PLC. In order to test the performance of the PLC-code without having physical equipment connected, the Simulation Unit PN128 as a hardware interface enables a real-time simulation of up to 128 profinet I/O-devices. This implies that the PLC can execute without any additional changes to the hardware configuration of the real plant PLC. Process Simulate then connects to the simulation unit as well to use the virtual hardware for controlling the virtual production system in real-time. Figure 2 shows the connection between the PLC, Process Simulate through the simulation unit with virtual I/O-nodes and robot modules.

With the software SIMULATIONUnit, it is possible to configure a simulation unit device to analyze and test basic functionality of the signal exchange between the PLC and Process Simulate. A successful connection makes it possible for the simulation and the execution of the PLC-code to occur simultaneously.

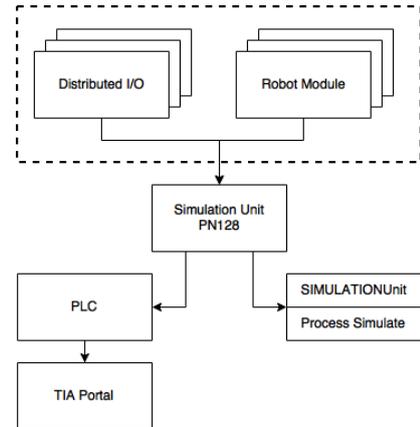


Fig. 2: Overview of the connection between the PLC and Process Simulate using a simulation unit device with virtual I/O-nodes and robot modules.

### C. The VR-equipment HTC-Vive

HTC-Vive is a consumer-grade VR-equipment that consists of one HMD and two hand held controllers. In the HMD, two different images are shown for the user, one for the left eye and one for the right eye. These two images are slightly shifted to each other and together they create a stereoscopic image. Beside the pure visualization, both the HMD and hand held controllers are equipped with sensors that can be used for 360° motion capture. By using these sensors in the HMD the image can be changed depending on the movement of the user's head.

A crucial part of the VR-system is tracking position in the room and the orientation of the HMD and the hand held controllers. This is achieved in the HTC-Vive system with the two base stations called lighthouses and the "on-board" sensors [28]. The lighthouses are two small cubes that emit light. Each lighthouse constantly performs a light scan both vertically and horizontally in the room. These two lighthouses do not receive any information themselves but instead allow the software to determine where and how the HMD and the hand held controllers are oriented in the room. Both the HMD and the hand held controllers have several light sensors and when a lighthouse emits light, the HTC-Vive measures the time it takes until the photo sensors are activated [28]. The HTC-Vive is then, after each flash from a lighthouse, able to calculate the positions and orientations based on when each sensor was activated.

## IV. THE DEVELOPED VR EXTENSION

The basic idea is to use the sensors in the HMD and the hand held controllers to make a human object in Process Simulate follow the movement of the user. At the same time, the images that the human object sees in the virtual production system are rendered to the HMD so that the user experiences the same simulation environment in 3D. The user should then be able to move in the virtual cell

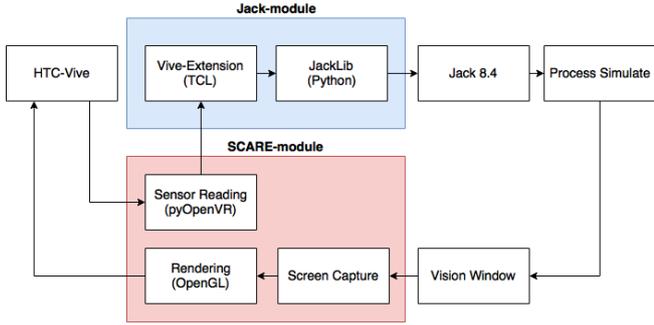


Fig. 3: The developed VR-system that interacts with the virtual simulation environment. The two highlighted parts (the Jack-module and the SCARE-module) are described in this section.

by walking and moving his or her hands in the real world. The user should also be able to trigger events by, for example, virtually pressing buttons and trigger sensors to actively interact with the virtual cell.

To extend the existing VC functionality with VR, the proposed VR-system consists of the HTC-Vive, Jack 8.4, Process Simulate and two modules developed as parts of this work: the SCARE-module (Sensor, CApture and REnDerinG) and the Jack-module. The interaction of the different modules is depicted in Fig. 3. The SCARE-module contains three Python programs. These programs handle sensor reading, rendering to the HTC-Vive, and capturing of the images that are going to be rendered in the HMD. The Jack-module handles the communication with Jack 8.4 and is divided into two parts, one for receiving sensor data from the SCARE module called Vive-Extension and one for controlling the human object called JackLib.

