# Virtual commissioning of a robotic cell: an educational case study

Igor Azkarate Fernández
*Dept. of Electronics and Computing*
*Mondragon Unibertsitatea*
Arrasate-Mondragón, Spain
iazkarate@mondragon.edu

Mikel Ayani Eguía
*School of Engineering Science*
*Högskolan i Skövde*
Skövde, Sweden
mikel.ayani@his.se

Luka Eciolaza Echeverría
*Dept. of Electronics and Computing*
*Mondragon Unibertsitatea*
Arrasate-Mondragón, Spain
leciolaza@mondragon.edu

***Abstract*—The emergence of software tools for testing control programs and virtual commissioning (VC) in industrial automation projects makes it possible to shorten lead times and improve product quality, but it also brings to light the need for competent technicians in these technologies. The academic environment can support the education of future professionals by reproducing and solving industrial problems in the classroom. This article presents a use case in which students work on a project to develop and validate the control system of a robotic cell. The study compares the conventional way of working against the use of a digital twin and exposes the benefits of it.**

***Index Terms*—Automation, digital twin, education, emulation, modelling, virtual commissioning**

## I. Introduction

Machine manufacturers and system integrators are facing a higher demand to reduce the delivery time of their new systems or the reconditioning of old automated solutions [1]. In a context of greater complexity, the commissioning time of automated systems needs to be shortened, which includes the verification and the validation of the control software, one of the most critical steps in a project.

The growth of the computational capacity of computer equipment and the irruption of tools for emulating machinery and processes, place us in front of a new paradigm. Thus, the software is tested before the real commissioning, which is called virtual commissioning (VC) [2]. But there is a lack of skilled professionals in these technologies [3].

The academic environment is one of the contexts in which future professionals develop. The use of modelling tools in this scenario allows a more effective management of the resources, avoiding experimenting with expensive real systems and reducing the occupancy rate of physical labs, with the consequent greater availability. The testing or debugging of the Programmable Logic Controller (PLC) program in its development environment requires a good knowledge of the process by the technician, as well as the reproduction of its behavior, assigning appropriate values to variables and following the desired sequences. A virtual model allows a better understanding of the system or process and reproduces the corresponding signals. Testing the control software against it makes possible to arrive at the commissioning of the installation with a higher quality program.

## II. Virtual Commissioning

Traditionally, the control software is validated at the end of the project, once equipment is already assembled. The emulation allows to test it without the need of this equipment, using a virtualization of the physical system.

### *A. Conventional Commissioning*

PLC programs are tested, in the first instance, in the control software development tools themselves, without direct vision of the process behavior. Then, with the system already assembled and connected, the commissioning is carried out, either at the manufacturer's or at the end customer's facilities. The main weaknesses of this procedure are:

- The start-up time can be approximately 25% of the total duration of the project [4]. In addition, setbacks such as unforeseen and unnecessary expenses, damage or injury can occur. This can affect the reputation of the manufacturing company and condition future orders.
- The time of machinery stopped, in case of reconditioning projects, or the delivery time, for complete developments, are key factors for the manufacturer's positioning in the market. Conventional commissioning does not guarantee short lead times.
- The debugging of control programs in the PLC development environment itself, without a proper testing procedure and without knowledge of the process, can cause the software to reach the start-up phase with insufficient maturation, which influences the described problem.

### *B. Use of Emulation*

The emergence of VC tools makes possible to test control programs for new projects, to modify existing installations, or to retrofit old machinery, without requiring the real equipment. Thus, software validation can be carried out before the commissioning phase, using a virtual model of the operational part, with various advantages, according to recent reports:

- Shorter duration of tests and validation during the development of an automation project.
- Detection of anomalous behavior before actual commissioning.

978-1-7281-0303-7/19/$31.00 ©2019 IEEE

- Performing tests with PLCs, robots and machine vision systems without causing downtime, or tests that would be costly or complex [5].

An useful example of this is presented by [6], with an industrial application study in which a reconditioning project is supported by emulation, with an important reduction of the commissioning time.

