Artificial Neural Network Applied to Virtual Commissioning and Control of a Robotic Cell

Gabriel Bastos de Miranda, and Fábio Lima, *Member, IEEE*

***Abstract*—The world is undergoing significant changes in information technologies and industrial processes. The rise of Industry 4.0 and the advancement of artificial intelligence are creating new opportunities and challenges for industries. This study investigates the feasibility of substituting a traditional programmable logic controller (PLC) with a neural network-based control system for discrete event management within a robotic cell. The research assesses the feasibility of this replacement, analyzing the associated challenges, limitations, and advantages compared to traditional methods. Digital manufacturing software is employed for simulating and validating the proposed model through a Virtual Comissioning (VC). The control system of the proposed model utilizes artificial neural networks, trained using data derived from a Boolean logic model. The results indicate that it is possible to swiftly train an artificial neural network (ANN) to take over cell control. This approach opens up the possibility of implementing low-cost hardware, aligning the system with the concepts of Industry 4.0. Additionally, the virtual modeling conducted using digital manufacturing software paves the way for a future implementation of a digital twin. Findings indicate that the neural network control approach is feasible and offers operational advantages over traditional programming methods.**

Link to graphical and video abstracts, and to code: https://latamt.ieeer9.org/index.php/transactions/article/view/9233

***Index Terms*—Machine Learning, Digital Manufacturing, Virtualization, Industry 4.0, Programmable logic controller.**

I. INTRODUCTION

SINCE the Industrial Revolution, manufacturing technology has been developing rapidly. It began with the introduction of machines to manual production, passing through Taylor's rationalization and Henry Ford's assembly line, and then reaching the third revolution, which was marked by the development of Information and Communication Technologies (ICT) [1]. However, Cyber-Physical Systems (CPS) and the Internet of Things (IoT) are technologies that have made great strides in the last decade, and with this development, a new concept was introduced in Germany in 2011, Industry 4.0, which symbolizes the beginning of the fourth industrial revolution [2], [3].

Currently, the manufacturing industry is experiencing an unprecedented increase in available data. This data arises in a variety of formats, such as from a sensor on a production line, environmental data, or tool parameters [4]. Consequently, the industrial world is undergoing profound changes with the unfolding of the information age. The competitive advantage in manufacturing has shifted from the mass production paradigm to one based on flexibility and rapid response [5]. Industry 4.0 is vital to take production capabilities to a new level, with transparent data for decentralized decision-making and flexible production technologies for mass customization [6].

The adoption of Industry 4.0 models and the emergence of Artificial Intelligence (AI) applications in everyday life bring a reflection on how these technologies could be used in industry [7]. Aiming to obtain some of these answers, this study examines the feasibility and benefits of replacing traditional PLCs with machine learning-driven control systems in robotic cells, addressing practical limitations, challenges, and advantages over conventional approaches. The application is validated with the programming and control of a robotic cell, from the training of an Artificial Neural Network (ANN) with Boolean data. The commissioning of the classic model and the model with machine learning is done virtually (VC) using digital manufacturing software. Virtual commissioning (VC) is the process of using a digital model to test and validate a system's design and operation before physical implementation. This approach can reduce design time and mitigate the risks associated with physical testing. The use of models like this introduces the possibility of the controller learning what the expected behavior of the cell should be from data observations and, once learned, this behavior can be compiled into a mathematical control model. While the PLC remains the most commonly used control element in industrial automation, the rise of Industry 4.0 has increased the demand for mass customization, flexibility, and adaptability. As a result, there is a gradual shift away from using PLCs, as their inherent characteristics do not fully meet these emerging requirements. The use of an ANN-based control system can enhance the flexibility and adaptability of manufacturing systems, which is a key concept of Industry 4.0. This work aims to answer if a traditional PLC can be replaced by an ANN in controlling a robotic cell, while also assessing the advantages and disadvantages of this substitution.

The work is organized as follows: Section II presents a literature review, considering the main topics addressed in the developed solution. The modeling of the robotic cell, considering the replacement of the conventional control system by machine learning, is presented in Section III. The results and discussions are presented in Section IV. Section V finishes the article with the conclusions.

The associate editor coordinating the review of this manuscript and approving it for publication was Alexandre S. Brandao (*Corresponding author: Fabio Lima*).

G. B. D. Miranda, and Fábio Lima are with the Centro Universitário FEI, São Bernardo do Campo, Brazil (e-mails: Gabriel.Miranda@kuka.com, and flima@fei.edu.br).

II. Literature Review

A. Theoretical Background

1) Industry 4.0: Information and Communication Technologies (ICT) are in rapid development, and this technological advance enters the manufacturing industry, marking the advent of the fourth stage of industrial production [8]. Industry 4.0 is a strategic initiative of the German government that, according to [9], it is the junction between the digital, physical, and biological worlds and their respective technologies, focused on the creation of intelligent factories, where technologies are updated and transformed by CPS [10], IoT [11], and cloud computing [12].

In the Industry 4.0 era, manufacturing systems are capable of monitoring physical processes through what is called a "digital twin" [13], [14], [15] and making intelligent decisions through real-time communication and cooperation with humans, machines, and sensors. Industry 4.0 combines embedded production system technologies with intelligent production processes [16].

The need to reduce development time along with the growing demand for more consumer-oriented products has led to the next generation of information technology systems in manufacturing [17]. Virtual manufacturing [18] is a computer system technology that integrates manufacturing activities with models and simulations, rather than real-world objects and their operations. This provides a tool for optimizing production efficiency through, mainly, simulations before starting actual production [5].

2) Artificial Intelligence: With the popularization of the internet, the universal existence of sensors, the emergence of Big Data, the development of e-commerce, and the merging of knowledge with the physical and cyber-physical society, the information environment for the development of artificial intelligence has been profoundly transformed [19]. Artificial intelligence will be an important design resource in the future of automation, playing a fundamental role in reducing the programming and engineering efforts required to create automation solutions, making control logic more agile and production processes more flexible and precise, as in the application presented in [20].

In the field of artificial intelligence, machine learning has emerged as a method of choice for the practical development of software for computer vision, speech recognition, robotic control, and other applications. Developers recognize that, for many applications, it can be easier to train systems by demonstrating examples of desired inputs and outputs than to program them manually, anticipating responses to all possible inputs. A perspective on the use of new machine-learning techniques for automation can be found in [21].

Artificial neural networks are an artificial intelligence information processing technique inspired by the biological brain model, which contains numerous interconnected neurons. Its structure seeks to mathematically represent the human cognition model, such as the ability to learn and store information from past experiences [22]. Basically, artificial neural networks can be divided into three parts: the input layer, hidden layers, and output layer. A feedforward architecture indicates that the processed information follows only one direction, from the input layer to the output layer. Multilayer perceptron networks are feedforward networks with a series of applications in complex and nonlinear problems [23].