#### A. Control of the human object with a HTC-Vive

The core of the system is the developed Jack-module. This module is loaded into Jack 8.4 and contains all functionality for controlling the movement of the human object. It also handles the sensor values from the HMD and the hand held controllers.

JackLib contains methods for moving the hands and arms, rotating the head and rotating the torso along with other pure functionality methods like resetting orientations. To be able to map the movement of the user to the human object, sensor data from the HMD and controllers is needed.

Vive-Extension can be loaded into Jack 8.4 and can be seen as a shell for JackLib in order to control the human object's movement. By reacting to sensor data changes in the SCARE-module, this module triggers events that call the appropriate method in JackLib to move the human object. The sensor data is retrieved using the SDK "pyOpenVR" [29], which is python bindings for Valve's OpenVR virtual reality SDK.

The developed hand tracking in the SCARE-module reads position data provided by the hand controllers and

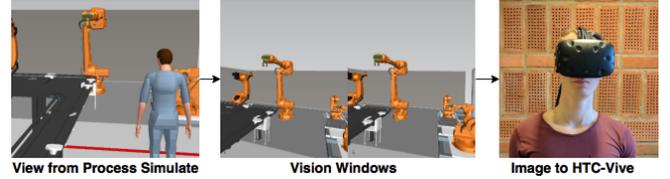


Fig. 4: The data flow when rendering the virtual environment to the HTC-Vive.

checks whether the change in position is larger than the chosen threshold. The threshold is used to provide a smooth hand tracking and to prevent the risk of flooding the communication system with too much data. Jack 8.4 has built-in functionality for making the human object reach towards a point in space while maintaining a natural motion of the arms and torso. By moving the desired goal points (one for each hand) to the updated hand positions, the human object in the virtual production model will move its hands to these new positions.

The user is able to navigate the human object in two different ways, either by pressing different buttons, corresponding to different directions, on one of the controllers, or by moving around in the real world inside the area which is predefined by the lighthouses belonging to the HTC-Vive system.

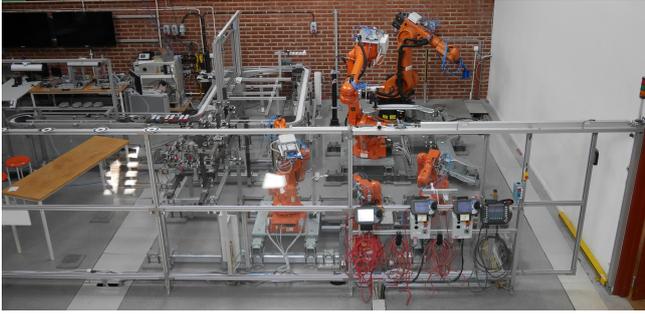
#### B. Visualization of a production cell with HTC-Vive

Two slightly shifted images are needed to render a 3D-image to the HMD. "Vision Window" in Process Simulate generates two images according to what the human object sees, one for the right eye and one for the left eye. The content of these two windows are then captured from the screen and then individually rendered to the HMD. The data flow can be seen in Fig. 4. This process is done continuously in the SCARE-module and is constantly updating the eye-view to show what the human object sees.

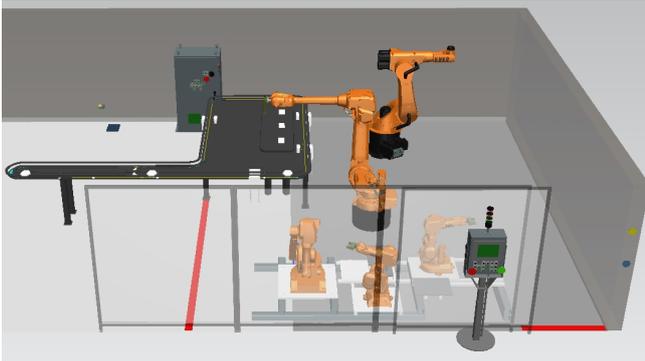
#### C. Interacting with the virtual cell

The interaction between the user and the virtual cell takes place in Process Simulate where the human object affects the running simulation by interfering with light and proximity sensors. This is done either by moving through a light sensor acting as a light curtain or by raising the hand close to a proximity sensor which acts as a button.