### C. An emulation tool: Simumatik3D

In recent years, numerous tools for emulating industrial systems and processes have emerged on the market. These tools model different aspects of the real systems. It is mandatory to analyze the use and expected outcome of the model in order to select the proper emulation tool. Siemens SIMIT Simulation Platform i.e. is a tool that can suit for virtual commissioning of process applications, such foundries or water treatment plants. It does not model the geometry and physics of the system, but in the other hand, it provides an easy-to-use behavior modeling user interface and good connectivity for Siemens PLCs. Other tools such as ABB Robot Studio, Excelgo Xperior, Simumatik3D or NX Mechatronic Concept Designer [7] provide tools to model the geometry and physics of the system. These tools are therefore more suitable for robot cell and material handling applications, among others. Tecnomatix Process Simulate is a tool in between the previous ones that provides the posibility of modelling the geomtery and behavior of systems but does not include a physics engine [8]. For this case, Simumatik3D® v1.0.3 (S3D) has been used, which has a free license for educational use and is very intuitive.

S3D is an emulation tool developed by Simumatik, which started in 2014. It enables users to test PLC and industrial robot programs in an easy way. Nowadays, the software is mainly used in several automation courses and research projects by the University of Skövde (HiS) in Sweden and Mondragon Unibertsitatea (MU) in Spain, and even other universities over the world.

With S3D it is possible to emulate different aspects of mechatronic systems: geometry, physics, kinematics and its behavior [1]. The emulation models are built introducing components, such sensors, motors or cylinders, and connecting them to each other in order to create a system. S3D integrates an OPC Unified Architecture (OPC UA) client to easily connect the model to the PLC or robot controller. OPC UA has become the standard for the communication between industrial control systems and 3rd party software, like emulation software, providing a transparent way to read and write controller variables. Most of the last generation PLCs include an OPC UA server that can be enabled and configured to allow access to some internal variables. For those controllers that do not have it, external software like Simumatik3D® OPC UA Server v0.1.6 or commercial OPCUA servers (Matricon, UAExpert or Kepware) can be used. Once variables are created in the industrial controller and made available through the OPC UA server, in order to setup the communication, the model in S3D just requires to setup the IP address of the controller and define the variables that want to be exchanged using the same name. During the emulation, S3D will read the output variables in the controller, i.e. relay or valve solenoid outputs, and write the value of input variables, such as sensors.

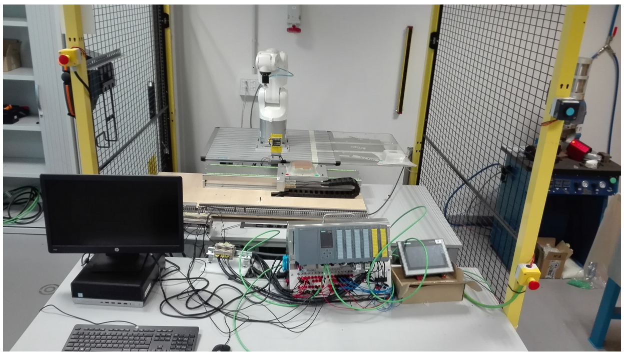

Fig. 1: Linear table and KUKA Agilent robot.

### D. Configurations for emulation

Testing and validation of control programs can be performed using one of these two configurations [9]:

- Virtual model of the process, and an emulated controller, or software-in-the-loop (SIL): consists of a combination of a model of the system to be controlled and an emulated PLC. This is a fairly inexpensive alternative, but because real equipment is not used, safety aspects, such as possible robot collisions or personal injuries can be overlooked.
- Virtual model of the process, and real controller, or hardware-in-the-loop (HIL): the real PLC hardware is used and even some auxiliary modules and field bus devices. An identical or very similar environment to the installation is created, and the software is tested directly on the controller. But this configuration means a higher cost, and less flexibility to changes during the design phase. In addition, the risk of accident is greater.

## III. Use Case: the robotic cell

This case study made use of a real robotic cell placed in a laboratory of the university, and its digital twin or emulation model created specifically for the experiment. The robotic cell acts as a workstation for identification, quality control and classification of drone housings, as shown in fig. 1.

### A. Cell

As shown in fig. 2, the process to be automated consists of:

- A table with a KUKA Agilent robot (it could be a gantry with X, Y and Z coordinate control without orientation control) in order to carry out the necessary manipulations in the process.
- A linear table (controlled by a servo-variator) in which the drone casing is moved, with four positions:
  - Manual input.
  - Height measurement, which depending on the measuring range is defined as good or bad.
  - Chromatic sensor to identify the color of the piece.