The training process of an artificial neural network can be performed, basically, in a supervised or unsupervised manner. Training is the process of determining the ideal weights and bias points of the artificial neural network. This is done by defining the total error function between the network output and the desired target and then minimizing it in relation to the weights. The training process of a multilayer perceptron network is given through a supervised technique called backpropagation, whose name refers to the backward propagation of the error during network training, such that the algorithms of this family use the chain rule several times for the calculation of the partial derivative of the total error function of the network [22].

3) Programmable Logic Controller, Virtual Commissioning and Communication: PLCs are the most widely used industrial controllers, especially for supervisory control, sequencing, and safety logic. The PLC system can normally be programmed in Ladder languages, function block diagrams, and statement lists [23]. The PLC emulates the behavior of an electrical ladder diagram, using an input/output symbol table and a scan cycle, as they are sequential machines, to emulate the operation of parallel circuits that respond instantly. The program scan resolves the Boolean logic related to the input information with the internal relay output tables, used to command devices [24].

To succeed in global competition, builders are under constant pressure to deliver low-cost, high-quality products in the shortest time to market. Not only the need for an efficient and flexible production system but also the development time of these systems must be considered [25], [24]. In traditional commissioning, an automation program is tested on the actual machines on the production line. VC emerged as a response to the need for a technique that brings time and cost efficiency to the commissioning process [26].

VC for production lines has advanced as an innovation brought about by the fourth industrial revolution. In traditional approaches, the program was only tested on the factory components, and only then were design and automation errors detected and corrected, a period that used about 25% of the total time of the construction phase [26]. With the application of VC, a reduction of between 10 and 30% in system commissioning time is expected, in addition to increasing the quality of engineering solutions [27]. One of the biggest motivations for using VC is to reduce testing and integration time during the development phase before the physical production system is fully installed [25]. The VC of a production system can be considered as a simulation involving a real PLC interacting with a virtual plant [24].

Developed by the OPC Foundation, OPC (Open Platform Communications) is the interoperability standard for the secure and reliable exchange of data. The OPC standard is a series of specifications developed by industrial vendors, end users, and software developers. These specifications define the interface between Clients and Servers [28].

B. Related Works

The following section presents various works related to the solutions proposed in this article, organized into distinct groups. The first group focuses on the use of PLC within the context of Industry 4.0. The second group highlights various VC solutions, which are another significant contribution of this article. Finally, we discuss initiatives that involve both PLC and machine learning. The section concludes with a summary of the literature review, including Table I, and the key contributions of our work.

The work [29] presents a significant discussion about the future of PLC within the context of Industry 4.0. The authors explore both the advantages and disadvantages of PLC in this new industrial framework, highlighting the relevance of our article. They state that trends like Industry 4.0 foresee decentralized control and enhanced intelligence. Currently, PLCs function as centralized controllers that communicate with sensors and actuators, forming part of the traditional automation pyramid.The authors emphasize that for PLC opportunities in industry, the systems must be testable using virtual prototypes. Additionally, the behavior of the software should be more independent of the hardware. This independence allows for the deployment of new hardware without disrupting existing functions.

In [30], the authors propose a service-oriented automation framework. Among the various elements treated as services is a PLC. One of the benefits of this type of implementation is to ensure interoperability, from an automatic communication network between applications and microservices. The authors' approach is significant in the context of PLC usage and Industry 4.0, as interoperability is one of the main challenges in implementing this new manufacturing model. Additionally, the transition away from PLCs in the industry should occur gradually.

The study by [31] highlights the significance of aligning VC strategies with Industry 4.0 implementation. The authors utilized VC to facilitate communication between a PLC and an equipment modeling software. In our research, we extended the use of VC beyond just the communication between the PLC and the virtual model of the robotic cell; we also employed it for the interaction between the ANN-based control system and the virtual model of the cell.

The study described in [32] predicts the VC between a physical PLC and a virtual plant for the purpose of tuning a PID controller. VC is especially important in mission-critical systems. In our research, VC played a crucial role in validating the ANN-based control system. Conducting physical tests without proper virtual validation can pose risks to both equipment and operators.

The work presented by [23] foresees the development and implementation of a neural block for PLC programming software. In that application, an ANN was trained to replace the continuous control of a temperature plant. Unlike our work, the control being replaced is continuous rather than discrete event-based. Additionally, the goal is not to replace the PLC with more flexible hardware, but rather to integrate a new programming block into the existing PLC.

The work presented in [33] describes an alternative design using an open-source PLC that was modified to encrypt all data sent over the network, regardless of the protocol used. In addition, a machine learning-based intrusion protection system was added to the PLC's network stack, providing a secure mechanism against attacks. Although this application differs from our proposal, it highlights the necessity for PLC adaptations given the new connectivity context of Industry 4.0, specifically by integrating machine learning into PLC systems to enhance security.

The authors in [34] propose a framework based on automatic synthesis methods that learn the behavior of an existing PLC and generate state machines that can be incorporated into IEC 61499 function blocks. The control method employed is discrete event control, as outlined in our proposal. However, the traditional PLC is being replaced by a functional block within a new PLC. Our work suggests substituting the discrete event control of the PLC with a solution based on ANN. As discussed later, this approach can be implemented on lower-cost hardware, providing greater flexibility to the control system and enhancing its alignment with Industry 4.0.

The work [35] shows that machine learning can be leveraged to assist the automation engineer in classifying automation code, locating similar code snippets, and reasoning about the selection of sensor and actuator hardware. The architecture was validated on two real datasets consisting of 2,927 Arduino projects and 683 PLC projects. The goal was to develop and integrate intelligent assistants in existing automation engineering development tools to minimally disrupt existing workflows. Testing the PLC solution enhances the integration of machine learning technologies into programmable logic controller systems.

The work developed in [36] uses a combination of some machine learning concepts and algorithms to develop a framework for configuring parameters in an industrial environment containing PLC and Human-Machine Interface (HMI) from a Supervisory Control and Data Acquisition (SCADA) System. In this solution, PLC input signals are utilized to replace the current SCADA system with an intelligent SCADA system. While the system does incorporate PLC technology, the focus is on upgrading the SCADA system itself, with a machine learning approach that employs multiple linear regression and decision trees. In contrast, our proposal suggests replacing the PLC.

In [37], a framework was developed for the introduction of the Industrial Internet of Things (IIoT) in a heating process. The connected hardware for validation contains a PLC and there is data preparation for use in machine learning.

The literature review indicates that PLCs need to be adjusted or replaced to meet the flexible demands of Industry 4.0. Additionally, it highlights the significance of VC in testing new systems, which promotes both agility and safety during these evaluations. To the best of our knowledge, there are currently no studies that explore the replacement of PLCs in discrete event control with a machine learning-based system that can be implemented in embedded systems and tested through VC methods. Therefore, this work addresses an important research gap by providing a potential solution.