There are a total of 5 buttons and 2 light curtains implemented in the virtual model of the production system. They correspond to a HMI that starts, resets, and stops full auto production, as well as emergency stops and safety zones. For example, the safety zones and emergency stops are connected to a fail safe module in the PLC which triggers immediate stop of any moving devices.



(a) The real cell that was validated.



(b) The virtual model of the cell.

Fig. 5: Comparison between the real- and virtual production cell located at Chalmers University of Technology.

## V. RESULTS

The resulting VR experience runs simultaneously with the simulation of the production cell, controlled by the real control system. The user can move around, move both hands and rotate the head and torso. This is achieved by moving in the real world using the developed software described in Section IV in combination with new available VR technologies such as HTC-Vive together with widely used simulation softwares as Jack 8.4 and Process Simulate.

The environment used to validate the result is a virtual model of the cell in PSL at Chalmers University of Technology. An comparison between the virtual model and the real cell in PSL is shown in Figure 5. All objects in the PSL cell are not included in the virtual model, it is limited to the objects used in the test case.

One scenario that was used to validate the developed system is shown from the user perspective in Figure 6. When the simulation starts, the human object is located outside the cell. The user can then reach out and press a virtual start-button on the HMI in the simulation, which will start the virtual production. It is also possible to open the door to the cell by pressing a button next to it and then enter the cell. This action will immediately activate an emergency stop since the user violates the safety protocols of the running cell by crossing the red line where the light curtain is placed. The user is then allowed to move around freely in the cell, since the virtual

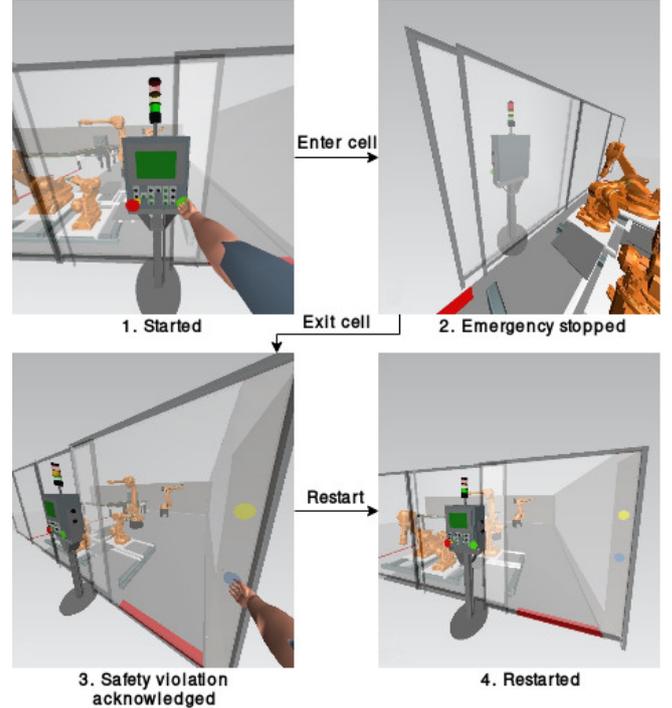


Fig. 6: One scenario used to validate the developed system, seen from the user's point of view.

production has been stopped, and can for example validate if all parts in the system are correctly placed. This is done by walking up to them and performing a visual inspection from different (ergonomically possible) angles. To resume the virtual production, it is required that the user navigates out of the cell and first acknowledges the safety violation by pressing the reset button at the door before pressing the start button again. The user can also stop the virtual production by pressing the emergency stop button beside the start button instead of activating the light curtain.

## VI. CONCLUSION AND FUTURE WORK

From the experiments on the setup described in this paper it is shown that it is possible to use VR to interact with a simulation environment while it is running. This extension to VC which opens up possibilities for operators, designers, and programmers to investigate and validate the environment and systems on another level. They can now take the step from being the observer to actually interact with a virtual production system.

Safety systems and PLC logic can be tested through interactions with the production system before the real physical system has been installed. This can reveal design problems in the development phase which otherwise might not have been noticed until the real system would be up and running. These problems can, in other words, be solved in a safer and earlier stage of the process and thereby save both time and money.

When verifying a new production cell without virtual

preparation, two possible errors can occur when the safety is tested. One is that the physical wiring of the devices is incorrectly performed or that the logic in the PLC is not working properly. In virtual preparation with VR, logic can be tested interactively in real time, thus reducing the time of troubleshooting during physical commissioning. For example, if the safety implementation in the real cell do not behave according to the virtual testing of the logic, it is more likely due to error with the wiring rather than the control logic.

