Human-Centric Digital Twin-Driven Approach for Plug-and-Produce in Modular Cyber-Physical Production Systems

Andre Dionisio Rocha[Ω], Miguel Arvana[Ω], Nelson Freitas[Ω], Raul Mestre Dinis[∂], Tiago Gouveia[∂], Daniela Machado[∂] and Jose Barata[Ω]
NOVA School of Science and Technology, Center of Technology and Systems (UNINOVA-CTS),
and Associated Lab of Intelligent Systems (LASI), NOVA University Lisbon, 2829-516 Lisbon, Portugal
Email: [Ω]{andre.rocha, m.arvana, n.freitas, jab} @uninova.pt
[∂]{rj.dinis, td.gouveia, dn.machado} @campus.fct.unl.pt

2024 IEEE 29th International Conference on Emerging Technologies and Factory Automation (ETFA) | 979-8-3503-6123-0/24/$31.00 ©2024 IEEE | DOI: 10.1109/ETFA61755.2024.10710730

Abstract— **The evolution of manufacturing, driven by the Fourth Industrial Revolution, demands adaptable and interconnected production methodologies to meet the increasing demand from consumers for personalised products. Cyber-Physical Production Systems have emerged as a crucial innovation in this context, facilitating virtual representations of production resources and enabling dynamic interactions between computational and physical processes. This article proposes a framework integrating Cyber-Physical Production Systems with Digital Twins to understand manufacturing systems and enhance their functionality comprehensively. The framework comprises three layers: Physical Assets, Modular Cyber-Physical Production System, and Digital Twin. Utilising Multi-Agent systems, the Modular Physical Production System orchestrates production processes, optimises efficiency, and ensures adaptability to market fluctuations. Meanwhile, the Digital Twin serves as a real-time representation of the physical system, facilitating monitoring, interaction, and understanding of the system's behaviour. The implementation involves creating 3D models using Blender and a virtual environment using Unity. The proposed framework has demonstrated its effectiveness in addressing the complexities of modern manufacturing, fostering adaptability, and facilitating collaboration between humans and machines.**

Keywords—Digital Twin, Industrial Cyber-Physical Systems, Modularity, Plug-and-Produce, Reconfigurability, Multiagent Systems

I. INTRODUCTION

The manufacturing landscape has experienced significant transformations since the beginning of the first industrial revolution. The fourth industrial revolution has continued this trend, with a marked shift towards the demand for highly customised and personalised products mirroring individual tastes and needs. This evolution in consumer expectations needs a corresponding change in manufacturing paradigms, convincing companies to adopt production methodologies that are modular, flexible, and capable of seamless integration and connectivity to meet new consumer demands.

In the context of Industry 4.0, a broad understanding of the manufacturing system from multiple dimensions becomes imperative [1]. The industry is challenged with adapting to the rapid evolution and expansion of production lines, necessitating a novel approach to the design and development of manufacturing systems. This approach must address the immediate challenges and leverage the potential of heterogeneous and autonomous components, transitioning from focusing on interoperability to promoting a collaborative manufacturing environment.

The Cyber-Physical Production System (CPPS) concept has emerged as a pivotal innovation, facilitating the virtual representation of production resources to enhance functionality and manage complex tasks more effectively [2]. As a particular application of Cyber-Physical Systems (CPS) within industrial settings, CPPS integrates computational and physical processes, enabling dynamic interactions between them. A significant advantage of CPPS lies in its ability to digitally mirror the factory environment digitally, thereby supporting distributed and collaborative strategies. These strategies are essential in creating manufacturing systems that are not only flexible and reconfigurable but also resilient to disruptions. Implementing distributed CPPS through agent-based methods has shown promise, utilising the strengths of Multi-Agent Systems (MAS) for enhanced efficiency and reconfigurability [3].