TABLE I
RELATED WORKS

Article	Subject	Contributions
[29]	Industry 4.0	Future of PLC within the context of Industry 4.0
[30]	Industry 4.0	Service-oriented automation framework in Industry 4.0
[31]	Virtual Commissioning	The significance of aligning VC strategies with Industry 4.0
[32]	Virtual Commissioning	The importance of VC in mission-critical systems
[23]	Machine Learning	Implementation of a neural block for PLC programming software
[33]	Machine Learning	Machine learning-based intrusion protection system added to a PLC
[34]	Machine Learning	Framework based on automatic synthesis methods to learn the behavior of an existing PLC
[35]	Machine Learning	Machine learning used to classify automation code
[36]	Machine Learning	Machine learning to develop a framework for configuring parameters in an industrial environment
[37]	Machine Learning	Framework developed for the introduction IIoT in a heating process

III. MATERIAL AND METHODS

A. General

For the elaboration of the robotic cell simulation model, the Tecnomatix - Process Simulate software was used. Process Simulate is a digital manufacturing solution for verifying the manufacturing process in a 3D environment, allowing virtual validation of the main concepts of manufacturing. Process Simulate Commissioning allows the simplification of existing manufacturing and engineering data from conceptual design to the factory floor. With virtual commissioning, it is possible to simulate the actual PLC code with the hardware that will be used through OPC communication, in addition to using the actual robot programs, thus allowing a more realistic virtual commissioning environment [38]. For this research, a robotic cell simulated has the following hardware resources:

1) A Fanuc Arcmate 100iB robot
2) An input device
3) An output device
4) A stationary application pedestal
5) A vacuum suction cup gripper
6) Cell closure with door, pushbuttons, light barriers, and indicator lights

Fig. 1 presents an image of the robotic manufacturing cell used in this work.

Fig. 1. Image of the robotic cell.

B. Signals

For virtual commissioning to take place, it is necessary to create an identical model in terms of resources, operations, and behavior in the simulation software, a model that can be seen in Fig. 2.

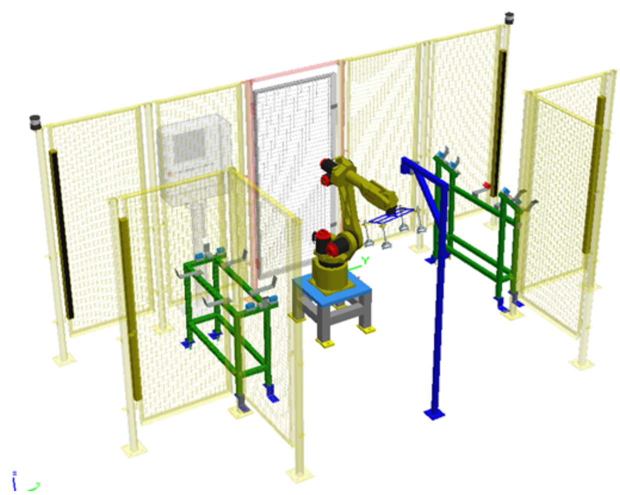


Fig. 2. Virtual model of the cell.

The studied cell presents an operation flow where the operator is used for supply, the automatic operation of the robot is performed, and the finished part is removed. The operation sequences and conditions for its correct functioning can be observed in the flowchart in Fig. 3.

For the correct application of virtual commissioning, it is necessary that the simulated model contains all the signals used by the process and that its behavior corresponds exactly to reality, thus guaranteeing its correct logical and operational functioning. Therefore, it was necessary to survey all existing signals in the cell, signals that are listed in Tab. II.

With the signals necessary for the operation of the cell mapped, these must be duly created in the simulation software, paying attention to the correct classification of inputs and outputs, and enabling the external connection to the server used.

C. Communication

To enable integration between the simulation software (Process Simulate) and the control software Matlab [39], this

TABLE II
LIST OF SIGNALS

Inputs	Description	Outputs	Description
PLC_SR_1	Part sensor on the gripper	PLC_OP_1	Operator in zone 1 (Input)
PLC_SF_1	Part sensor input device	PLC_OP_2	Operator in zone 2 (Output)
PLC_SF_2	Part sensor output device	PLC_RG_0	Send robot to Home
PLC_RZ_0	Robot in Home position	PLC_RG_1	Robot release for zone 1
PLC_RZ_1	Robot in zone 1 (Input)	PLC_RG_2	Robot release for zone 2
PLC_RZ_2	Robot in zone 2 (Output)	PLC_LR_1	Zone 1 indicator light - Red
PLC_RI_1	Robot vacuum signal	PLC_LG_1	Zone 1 indicator light - Green
PLC_BZ_1	Zone 1 input request button	PLC_LB_1	Zone 1 indicator light - Blue
PLC_BZ_2	Zone 2 input request button	PLC_LR_2	Zone 2 indicator light - Red
PLC_BC_0	Emergency button	PLC_LG_2	Zone 2 indicator light - Green
PLC_BC_1	Process start button	PLC_LB_2	Zone 2 indicator light - Blue
PLC_BC_2	Process end button	PLC_RO_0	Emergency signal
PLC_SC_1	Safety signal light barrier 1	PLC_RO_1	Start signal for the robot
PLC_SC_2	Safety signal light barrier 2	PLC_RO_2	End signal for the robot

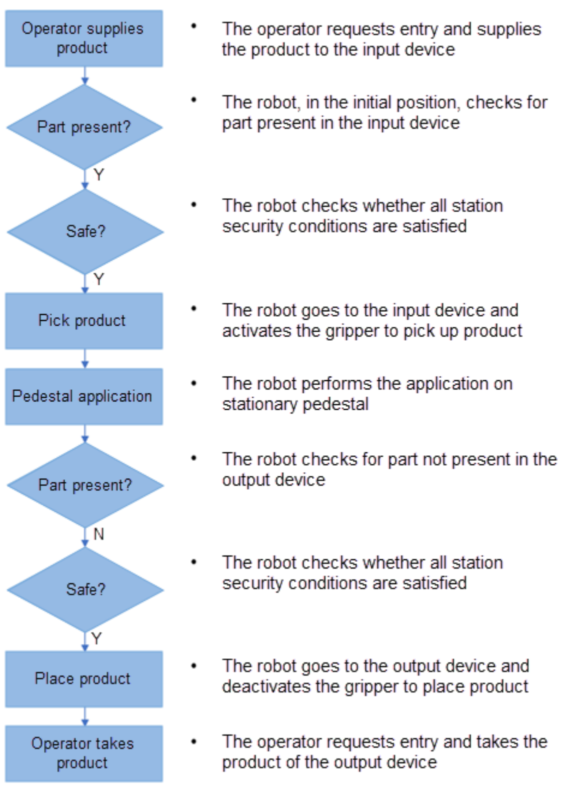


Fig. 3. Process flowchart.

work uses an OPC communication protocol. This work brings the replacement of a PLC by the Matlab software which, using integration with the OPC protocol, is also capable of certifying the logic in the simulation, but in this case with the mathematical resource of an ANN.