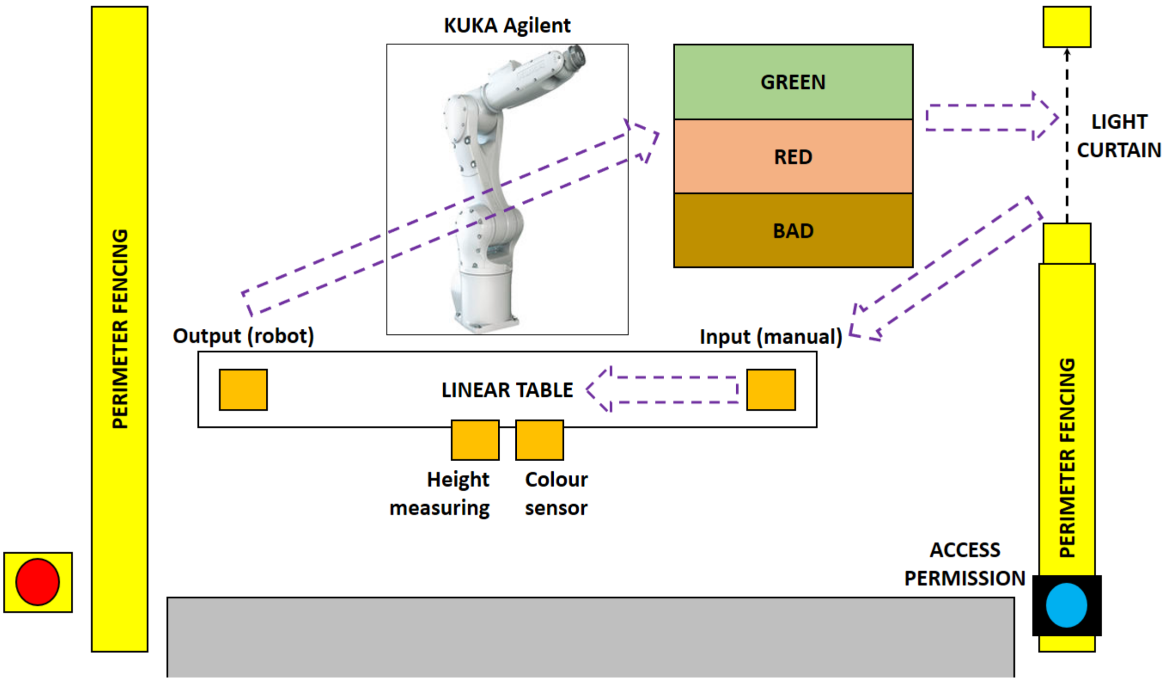


Fig. 2: Layout of the process.

  - Output, where the robot takes the part to the corresponding area.
- A security system consisting of perimeter fencing and an access area protected by a light curtain. This way, the classified material can be safely collected from the corresponding baskets.

### B. Virtual model

A virtual model of the robotic cell was developed in order to have the possibility of carrying out a virtual commissioning of the process. The modeling was carried out by the teaching staff, as the objective of the academic activity was the development of competences related to control software programming and the commissioning of automated machinery.

The model, which is shown in fig. 3, includes the linear table, including the servo-variator, the sensors and the robot. All sensors and actuators are connected to the Siemens PLC I/O signals using OPC UA. The virtual robot is connected to the robot controller and reproduces the movements programmed on it.

The signals used in the digital twin are shown in the table I. Note that there is correspondence between variables of both controllers. In order to coordinate the tasks to be performed by them, these signals are connected to each other in the virtual model, in the same way that is usually done with sensors to PLC inputs and outputs to actuators.

For the connection of signals of the controllers with the digital twin by means of OPC UA, two cases are differentiated. The PLC includes a server, so it must be enabled in the CPU. In addition, the automatic update of the input memory is disabled, so that the virtual model can write on it. Finally, it is required to indicate the IP address of the server in the parameters of the PLC block in S3D.