The advancements in Information and Communication Technologies (ICT) have facilitated the widespread adoption of MAS, positioning them as key enablers of innovative manufacturing practices [4]. The application of agent technology in various industrial contexts has significantly increased, verified by numerous factory implementations and demonstration projects. Despite this, using agent-based approaches for manufacturing control remains an active area of research, with various architectures and solutions being explored [5]. Notable examples include ADACOR2 [6], IADE [7], and BIOSOARM [8]. The FP7 GRACE project demonstrated an agent-based method for monitoring product quality along the production line and dynamically adjusting production parameters to enhance quality or resolve issues, as seen in the Whirlpool pilot [9], [10]. Similarly, the FP7 PRIME project employed agent-based solutions for reconfiguring both standard and legacy production systems, with demonstrations at the University of Nottingham, and for optimising flexible monitoring in the automotive industry, showcased in two industrial cells [11], [12]. Within the openMOS project, the exploration of integrating a MAS with a Manufacturing Service Bus as an integration layer highlighted using agents to control and optimise processes at a higher level, while lower-level software components managed time-critical activities [13], [14]. These developments highlight the diverse potential of agent technologies in enhancing production environments, primarily focusing on overcoming control challenges and contributing to developing new, adaptive control environments responsive to market fluctuations and disturbances.

Although the presented approaches demonstrate advantages in system performance and their ability to self-organize and respond to the needs identified at any given moment, it is difficult to integrate and frame human interaction with these types of solutions due to their complexity. Therefore, it is necessary to develop new tools that assist humans in working and collaborating with these ecosystems to correctly understand what they can do with them, what they should do at each moment, and even perceive and predict what the system is and will do. Various authors have identified some barriers to adopting these types of systems [15]. One of the most relevant barriers is that humans must understand how the systems behave and their different states over time [16].

Through these studies that show the difficulty of these solutions penetrating the market, it is possible to verify that their added value, namely their ability to handle complexity and various scenarios autonomously, is also one of the biggest barriers to their adoption. This is because humans lack the capacity and ease to deal with such complex systems. However, humans will play a fundamental role in the implementation and operation of this type of system, as they will be the key facilitators of this flexibility, responsible for making physical changes, adding/removing stations, launching products, among other tasks. Thus, it becomes essential to answer the following question:

RQ: How can tools be designed to allow human users to understand what the system is doing and increase their confidence in these emerging technologies?

To answer this question, the following hypothesis arises:

H: Through the creation of a digital tool that represents in real-time what is happening in the physical system and allows the operator to have partial control of the control system through reconfiguration functions, it will be possible for operators to be an integral part of the reconfigurable production environment and increase their confidence in these types of systems.

In this way, the proposed work aims to create a Digital Twin (DT) that can, at any moment, represent the current state of the physical components of the system (conveyor belts, stations, products, robots, and human operators, among others) that are connected and operating at that time. DTs can help understand the behaviour of these highly dynamic and reconfigurable systems, even during execution [17], [18]. DTs focus on modelling physical systems' characteristics, behaviours, and performance. It is possible to have a real-time cybernetic high-fidelity representation of the characteristics of the physical world [19]. This virtual model allows for monitoring and interaction with the physical system in a way that is easier for humans and better to understand the operating mode and evolution of the system.

II. Digital Twin Framework

The proposed framework aims to provide a logical representation in the DT of the physical system's current state and the Modular Cyber-Physical Production System (MCPPS) responsible for controlling and optimising the execution of the physical system. Thus, the DT is the top layer of the framework. The following layers can define it:

1) ***Physical Assets:*** This layer contains all the physical components on the factory floor. This includes all workstations, people (operators), and conveyors that are in operation.

2) ***Modular Cyber-Physical Production System:*** This layer abstracts the entire distributed control system, typically consisting of a multi-agent system. Recently, these control systems have been proposed to deliver flexibility and adaptability so that the physical layer can handle disturbances such as small batch production and unexpected orders, among others. This highly dynamic environment consists of distributed entities with social abilities that must collaborate to respond to needs in the best possible way.

3) ***Digital Twin*:** This layer creates the virtual environment, representing the highest-level layer of the entire framework. It is responsible for showing not only the current state of the physical system but also the current configurations of the multi-agent system and, consequently, the physical components that each agent abstracts.