To establish an OPC network, it is necessary to use software that operates as a server, thus providing the sharing of the signals created in it with the integrated software. There are several OPC servers available, and in this work, the Matrikon OPC Server GDA software was used, which has an easy-to-use interface and complete integration in applications for general purposes. Generic Data Access (GDA) OPC communication

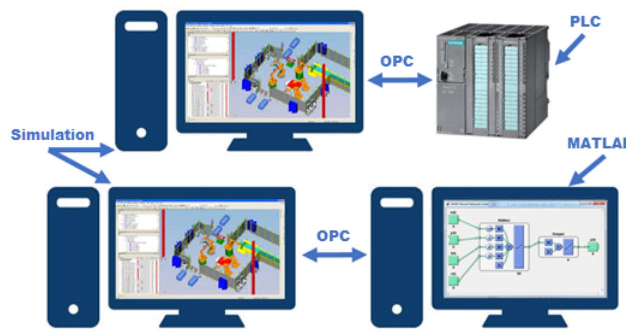


Fig. 4. Communication scheme.

is used because, due to the low complexity of the data used in this research and the easy integration between the software, a platform with data history (OPC-HDA) or unified architecture (OPC-UA) is not necessary, although it is also possible to use them for this purpose.

D. Boolean Data Generation

After elaborating on the simulation model, it is necessary to model the system that will control the cell, replacing conventional PLC. For this purpose, Matlab software is used as a control interface, thus having resources to elaborate the process operation logic, extract training data, train an ANN, and control the simulation model with the result of the training, communicating with the simulation software through the available OPC interface. Matlab software is a mathematical tool and a computational development environment. Its interface works with high-level programming and block language through the package called Simulink. The Simulink extension, widely used in control theory and digital signal processing, is a tool for modeling, simulating, and analyzing dynamic systems with a graphical block diagramming interface, offering integration with the rest of the Matlab environment [40].

Aiming to obtain the data for training the neural network, the first control modeling elaborated was represented by boolean logic, according to the correct functioning of the cell signals. Therefore, in a Simulink model, OPC connection blocks are added, and configured according to the server used, from which it receives and updates the information on the states of the cell signals. From an OPC reading block, an expansion of the input signal vector is made, and from these signals, the logic blocks are inserted in order to meet the cell's process logic. After passing through these logic blocks, the signal values are updated and arranged in vector form again, and thus, transmitted to the OPC writing block, which will update the variable values on the server.

The system controlled by logic blocks is presented in Fig. 5. Boolean logic was designed to align with the process flow illustrated in Fig. 3. To ensure effective control of this flow, the digital signals shown in Table II are utilized. It is possible to observe that there is a flow of information from the left (starting at the OCP server reading block) to the right (writing block on the OPC server). Between the communication blocks, there is the presence of logic gates arranged in a way to correctly represent the cell's operation logic, transforming the received values and transmitting them to the output.

To be able to extract the necessary data for training, a block (called "IOs") was added to the control system that transmits the values of the input signal vector and the output signal vector to a Matlab variable. At each processing instant, these values are added and stored in this variable, resulting in a matrix containing the values of each process signal, thus recording all existing logical variations.

Once the integration between the simulation software and the control software is established, it is necessary, therefore, to execute the simulation model integrated with the boolean control in order to explore the maximum number of process situations and conditions, covering the largest range of signal combinations possible. Having recorded the matrix containing the signal values during the simulation, these values are exported to a ".csv" type file, which will be used to subsequently feed the training process.

E. ANN Training

With the training data exported, the ANN pattern recognition module of the Matlab software is used, which allows solving data classification problems using a feed-forward Multi-Layer Perceptron (MLP) type network. The supervised learning algorithm used by the neural module is the gradient backpropagation, which guarantees rapid convergence. It improves upon the standard conjugate gradient method by adapting the step size dynamically, and more efficiently.

To determine the number of neurons to be used in the proposed model, tests were performed with some topologies. Topologies with 3, 5, and 8 neurons were empirically tested.

Comparing the desired outputs with the outputs obtained from the model, it was possible to notice that the topology with 5 neurons managed to present, in some training sessions, 100% of the combinations taught. Therefore, it was found that, for the 44 logical combinations taught, the network topology must contain a number of 5 to 8 neurons in its hidden layer in order to guarantee the correct operation of the studied cell. As the objective of this network is to acquire the logic presented to it and not to perform a generalization, the entire dataset obtained must be used for the training process, being used by the neural training module, randomly and automatically, 5% of this same set for testing and validation.

The training process ends automatically when the validation set error stops decreasing, presenting the generation with the best performance. As the initial synaptic weights have random values, the training must be run a few times until a better and satisfactory training result is verified.

Therefore, the chosen neural model, illustrated in Fig. 6, has 14 inputs and 14 outputs, consisting of 5 neurons in the hidden layer, whose training obtained 100% accuracy of the combinations presented. Out of the 14 inputs, three had constant values and were automatically ignored during the neural network training process. Consequently, for the 5 neurons in each hidden layer, 11 synaptic weights (W) were generated, which are presented in the Appendix. The MATLAB module used is a two-layer feed-forward network, with sigmoid hidden and softmax output neurons, which can classify vectors arbitrarily well, given enough neurons in its hidden layer. The number of epochs was determined by convergence. Based on the cross-entropy error, the training stops when the error is detected to be constant.

With everything defined and with a satisfactory learning result, the network is compiled into a single block of the Simulink package, which will replace all the Boolean logic of the control system. Therefore, the final control system can be checked in Fig. 7.

Similar to the previous boolean system, the system will update the values of the OPC server signals according to the signals emitted by the simulation software. It is important to emphasize, however, the need to use a block to add a discontinuity in the output of the neural network, because the resulting value of the neural block output is not boolean, and because it is no longer a boolean signal, any values other than zero would be interpreted as true. Therefore, it is necessary to insert a true or false value selector, such as a dead zone block, so that only signals greater than a certain value are considered different from zero, thus solving the problem in the definition of true or false values.

IV. Results and Discussion

To select the best value for the discontinuity block, tests were performed by changing the selection threshold value between true or false and comparing these results with the desired values. It was found that the most suitable value for separating true or false variables should be between 0.1 and 0.2. For the studied model, the value of 0.2 was adopted, guaranteeing 100% accuracy for the combinations taught. The discontinuity block used in the Simulink package is the dead zone, which outputs zero for inputs within the range of 0 to 0.2.

It is possible to conclude, therefore, that for outputs of the neural system, which are no longer in the discrete universe of 0 or 1 but rather any value between 0 and 1, with 0 being a totally deactivated output and 1 being a totally activated output, some combinations did not reach a level higher than 0.25 to assume a true value.