As the robot does not incorporate a server, the Simumatik3D® OPC UA Server v0.1.6 software is used. After adding the device to it, its type is chosen from a drop-down list, and the variables that will be used in S3D are added.

For both devices, the variable names must match in the controller and in the virtual model. They must be identical.

### C. Application

The work to be done consisted of programming the PLC based on specifications, so that it worked in coordination with the robot, whose program had been previously developed in another academic activity unrelated to this project. For the interaction between user and control devices, a Supervisory Control and Data Acquisition (SCADA) application was executed on a PC, developed with Wonderware InTouch software, and a SIMATIC HMI TP700 panel was also used. Both interfaces were previously operational, too.

TABLE I: Summary of controller signals in the virtual model.

| Robot output signals | | PLC input signals |
|---|---|---|
| oLock | $\Longrightarrow$ | iRobotLock |
| oReady | $\Longrightarrow$ | iRobotReady |
| **Robot input signals** | | **PLC output signals** |
| iPick | $\Longleftarrow$ | oRobotPick |
| iHeightOK | $\Longleftarrow$ | oHeightOK |
| iRedColourOK | $\Longleftarrow$ | oRedColourOK |

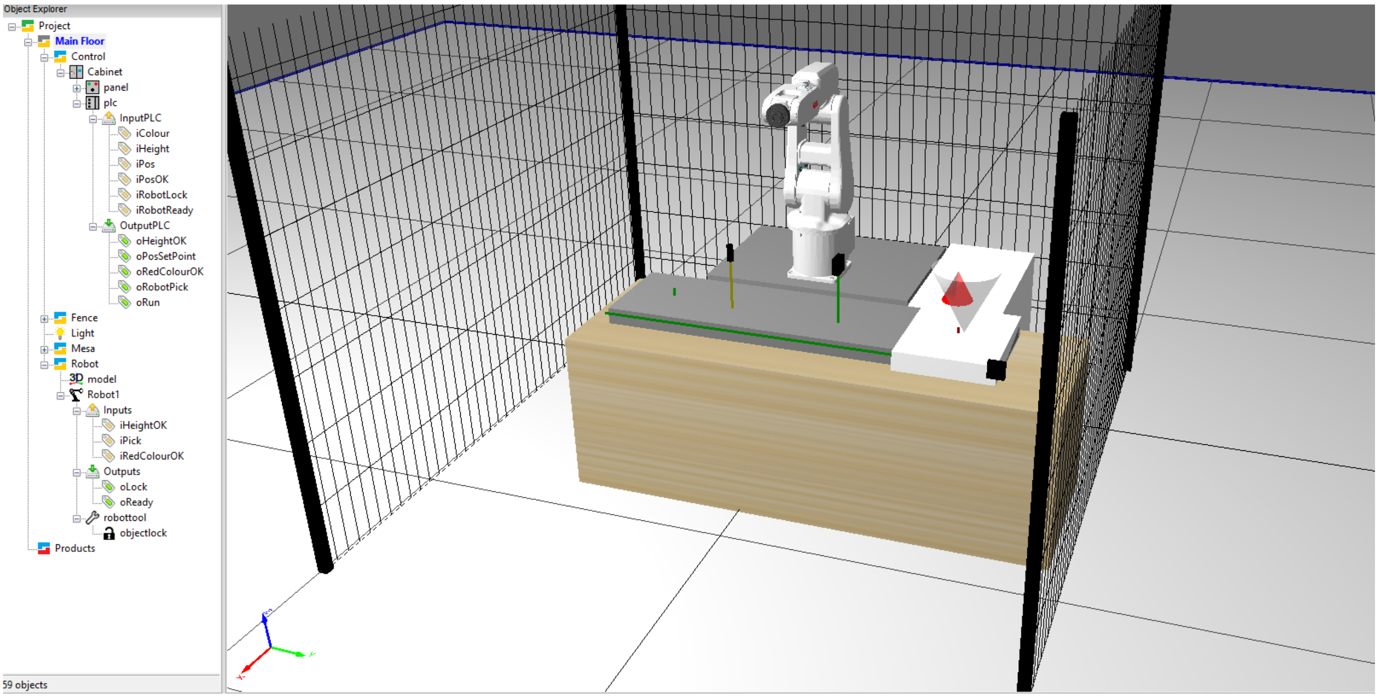


Fig. 3: Virtual model and signals in S3D.