An overview of the proposed framework can be visualised in Fig. 1. The figure shows that the MCPPS consists of a multi-agent system. The abstraction of the physical system is done in a one-to-one relationship in the multi-agent system that creates the Modular CPPS. Thus, there will be as many Industrial Agents (agents abstracting a physical component) as physical components. The multi-agent system performs the production process control, and optimisation is achieved through interactions between different entities. For example, whenever a product requires a task, it must search for an agent that performs that action. When choosing which agent to execute, it asks another industrial agent responsible for transportation to carry out the requested resource.

Therefore, the DT needs to represent not only what the physical environment is doing at any given moment and the state of each machine but also the criteria used by the Modular CPPS to make decisions and reconfigure these criteria as needed. Thus, the physical assets constantly send their state to the DT, and the DT sends the new configurations that must be implemented by the MAS whenever necessary.

III. Engineering Process

Some steps must be followed to implement the proposed framework. The first stage focuses on the study of the physical layer, where the physical components that the remaining layers will abstract are identified. Subsequently, we have the design and implementation of the MCPPS using a MAS. The process ends with creating the DT, designing the physical components in a 3D modelling tool like Blender, and creating the visualisation scenario and animations with a game engine, such as Unity.

A. Physical Assets

The physical assets consist of an educational kit from Staudinger GMBH, a robotic arm from ABB, and an operator who interfaces the system with a human-machine interface (HMI). The educational kit plays a central role in the environment because it simulates a transport system and allows the other two resources to be plugged into two positions.

The educational kit includes six conveyor belts and one station. Regarding the six conveyors, one can perform lateral movement, and two can perform rotational movement, while the other three are simple unidirectional conveyors.

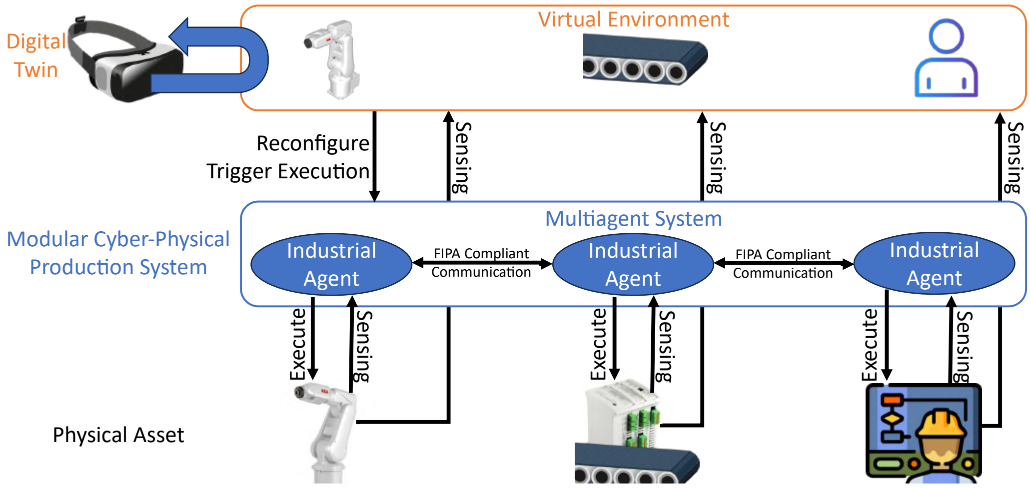

Fig. 1 – Digital Twin Framework for Modular Cyber-Physical Production Systems

The station has an action tool that can be moved backwards, forward, and up and down. An Industrial Shields controller (M-Duino_58+) controls the components of this educational kit. This controller exposes the capabilities provided by the physical components, allowing higher-level software to trigger actions on them.

Fig. 2 shows that resources can be added/removed from the side stations on conveyors B and F.