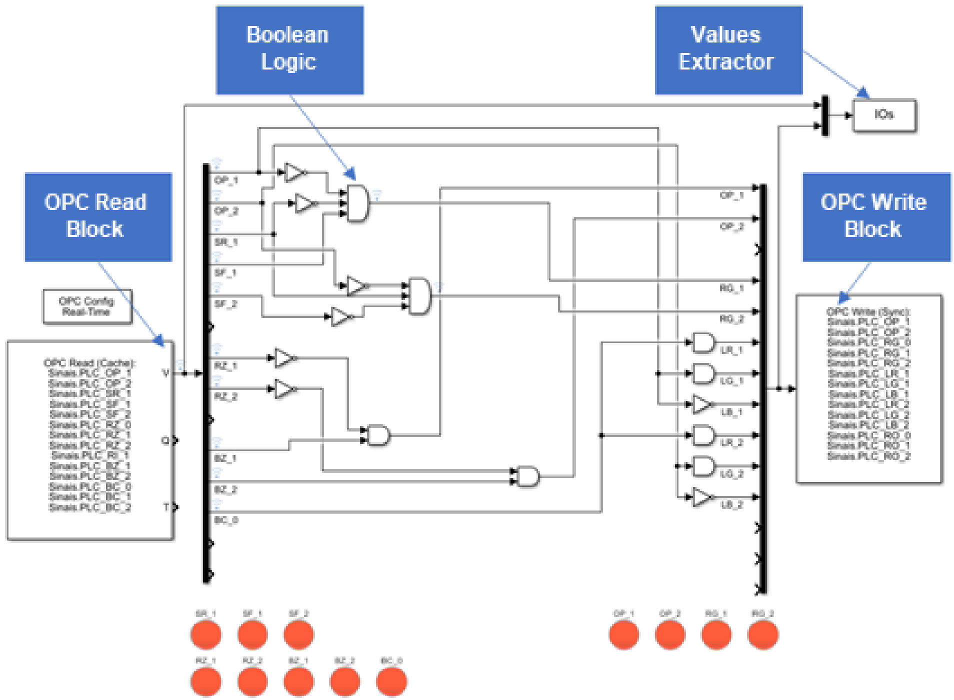


Fig. 5. Boolean control system.

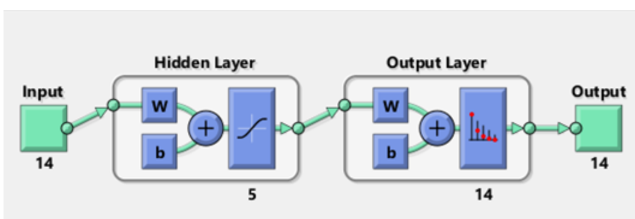


Fig. 6. ANN structure.

First, Figs. 8, 9 and 10 show the ANN training results that defined the number of neurons in the hidden layer. The modeling of the Boolean logic control system, necessary for extracting the training data for the neural network, presented correct functionality, behaving as desired in relation to the flowchart in Fig. 3. There was a good response for the integration of the simulation and control software, however, the response speed depends on the signal update rates of this software together with the OPC server. This rate is configurable and asynchronous between the software, with a cycle time on the scale of milliseconds. The cell signals generated within the simulation model were transmitted to the OPC server, then were read by the boolean control model, and thus, updated and sent back to the simulation model. The execution of this model generated the training data that allowed the learning of the neural model, which demonstrates that the logic employed here will be "copied" to the neural network and, therefore, the execution and testing of the most varied conditions and situations of the process are essential.

After training the neural network, through the data extracted from the previous model, and exchanging the logic gates for the resulting neural block, a functional execution of the system was observed, also respecting the sequence presented in the flowchart in Fig. 3.

As indicated by Fig. 11, there is the execution of a new virtual commissioning of the simulation model, this time using the ANN from the training process. There were no differences when comparing the execution of the cell in the neural model with the boolean model. It is noted, however, that before reaching a satisfactory model, some unexpected conditions occurred in some executions of the neural control, such as the indicator light that was on with a certain color and, for a certain combination of signals, came to turn off and on again when that combination was changed. This type of indecision behavior indicates that some process conditions were not well sampled in the training data, leading to a certain situation that the neural network had not learned. Therefore, a new data extraction was necessary to teach this condition.

Knowing that the network aims to replicate a set of logical combinations presented, and not generalization to unknown outputs, all samples must be used for training. However, the only way to ensure that all logic has been adequately demonstrated would be through a truth table with all possible combinations of inputs and outputs, which, in some cases, can become unfeasible, as 2^i (where "i" is the number of system inputs) different combinations would be needed to be used in the training set. For this research, for example, a truth table would consist of $2^{14} = 16,384$ different logical combinations to ensure that all possible conditions were sampled. This is the disadvantage of the proposed solution, which could make replacing PLC with ANN impractical.

Therefore, due to its purpose of not being a generalization network, logical combinations not used in the training process may or may not present a correct output, but, because it is

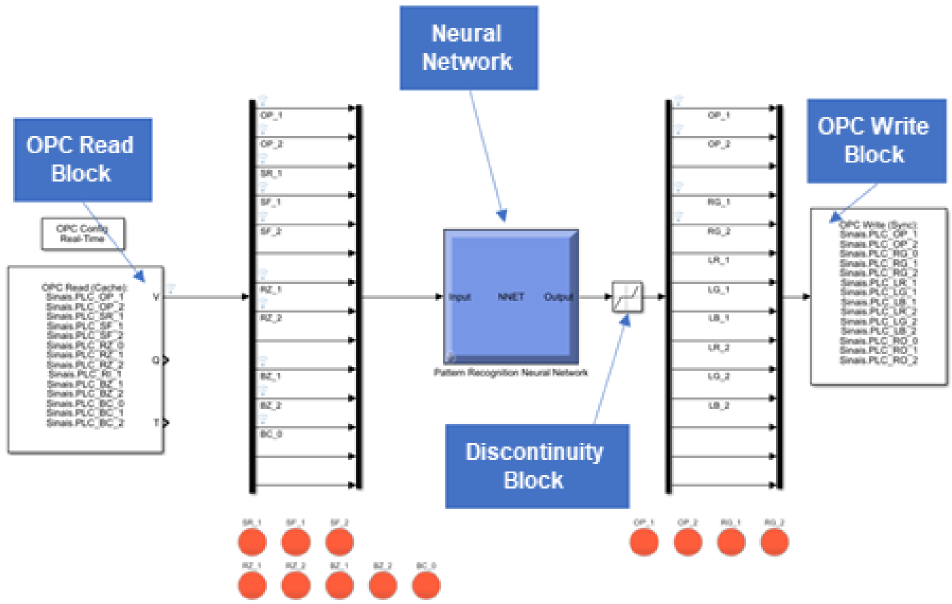


Fig. 7. ANN control system.

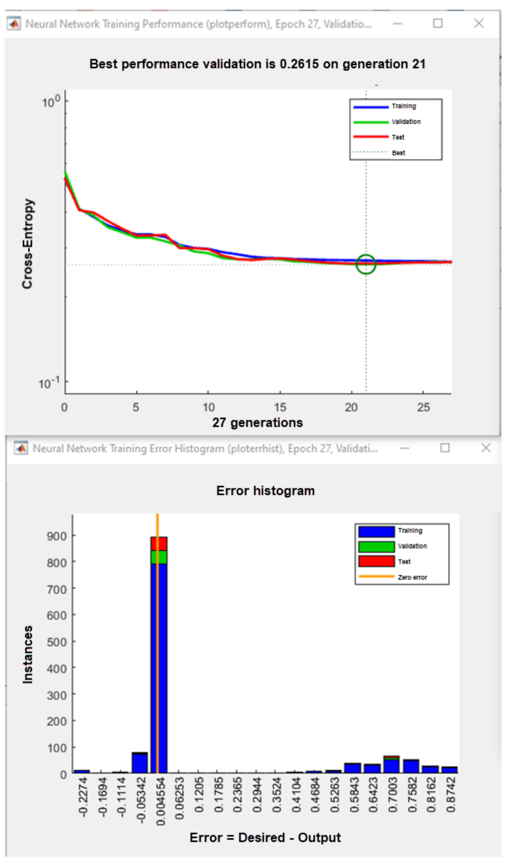


Fig. 8. Test with three neurons.