The PLC to be programmed was a Siemens 1516F-3 PN/DP. For the HIL configuration, the physical device was used. For the SIL one, it was emulated by PLCSIM Advanced.

The work sequence is launched when the access barrier is unlocked by means of the access permission button. Then the operator can access the enclosure to deposit a drone casing at the entry point. The control continuously provides the appropriate signals for the table to move the casing from one position to the next, carrying out the checking points already described. The cycle is finished when the casing is evacuated by the robot, and the trolley returns to the initial position.

The control also carries out a count of all the processed housings, as well as, in particular, of the defects, and the timing of stop situations, with all three cases having the option of reset.

In order to respond to the proposed operation, the basic structure of the PLC program may be as follows:

- Startup: reset of outputs and internal variables, and setting of the servo-variator in run mode.
- Scan cycle: execution of program blocks.
  - The sequence already described, which can be useful to learn how to apply GRAFCET diagrams, and program them in Graph language.
  - If the housing is in the measuring position, calculation of the measured height from the associated analogic signal. Comparison and determination of whether or not the height is in the appropriate range.
  - Calculation of the analogic position setpoint signal to the servo-variator, from what has been determined in the main sequence: 0%, 40%, 60% or 100%.
  - Other functionalities.

For programming the control system, students from the Master's Degree in Industrial Engineering at MU were divided into groups. In the context of a six-monthly multidisciplinary project, aspects of different subjects were worked on, such as the design and manufacture of a prototype of a dron casing, the control of the linear table of the robotic cell, and, in this case, the implementation of the control system. Every group worked according to some specific conditions, as described in IV, but were asked to fulfill the same functional description for the development of the robotic cell.

## IV. METHODOLOGY

While the research to date has tended to focus on projects taking place in industry rather than the clasroom, this project was developed with a group of students, as introduced in III-C.

### A. Context

The case study covers a project in which students programmed and validated the control system of a robotic cell getting advantage of an emulation model. The study was carried out under different working conditions to analyze the benefits of virtual commissioning.

There is just one physical unit of the robotic cell in the facilities of the university, and consequently, the following limitations were detected in previous academic years:

- Little understanding of the process to be automated, and of the desired operation.
- PLC debugging does not guarantee the development of good control programs, as it is not a very intuitive tool for students to reproduce all possible scenarios.

- Limited use of equipment for each group, based on a calendar with work shifts.
- Commissioning of considerable duration, as a consequence of being based on programmes with many errors.

This academic year, a virtual model of the cell, developed in S3D, was added into the material available for students.

### B. Working conditions

The students, working in project groups of three members, were divided into different study groups:

a) Conventional approach (3 project groups): working with a real PLC and sharing the use of the cell, in shifts, with a schedule calendar that covered the whole period. The classic procedure was followed: development of the control software, testing it in the programming environment itself, and real commissioning.
b) Use of emulation model with HIL (4 project groups): a real PLC and a virtual model of the cell, with a HIL configuration, as shown in fig. 4. The cell was available punctually, at the end of the period, exclusively for real commissioning. An innovative procedure was followed, continuously testing the software, as it was developed, against the S3D model.
c) Use of emulation model with SIL (6 project groups): a PLC emulated in PLCSIM Advanced, and a virtual model of the process, with a SIL configuration. Exclusively virtual commissioning.

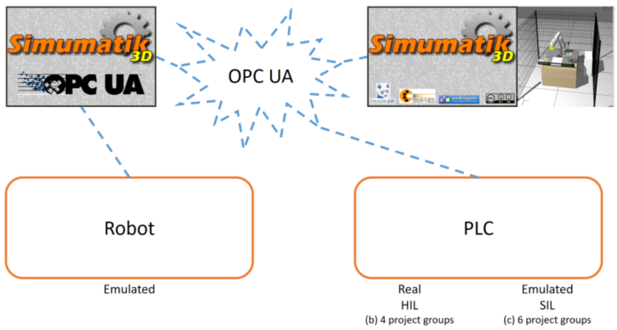

Fig. 4: Working conditions for type b and c project groups.