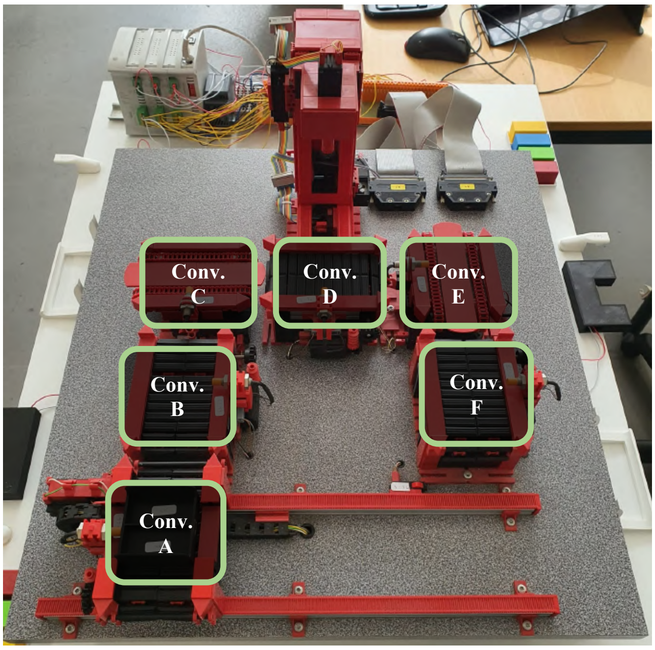

Fig. 2 – Educational kit part of the physical system.

In the work developed, it is possible to add an ABB industrial robot (IRB 120) or a human operator who interacts with the system through an HMI provided on a tablet. Fig. 3 shows a configuration where the robot is connected to conveyor F, and the operator works on conveyor B (it is possible to see the tablet where the operator receives and sends information to the agent that abstracts the operator).

B. Modular Cyber-Physical Production System

The MCPPS used is based on previous work, in which a multi-agent system was developed and integrated with industrial controllers exposing services [20].

Fig. 3 – Physical system with the education kit, operator, and robot.

The developed MCPPS in this study creates a distributed control environment for manufacturing systems structured across three distinct hierarchical layers, each tasked with specific functions. This tri-layered approach is designed to streamline such systems' complexity and operational demands, mainly when agents operate at the edge level. At the foundational level, the architecture is grounded in the physical assets previously introduced and components operational on the manufacturing floor.

Ascending from this base, the intermediate layer serves as a conduit, translating orders from the higher-level MAS into actionable control commands for the hardware. The higher-level layer is constituted by a MAS, engineered to abstract and manage the physical resources below. This system plays a pivotal role in monitoring the production line's status, including identifying active stations, their available tasks, and the real-time tracking of product production and location. This oversight allows the MAS to optimise system execution and production efficiency. The MAS is designed into four generic agents, each contributing to the system's overall functionality:

1) ***Product Agent (PA):*** The PA represents a physical product and navigates the manufacturing process. It orchestrates executing necessary tasks by interacting with other system agents, ensuring that each product's requirements are met efficiently. The PA's role extends to task negotiation and coordination with cyber-physical agents, culminating in the logistical management of product movement across the shop floor.

*2) **Resource Agent (RA):*** The RA's function is to virtualise physical assets that contribute to product development. RAs are essential for matching the dynamic requirements posed by PAs, offering their capabilities as skills. This ensures that at any given moment, the specific needs of a product are met by leveraging the appropriate resources.

*3) **Transport Agent (TA):*** Specializing in abstracting the system's logistical components, such as conveyor belts, TAs are pivotal in physically relocating products between different stations. Their integration ensures seamless connectivity between RAs and the physical infrastructure of the manufacturing floor.

*4) **Deployment Agent (DA):*** It oversees the manufacturing system's dynamic landscape, managing agents' deployment and retraction in response to changes in the physical resource pool. It ensures the system's adaptability by facilitating the seamless integration or removal of resources.

C. Digital Twin

The first step in creating the DT is the creation of 3D models representing the different physical assets of the system. For that, it was used Blender. Blender is a versatile and powerful 3D creation tool. Its comprehensive suite includes advanced modelling, animation, simulation, and rendering features, enabling the creation of detailed and dynamic digital replicas of physical assets. The software's physics engines and rendering options assist in visualising and simulating real-world phenomena, which are essential for creating an effective DT.

Compared to other tools, Blender stands out for its cost-effectiveness and accessibility. Unlike proprietary software such as Autodesk Maya and Siemens NX, Blender is free, reducing financial barriers for users and the adoption of solutions. While Maya excels in high-level animation and NX in engineering design, Blender offers a more user-friendly and flexible solution for various DT applications. Additionally, Blender's active community ensures continuous improvements and support.