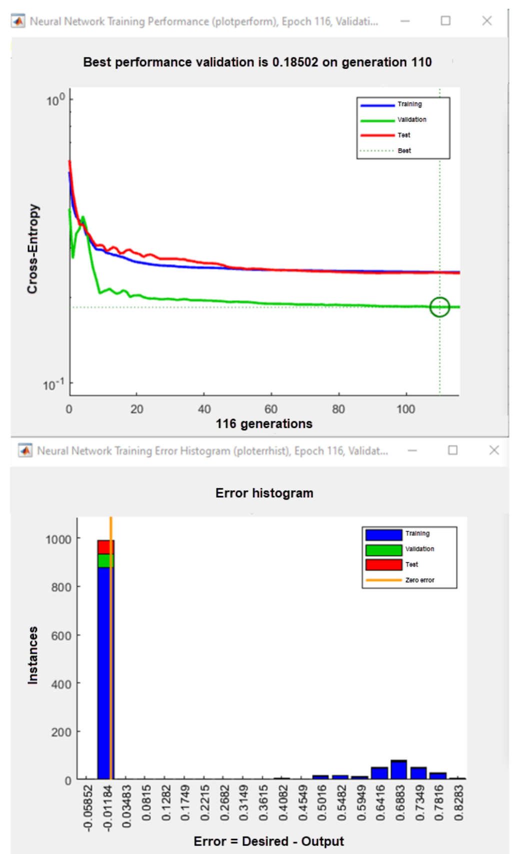


Fig. 9. Test with five neurons.

not a linear relationship between inputs and outputs, they will always be in the field of uncertainty. This indicates the importance of good sampling in this case, and that the number of logical combinations available, or possible to be sampled, could already make this approach unfeasible in a certain application.

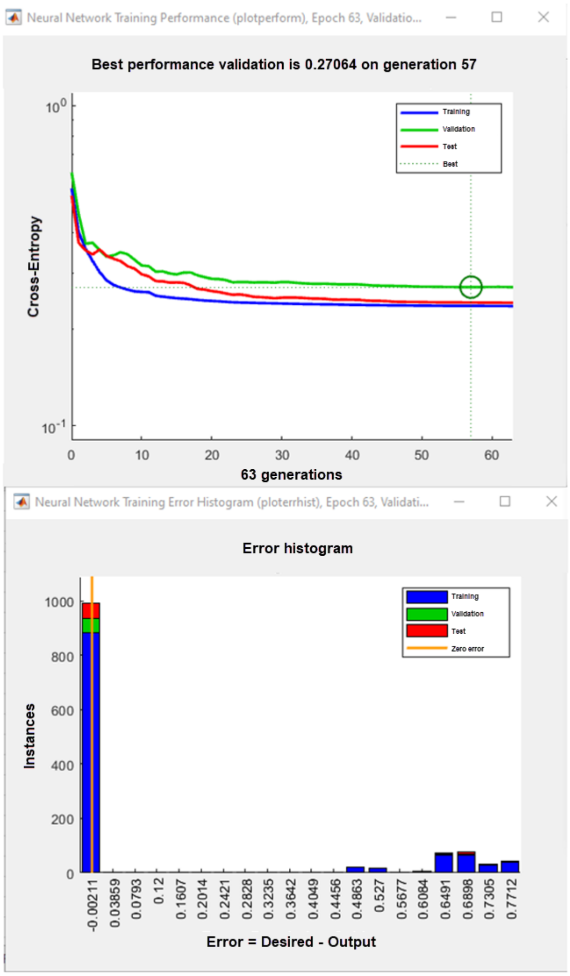


Fig. 10. Test with eight neurons.

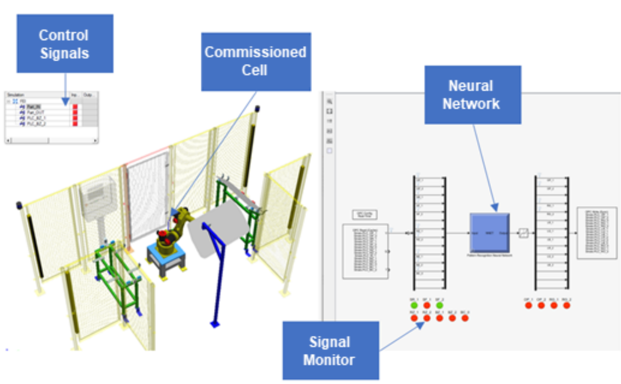


Fig. 11. ANN controled system.

V. Conclusion

This work presented the implementation of virtual commissioning of a robotic cell through ANN, replacing conventional PLC. The proposed control model aimed to be an alternative to the control methods already established in the industry, in an integrated approach with Industry 4.0 regarding the use of AI methods in industrial processes.

The modeling used deals with a virtualized cell integrated with a neural controller, whose operation logic was extracted from a boolean model. Therefore, it is necessary to initially develop a functional control model of the process, and then the data for training the neural network can be extracted from it. The performance of the ANN control showed itself reliable and the VC proved that this implementation is feasible.

Although conventional PLC remain widely used, translating process logic into a mathematical model can provide execution advantages over traditional PLCs. This method can also be implemented on low-cost hardware. The integration of artificial intelligence and its tools creates new opportunities, such as incorporating Boolean logic behavior into a Neural Processing Unit (NPU). This capability allows the neural network to manage both signal processing and logical decision-making, as well as other applications like quality control. Furthermore, utilizing neural processing enhances the system's compatibility with Industry 4.0 approaches.

This research proposes future work utilizing a control network with online training. This approach would enable the neural network to learn the behavior of the process adaptively. Additionally, it suggests incorporating an extra output to manage unexpected or undefined input states. Finally, we suggest testing other machine learning methods.

Acknowledgments

The authors thank Centro Universitário FEI for financial and infrastructure support.