This interaction is not only useful for the engineers and designers who construct the production system, but can also be used for training, e.g. a maintenance technician can be trained to operate on the future system before it has been constructed. This saves time in the educational process and the leap from learning about how the system works in theory to working with the real system is reduced compared to a performing a traditional VC without VR. Additionally, this opens up for training on a global level since the virtual environment can be easily shared.

Possibilities to evaluate new hardware or software to an already existing cell can also be evaluated. For example can a new robot can be added to an existing cell and the behaviour and collaboration with other objects can be validated. This can be used both in design of production cells and training of operators.

The experimental setup is based on simulation software widely used in industry combined with new available VR technology with the extra addition of the developed software described in this paper. The interaction between these components makes it possible to apply VR technology to an existing simulation project in its current configuration.

In the described setup, only one person can interact in the simulation at once. Future work could be to implement functionality for allowing multiple users to be in the same simulation simultaneously, enabling training for complex systems that require multiple operators. The simulation could also be improved by adding virtual human machine interfaces which could provide the user with readable information about the virtual production. The virtual experience could be further enhanced by providing the user with audible and haptic feedback.

One possible improvement regarding the interaction with the simulation environment could involve the implementation of interaction with the actual physical objects and not only sensors. A user could in that case navigate in the production system and move products around, e.g. if products have been placed in a wrong way. The user could then also lift up and inspect products in a more detailed way.

#### ACKNOWLEDGMENT

The authors thank the Master's students John Weman, Erik Sedelius and Hampus Lindvall at Chalmers

University of Technology for their contributions to the virtual model used in this paper.

#### REFERENCES

- [1] P. Hoffmann, R. Schumann, T. M. Maksoud, and G. C. Premier, "Virtual commissioning of manufacturing systems a review and new approaches for simplification." in *ECMS*, 2010, pp. 175–181.
- [2] C. G. Lee and S. C. Park, "Survey on the virtual commissioning of manufacturing systems," *J. Comput. Des. Eng.*, vol. 1, no. 3, pp. 213–222, 2014.
- [3] W. Hofmann, S. Langer, S. Lang, and T. Reggelin, "Integrating virtual commissioning based on high level emulation into logistics education," *Procedia Engineering*, vol. 178, pp. 24–32, 2017.
- [4] S. C. Park, M. Ko, and M. Chang, "A reverse engineering approach to generate a virtual plant model for PLC simulation," *Int. J. Adv. Manuf. Technol.*, vol. 69, no. 9-12, pp. 2459–2469, 2013.
- [5] N. Shahim and C. Møller, "Economic justification of virtual commissioning in automation industry," in *Winter Simulation Conference (WSC)*, 2016. IEEE, 2016, pp. 2430–2441.
- [6] P. Hoffmann, T. Maksoud, R. Schumann, and G. Premier, "Virtual Commissioning of Manufacturing Systems a Review and New Approaches for Simplification," *Proc. 24th Eur. Conf. Model. Simul.*, vol. 2, no. Cd, 2010.
- [7] M. Oppelt and L. Urbas, "Integrated virtual commissioning an essential activity in the automation engineering process: From virtual commissioning to simulation supported engineering," in *Industrial Electronics Society, IECON 2014-40th Annual Conference of the IEEE*. IEEE, 2014, pp. 2564–2570.
- [8] G. Reinhart and G. Wunsch, "Economic application of virtual commissioning to mechatronic production systems," *Production engineering*, vol. 1, no. 4, pp. 371–379, 2007.
- [9] R. Drath, P. Weber, and N. Mauser, "An evolutionary approach for the industrial introduction of virtual commissioning," in *Emerging Technologies and Factory Automation, 2008. ETFA 2008. IEEE International Conference on*. IEEE, 2008, pp. 5–8.
- [10] J. Bathelt and J. Meile, "Computer aided methods supporting concurrent engineering when designing mechatronic systems controlled by a plc," in *Proceedings of International Conference on Manufacturing Automation (ICMA07)*, Singapore, 2007.
- [11] S. Süß, S. Magnus, M. Thron, H. Zipper, U. Odefey, V. Fäßler, A. Strahilov, A. Kłodowski, T. Bär, and C. Diedrich, "Test methodology for virtual commissioning based on behaviour simulation of production systems," in *Emerging Technologies and Factory Automation (ETFA), 2016 IEEE 21st International Conference on*. IEEE, 2016, pp. 1–9.
- [12] S. Choi, K. Jung, and S. Do Noh, "Virtual reality applications in manufacturing industries: Past research, present findings, and future directions," *Concurrent Engineering*, vol. 23, no. 1, p. 56, 2015.
- [13] W. R. Sherman and A. B. Craig, *Understanding virtual reality: Interface, application, and design*. Elsevier, 2002.
- [14] L. P. Berg and J. M. Vance, "Industry use of virtual reality in product design and manufacturing: a survey," *Virtual Reality*, pp. 1–17, 2016.
- [15] Boeing, "Immersed in engineering," [Online], 2016, <http://www.boeing.com/features/2016/03/immersed-in-engineering-03-16.page>, Downloaded: 2017-03-29.
- [16] E. T. Corseuil, A. B. Raposo, R. J. da Silva, M. H. Pinto, G. N. Wagner, and M. Gattass, "Environ-visualization of cad models in a virtual reality environment." in *EGVE*, 2004, pp. 79–82.
- [17] F. Doil, W. Schreiber, T. Alt, and C. Patron, "Augmented reality for manufacturing planning," in *Proceedings of the workshop on Virtual environments 2003*. ACM, 2003, pp. 71–76.