Fig. 4 and Fig. 5 show the 3D models of the conveyor belt and the station created using Blender for the proposed DT.

After creating all the physical assets' 3D models, it is time to create the virtual environment using Unity. Unity is a leading cross-platform game development engine and interactive content creation tool developers use to create immersive experiences across various platforms, including desktop, mobile, console, and augmented reality/virtual reality (AR/VR) devices.

During execution, the DT always represents the current state of the MCPPS and the operating physical assets. In this case, presented in Fig. 6, all the agents are launched and ready to work.

The versatility and scalability of Unity make it a good candidate for creating games, interactive simulations, architectural visualisations, training applications, and more. Similarly to Blender, Unity's community is very active.

Fig. 6 shows the virtual environment in Unity with a configuration containing all the physical assets represented. Hence, the robot plugged in one conveyor (Conveyor F), and the operator worked on the opposite (Conveyor B).

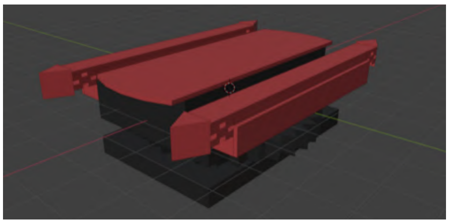

Fig. 4 – Conveyor Belt 3D Model

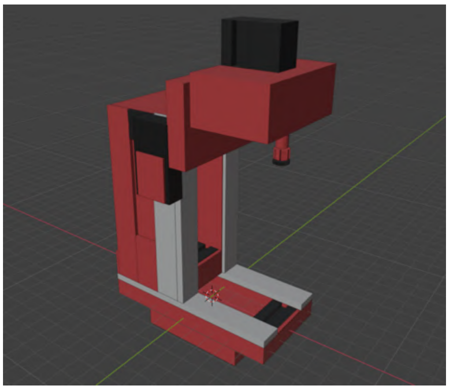

Fig. 5 – Station 3D Model

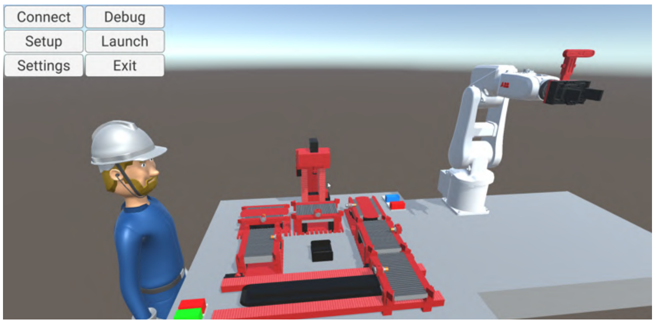

Fig. 6 – Virtual Representation in Unity with the Educational Kit (Station and Conveyor Belts), Human Operator, and Robot.

Fig. 7 represents the possible interactions between the system user, the DT, the MCPPS, and the physical assets. These interactions may focus on adding a new asset to the system, reconfiguring the existing system and its respective assets, initiating a new product launch, or removing an asset. When a new asset is added to the system, it wakes up and notifies the DT that it is connected. The MCPPS detects that it is active and launches a virtual representation by launching an RA. Whenever users want to configure an asset attached to the system, they must use the DT to specify which capabilities are provided by that asset. When this configuration is triggered, the DT communicates the new configuration to the MCPPS, which will reconfigure the specific RA. Similarly, when the user wants to launch a new product, they define the process (a list of skills to be executed), and the DT sends these specifications to the Modular CPPS. A new PA is launched to execute the process, which abstracts the product to be produced. The last interaction focuses on removing an asset from the system. The removal is done automatically; whenever an asset is removed, the DT receives a notification to update the digital environment, and the RA that abstracts the asset is also removed from the Modular CPPS.