Appendix

$$W = \begin{bmatrix} 0.0449 & -0.2294 & 2.9516 & -0.6393 & -0.0334 \\ 1.9164 & 1.7192 & -1.5759 & 0.5413 & -2.4679 \\ -0.7539 & -0.4366 & 0.3455 & -1.1475 & -1.2634 \\ 0.8898 & 0.8684 & 0.0281 & 0.5164 & -0.0010 \\ -0.2618 & 0.0771 & -0.8443 & 0.0568 & 1.3904 \\ 0.7785 & -0.1806 & 0.0721 & -0.2046 & 1.1230 \\ 0.1179 & -0.0547 & -0.0106 & 0.1887 & 0.0063 \\ 0.0992 & -0.0304 & 0.0101 & -0.1316 & -0.0082 \\ -0.7374 & -0.3731 & 0.3761 & -1.7459 & -0.4509 \\ 0.1779 & -0.9906 & 0.1057 & 2.0518 & -0.1261 \\ -0.0258 & 2.3727 & 0.4301 & 1.2879 & -0.9031 \end{bmatrix}$$

References

[1] K. Wang, "Intelligent predictive maintenance (ipdm) system – industry 4.0 scenario," *WIT Transactions on Engineering Sciences*, vol. 113, pp. 260–268, 2016. [Online]. Available: https://doi.org/10.2495/IWAMA150301

[2] J. Qin, Y. Liu, and R. Grosvenor, "A categorical framework of manufacturing for industry 4.0 and beyond," *Procedia CIRP*, vol. 52, pp. 173–178, 2016. [Online]. Available: https://doi.org/10.1016/j.procir.2016.08.005

[3] P. Leitão, F. Pires, S. Karnouskos, and A. W. Colombo, "Quo vadis industry 4.0? position, trends, and challenges," *IEEE Open Journal of the Industrial Electronics Society*, vol. 1, pp. 298–310, 2020. [Online]. Available: https://doi.org/10.1109/OJIES.2020.3031660

[4] T. Wuest, D. Weimer, C. Irgens, and K.-D. Thoben, "Machine learning in manufacturing: advantages, challenges, and applications," *Production & Manufacturing Research*, vol. 4, no. 1, pp. 23–45, 2016. [Online]. Available: https://doi.org/10.1080/21693277.2016.1192517

[5] W. Lee, C. Cheung, and J. Li, "Applications of virtual manufacturing in materials processing," *Journal of Materials Processing Technology*, vol. 113, no. 1, pp. 416–423, 2001. [Online]. Available: https://doi.org/10.1016/S0924-0136(01)00668-9

[6] B. Xu, M. R. Guertler, and N. Sick, "Analyzing industry 4.0 adoption barriers of small and medium-sized enterprises and existing support," in *2023 IEEE Engineering Informatics*, 2023, pp. 1–10. [Online]. Available: https://doi.org/10.1109/IEEECONF58110.2023.10520600

[7] Z. Jan, F. Ahamed, W. Mayer, N. Patel, G. Grossmann, M. Stumptner, and A. Kuusk, "Artificial intelligence for industry 4.0: Systematic review of applications, challenges, and opportunities," *Expert Systems with Applications*, vol. 216, p. 119456, 2023. [Online]. Available: https://doi.org/10.1016/j.eswa.2022.119456

[8] A. Khan and K. Turowski, "A perspective on industry 4.0: From challenges to opportunities in production systems," *Proceedings of the International Conference on Internet of Things and Big Data*, p. 441–448, 2016. [Online]. Available: https://doi.org/10.5220/0005929704410448

[9] K. Schwab, *The Fourth Industrial Revolution*. Crown Currency, 2017.

[10] F. J. Huertos, B. Chicote, M. Masenlle, and M. Ayuso, "A novel architecture for cyber-physical production systems in industry 4.0," in *2021 IEEE 17th International Conference on Automation Science and Engineering (CASE)*, 2021, pp. 645–650. [Online]. Available: https://doi.org/10.1109/CASE49439.2021.9551464

[11] A. Rani, V. K. Mishra, A. K. Mishra, V. Kumar, and N. K. Pandey, "Role and significance of internet of things (iot) in industry 4.0," in *2024 3rd International conference on Power Electronics and IoT Applications in Renewable Energy and its Control (PARC)*, 2024, pp. 199–202. [Online]. Available: https://doi.org/10.1109/PARC59193.2024.10486519

[12] M. Sharifzadeha, H. Malekpoura, and E. Shojab, *Cloud Computing and Its Impact on Industry 4.0*, 2022, pp. 99–120. [Online]. Available: https://doi.org/10.1002/9781119695868.ch4

[13] F. Pires, A. Cachada, J. Barbosa, A. P. Moreira, and P. Leitão, "Digital twin in industry 4.0: Technologies, applications and challenges," in *2019 IEEE 17th International Conference on Industrial Informatics (INDIN)*, vol. 1, 2019, pp. 721–726. [Online]. Available: https://doi.org/10.1109/INDIN41052.2019.8972134

[14] Q. Qi and F. Tao, "Digital twin and big data towards smart manufacturing and industry 4.0: 360 degree comparison," *IEEE Access*, vol. 6, pp. 3585–3593, 2018. [Online]. Available: https://doi.org/10.1109/ACCESS.2018.2793265

[15] S. Manickam, L. Yarlagadda, S. P. Gopalan, and C. L. Chowdhary, "Unlocking the potential of digital twins: A comprehensive review of concepts, frameworks, and industrial applications," *IEEE Access*, vol. 11, pp. 135 147–135 158, 2023. [Online]. Available: https://doi.org/10.1109/ACCESS.2023.3338530

[16] R. Y. Zhong, X. Xu, E. Klotz, and S. T. Newman, "Intelligent manufacturing in the context of industry 4.0: A review," *Engineering*, vol. 3, no. 5, pp. 616–630, 2017. [Online]. Available: https://doi.org/10.1016/J.ENG.2017.05.015

[17] G. Chryssolouris, D. Mavrikios, N. Papakostas, D. Mourtzis, G. Michalos, and K. Georgoulias, "Digital manufacturing: History, perspectives, and outlook," *Proceedings of the Institution of Mechanical Engineers, Part B: Journal of Engineering Manufacture*, vol. 223, no. 5, pp. 451–462, 2009. [Online]. Available: https://doi.org/10.1243/09544054JEM1241

[18] Z. Bi, A. Mikkola, A. W. H. Ip, K. L. Yung, and C. Luo, "Virtual verification and validation to enhance sustainability of manufacturing systems," *IEEE Transactions on Automation Science and Engineering*, pp. 1–10, 2024. [Online]. Available: https://doi.org/10.1109/TASE.2024.3370053

[19] B. hu Li, B. cun Hou, W. tao Yu, X. bing Lu, and C. wei Yang, "Applications of artificial intelligence in intelligent manufacturing: a review," *Frontiers Inf Technol Electronic Eng*, vol. 18, pp. 86–96, 2017. [Online]. Available: https://doi.org/10.1631/FITEE.1601885

[20] S. S. Sheuly, M. U. Ahmed, S. Begum, and M. Osbakk, "Explainable machine learning to improve assembly line automation," in *2021 4th International Conference on Artificial Intelligence for Industries (AI4I)*, 2021, pp. 81–85. [Online]. Available: https://doi.org/10.1109/AI4I51902.2021.00028

[21] B. Maschler and M. Weyrich, "Deep transfer learning for industrial automation: A review and discussion of new techniques for data-driven machine learning," *IEEE Industrial Electronics Magazine*, vol. 15, no. 2, pp. 65–75, 2021. [Online]. Available: https://doi.org/10.1109/MIE.2020.3034884

[22] I. N. da Silva, D. H. Spatti, R. A. Flauzino, L. H. B. Liboni, and S. F. dos Reis Alves, *Artificial Neural Networks: A Practical Course*. Springer, 2017.