- [18] F. Auinger, M. Vorderwinkler, and G. Buchtela, "Interface driven domain-independent modeling architecture for "soft-commissioning" and "reality in the loop"," in *Proc. 31st Conf. Winter Simul. Simulation—a Bridg. to Futur. 1.* ACM, 1999, pp. 798–805.
- [19] S. C. Park and M. Chang, "Hardware-in-the-loop simulation for a production system," *Int. J. Prod. Res.*, vol. 50, no. 8, pp. 2321–2330, 2012.
- [20] M. Schleipen, O. Sauer, N. Friess, L. Braun, and K. Shakerian, "Production monitoring and control systems within the digital factory," in *Proceedings of the 6th CIRP-Sponsored International Conference on Digital Enterprise Technology.* Springer, 2010, pp. 711–724.
- [21] M. H. Schwarz and J. Borcsok, "A survey on OPC and OPC-UA: About the standard, developments and investigations," in *Information, Commun. Autom. Technol. (ICAT), 2013 XXIV Int. Symp.* IEEE, 2013, pp. 1–6.
- [22] M. Dahl, K. Bengtsson, P. Bergagård, M. Fabian, and P. Falkman, "Integrated virtual preparation and commissioning: supporting formal methods during automation systems development," *IFAC-PapersOnLine*, vol. 49, no. 12, pp. 1939–1944, 2016.
- [23] M. Mihelj, D. Novak, and S. Beguš, *Virtual reality technology and applications.* Springer, 2014.
- [24] S. Pedram, P. Perez, S. Palmisano, and M. Farrelly, "A systematic approach to evaluate the role of virtual reality as a safety training tool in the context of the mining industry," 2016.
- [25] R. Riener and M. Harders, "Vr for medical training," in *Virtual Reality in Medicine.* Springer, 2012, pp. 181–210.
- [26] Siemens Industry Software, "Jack," [Online], 2011, [https://www.plm.automation.siemens.com/en.us/products/tecnomatix/manufacturing-simulation/human-ergonomics/jack.shtml#lightview%26url=/en.us/Images/4917\\_tcm1023-4952.pdf%26title=Jack%26description=Jack%20Fact%20Sheet%26docType=pdf](https://www.plm.automation.siemens.com/en.us/products/tecnomatix/manufacturing-simulation/human-ergonomics/jack.shtml#lightview%26url=/en.us/Images/4917_tcm1023-4952.pdf%26title=Jack%26description=Jack%20Fact%20Sheet%26docType=pdf), Downloaded: 2017-03-29.
- [27] S. Preuße, *Technologies for Engineering Manufacturing Systems Control in Closed Loop.* Logos Verlag Berlin GmbH, 2013, vol. 10.
- [28] T. Hilfert and M. König, "Low-cost virtual reality environment for engineering and construction," *Visualization in Engineering*, vol. 4, no. 1, p. 2, 2016.
- [29] C. Bruns, "pyopenvr," GitHub, 2017, <https://github.com/cmbruns/pyopenvr>, fbceec431140873d621eff60953c42a830e9c464.