### C. Project closing questionnaire

Each project group worked under the conditions assigned to it, as described in IV-B. This led to the real commisioning of the automated solution, for those groups for which the completion of that phase was required. It was difficult to quantify the quality of the program and the time needed to put it into service for each group, as they were made up of people with different profiles and different interests in the subjects involved in the project. Consequently, it was decided to incorporate, to the closing activities, a qualitative questionnaire to the students, similar to what we would know as focus group. Participants were questioned about the use they had made or would made of virtual models for software validation in automation projects. From there, results were extracted, to be analysed and debated.

## V. Results

The table II shows the results of the questionnaire which the students have been submitted at the end of the project, about the experience (groups of type b and c) or possibility (groups of type a) of using a digital twin, since each group has experienced a work dynamic and has observed the others.

TABLE II: Summary of results.

| |
|---|
| a) Conventional approach. Only real commissioning. |
| Comments on emulation for software validation. |
| - Long commissioning time due to program errors. Detected a gap between controller debugging and real commissioning. Would use emulation in future projects.<br>- Difficulties in understanding the process to be controlled.<br>- High and inefficient use of resources. |
| Contributions on the use of virtual models in other phases. |
| - It can be useful for the detection and correction of design defects. |
| About skills development. |
| - It can give students a good position in the job market. |
| b) Emulation model with HIL. Both VC and real commissioning.<br>c) Emulation model with SIL. Only VC. |
| Comments on emulation for software validation. |
| - The real process is known without having it available, through a realistic virtual model of it.<br>- Possibility of developing and testing the software previously, in parallel with other tasks, against this model.<br>- All modes of operation can be tested in a realistic and intuitive way. In the PLC debugging environment a lot of abstraction is required, and what is tested seems to be fine without being so.<br>- Programs with fewer errors and in less time. Easy and early detection of errors and risk situations. (Contribution made by type b groups).<br>- The real start-up becomes a mere procedure. Reduction of downtime (reconditioning) or delivery times (complete project), and associated costs. (Contribution made by type b groups). |
| Contributions on the use of virtual models in other phases. |
| - Useful from the design phase.<br>- Testing of modifications and improvements in installations.<br>- Training of plant personnel and maintenance technicians. |
| About skills development. |
| - Learning of computer tools and ways to proceed with incipient implantation in industry.<br>- Reinforcement of concepts of programming and industrial communications. |

In the groups in which the classic procedure has been followed (a), the problems of previous years, previously described, are highlighted. It is observed that emulation can be a useful tool to improve in several aspects.

Those students who have used virtual models note the virtues of these. The ideas that have emerged in the questionnaire have been of great similarity both having carried out the final implementation (b) and not (c). They also see the applicability of the emulation tools in other phases, beyond the validation of the software.

In cases where commissioning has been carried out, the time reduction has been checked. This has provided a positive response to the problems that had arisen in previous academic years, as well as in some groups (those of type a). Consequently, type b groups have found softened, thanks to emulation support, aspects relating to:

- The understanding of the process and required performance is better.

- The PLC program is tested with greater rigor, easily reproducing all the scenarios that may occur.
- More efficient use of real equipment, after virtual validation of the developed software.
- Shorter commissioning time.

Corroborated by the teaching and research staff involved in this project, it has been proven that, for those programs validated in digital twin, cell commissioning has required less effort and time.

In summary, it is observed that students, whether they have used emulation or not, have recognized the virtues of using these tools in the validation of control programs. The former would return to work with these tools, and the latter would begin to use them.

They also anticipate the usefulness of modeling in other phases of the system life cycle, which puts us in front of the concept of Integrated Virtual Commissioning (IVC) [10]: from design phase to operator training.

Finally, regarding the development of skilled professionals for the industry, the participants value the knowledge acquired in an area they consider useful and emerging in manufacturing.