[23] H. Abdi, A. Salami, and A. Ahmadi, "Implementation of a new neural network function block to programmable logic controllers library function," *International Journal of Computer and Systems Engineering*, vol. 1, no. 5, pp. 745–748, 2007. [Online]. Available: https://doi.org/10.5281/zenodo.1062844

[24] M. Ko, E. Ahn, and S. C. Park, "A concurrent design methodology of a production system for virtual commissioning," *Concurrent Engineering*, vol. 21, no. 2, pp. 129–140, 2013. [Online]. Available: https://doi.org/10.1177/1063293X13476070

[25] 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, 8th IFAC Conference on Manufacturing Modelling, Management and Control MIM 2016. [Online]. Available: https://doi.org/10.1016/j.ifacol.2016.07.914

[26] A. Aras, M. Ayaz, E. Ozdemir, and N. Abut, "Investigation on industry 4.0 and virtual commissioning," *International Journal of Engineering Technologies-IJET*, vol. 4, no. 2, pp. 107–113, 2018. [Online]. Available: https://doi:10.19072/ijet.412125

[27] R. Drath, P. Weber, and N. Mauser, "An evolutionary approach for the industrial introduction of virtual commissioning," in *2008 IEEE International Conference on Emerging Technologies and Factory Automation*, 2008, pp. 5–8. [Online]. Available: https://doi.org/10.1109/ETFA.2008.4638359

[28] P. Blanco, M. Poli, and M. Barretto, "Opc and corba in manufacturing execution systems: a review," in *EFTA 2003. 2003 IEEE Conference on Emerging Technologies and Factory Automation. Proceedings (Cat. No.03TH8696)*, vol. 2, 2003, pp. 50–57 vol.2. [Online]. Available: https://doi.org/10.1109/ETFA.2003.1248669

[29] M. A. Sehr, M. Lohstroh, M. Weber, I. Ugalde, M. Witte, J. Neidig, S. Hoeme, M. Niknami, and E. A. Lee, "Programmable logic controllers in the context of industry 4.0," *IEEE Transactions on Industrial Informatics*, vol. 17, no. 5, pp. 3523–3533, 2021.

[30] M. d. M. Fernandes, J. A. Bigheti, R. P. Pontarolli, and E. P. Godoy, "Industrial automation as a service: A new application to industry 4.0," *IEEE Latin America Transactions*, vol. 19, no. 12, pp. 2046–2053, 2021. [Online]. Available: https://doi.org/10.1109/TLA.2021.9480146

[31] K. Tebani, R. Plateaux, C. Puyenchet, O. Penas, C. Baroux, and S. Limou, "Real-time communication between plc and dymola for virtual commissioning application," in *2020 4th International Conference on Advanced Systems and Emergent Technologies*, 2020, pp. 83–88.

[32] S. Bysko, S. Bysko, M. Fratczak, P. Nowak, T. Klopot, J. Czeczot, K. Stebel, and P. Laszczyk, "Pid controller tuning by virtual commissioning - a step to industry 4.0," *Journal of Physics: Conference Series*, vol. 2198, pp. 1–12, 2022.

[33] T. Alves, R. Das, and T. Morris, "Embedding encryption and machine learning intrusion prevention systems on programmable logic controllers," *IEEE Embedded Systems Letters*, vol. 10, no. 3, pp. 99–102, 2018. [Online]. Available: https://doi.org/10.1109/LES.2018.2823906

[34] D. Chivilikhin, S. Patil, K. Chukharev, A. Cordonnier, and V. Vyatkin, "Automatic state machine reconstruction from legacy programmable logic controller using data collection and sat solver," *IEEE Transactions on Industrial Informatics*, vol. 16, no. 12, pp. 7821–7831, 2020. [Online]. Available: https://doi.org/10.1109/TII.2020.2992235

[35] A. Canedo, P. Goyal, D. Huang, A. Pandey, and G. Quiros, "Arducode: Predictive framework for automation engineering," *IEEE Transactions on Automation Science and Engineering*, vol. 18, no. 3, pp. 1417–1428, 2021. [Online]. Available: https://doi.org/10.1109/TASE.2020.3008055

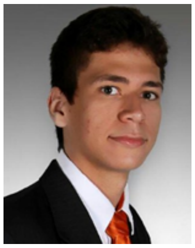
[36] K. S. Kiangala and Z. Wang, "An adaptive framework for configuration of parameters in an industry 4.0 manufacturing scada system by merging machine learning techniques," in *2020 International Conference on Artificial Intelligence, Big Data, Computing and Data Communication Systems (icABCD)*, 2020, pp. 1–6. [Online]. Available: https://doi.org/10.1109/icABCD49160.2020.9183818

[37] S. S. Chaudhari, K. S. Bhole, and S. Rane, "An application of iiot framework in system design, performance monitoring and control for industrial process heater," *International Journal on Interactive Design and Manufacturing (IJIDeM)*, 2023. [Online]. Available: https://doi.org/10.1007/s12008-023-01235-6

[38] Siemens. Tecnomatix process simulate. [Online]. Available: https://plm.sw.siemens.com/en-US/tecnomatix/products/process-simulate-software/

[39] Mathworks. Matlab. [Online]. Available: https://www.mathworks.com/products/matlab-online.html

[40] ——. Simulink. [Online]. Available: https://www.mathworks.com/help/simulink/index.html



Gabriel Bastos de Miranda holds a degree in Electrical Engineering with a focus on Electronics from Centro Universitario FEI (FEI), São Paulo - Brazil, (2017), and a master's degree in mechanical engineering, specializing in the Production area, from the same institution (2020), with a research focus on modeling and advanced manufacturing. In 2019, he served as a professor of robotics and automation at SENAI-Brazil. Over the past 12 years, he has been involved in the automation and planning of automatic lines, specializing in simulation, virtual commissioning, and software development.

Fábio Lima (Member, IEEE) received his B.Sc. and M.Sc. degrees in Electrical Engineering from Paulista State University (UNESP - Brazil) and the University of São Paulo (USP - Brazil) in 1998 and 2001 respectively. He completed his Ph.D. in Electrical Engineering at USP in 2010. He is a full-time professor in the Industrial Engineering department at Centro Universitário FEI (FEI), São Paulo - Brazil, where he teaches subjects related to industrial automation and manufacturing systems. He is also a member of the master's degree program in Mechanical Engineering at FEI in the production systems area. Additionally, he is a scientific adviser to FAPESP (São Paulo Research Foundation - Brazil) and a member of the IEEE Industrial Electronics Society. His presentations at IEEE-IECON conferences in 2013 (Vienna) and 2015 (Yokohama) won the best presentation award. Currently, he is the coordinator of the Digital Manufacturing Laboratory and the 5G Solutions Center at Centro Universitário FEI, developing projects related to Industry 4.0.