## VI. Conclusions

The conclusions drawn from this experience can be listed as follows:

- In the search for a tool that would facilitate the validation of control software in automation projects developed in the academic field, and given the successful industrial use cases of emulation presented in the literature, the use of this technology has been introduced in a project with students.
- In the use case presented, a virtual model is used for the verification of the PLC control software developed by the students, and for the execution of the program of a robot, from another academic activity prior to the one that concerns us.
- The virtues of using emulation for the validation of PLC programs or virtual commissioning have been tested in an educational environment. Satisfactory results have been obtained in terms of reduction of commissioning time, understanding of the system to be automated and better management of material resources, among others.
- Consequently, more activities similar to those presented in this article will be designed, as well as to use emulation in other phases of the engineering lifecycle, not only in the validation of PLC programs. An example of this is the study carried out by [11] in which a multi-objective optimization application is presented, targeting cycle time and energy consumption of a robotic cell. Modeling in other phases can lead to more projects from which the corresponding results would be disseminated, if appropriate.
- It is difficult to estimate cost of modeling. But this is an effort made once, and develops skills of the teaching staff, as well as a virtual model that can be reused. It is considered an investment for the upcoming transfer of knowledge to teachers, students and company personnel, and for the development of projects.

## References


[1] N. Shahim and C. Møller, "Economic justification of virtual commissioning in automation industry," in *Proceedings of the 2016 Winter Simulation Conference*, ser. WSC '16. Piscataway, NJ, USA: IEEE Press, 2016, pp. 2430–2441. [Online]. Available: http://dl.acm.org/citation.cfm?id=3042094.3042396

[2] P. Hoffmann, R. Schumann, T. Maksoud, and G. Premier, "Virtual commissioning of manufacturing systems - a review and new approaches for simplification," 06 2010, pp. 175–181.

[3] Z. Liu, C. Diedrich, and N. Suchold, *Virtual Commissioning of Automated Systems*. INTECH Open Access Publisher, 2012. [Online]. Available: https://books.google.es/books?id=YeTmoAEACAAJ

[4] G. Reinhart and G. Wünsch, "Economic application of virtual commissioning to mechatronic production systems," *Production Engineering*, vol. 1, no. 4, pp. 371–379, Dec 2007. [Online]. Available: https://doi.org/10.1007/s11740-007-0066-0

[5] M. Dahl, K. Bengtsson, M. Fabian, and P. Falkman, "Automatic modeling and simulation of robot program behavior in integrated virtual preparation and commissioning," *Procedia Manufacturing*, vol. 11, pp. 284 – 291, 2017, 27th International Conference on Flexible Automation and Intelligent Manufacturing, FAIM2017, 27-30 June 2017, Modena, Italy. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S2351978917303116

[6] M. Ayani, M. Ganebäck, and A. H.C. Ng, "Digital twin: Applying emulation for machine reconditioning," *Procedia CIRP*, vol. 72, pp. 243–248, 01 2018.

[7] K. Herbuś, P. Ociepka, and A. Gwiazda, "Virtual activating of a robotized production cell with use of the mechatronics concept designer module of the PLM siemens NX system," in *Advances in Intelligent Systems and Computing*. Springer International Publishing, aug 2018, pp. 417–425.

[8] L. V. Guerrero, V. V. López, and J. E. Mejía, "Virtual commissioning with process simulation (tecnomatix)," *Computer-Aided Design and Applications*, vol. 11, no. sup1, pp. S11–S19, 2014. [Online]. Available: https://doi.org/10.1080/16864360.2014.914400

[9] M. Schamp, S. Hoedt, A. Claeys, E. Aghezzaf, and J. Cottyn, "Impact of a virtual twin on commissioning time and quality." *IFAC-PapersOnLine*, vol. 51, no. 11, pp. 1047 – 1052, 2018, 16th IFAC Symposium on Information Control Problems in Manufacturing INCOM 2018. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S2405896318315970

[10] M. Oppelt and L. Urbas, "Integrated virtual commissioning an essential activity in the automation engineering process: From virtual commissioning to simulation supported engineering," *IECON 2014 - 40th Annual Conference of the IEEE Industrial Electronics Society*, pp. 2564–2570, 2014.

[11] M. Ayani, A. H. C. Ng, and M. Birtic, "Optimizing cycle time and energy efficiency of a robotic cell using an emulation model," in *Advances in Manufacturing Technology XXXII : Proceedings of the 16th International Conference on Manufacturing Research, incorporating the 33rd National Conference on Manufacturing Research, September 11–13, 2018, University of Skövde, Sweden*, ser. Advances in Transdisciplinary Engineering, vol. 8, no. 8, 2018, pp. 411–416.