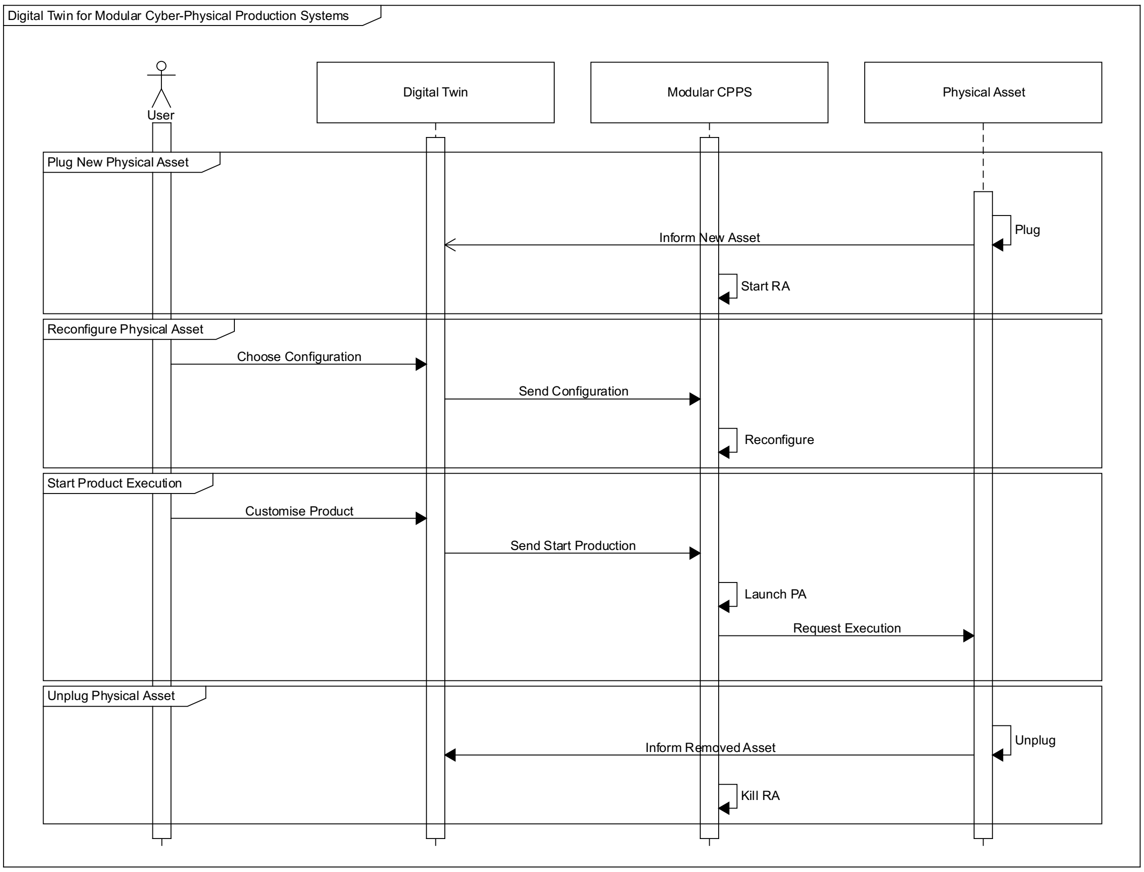

Fig. 7 – Digital Twin Framework Sequence Diagram

The respective agent dies when the operator and the robot leave the system. The same notification is sent by the controller (robot) or the mobile app (operator) to the DT (Unity) informing that the asset is no longer plugged. Hence, the DT updates the virtual representation, as shown in Fig. 8.

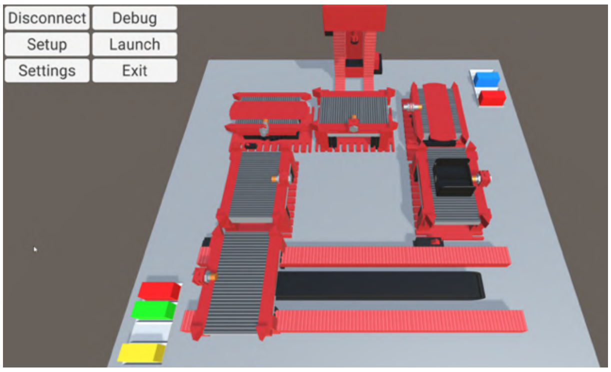

Fig. 8 – Virtual Representation in Unity with the Educational Kit (Station and Conveyor Belts) only.

IV. Tests and Validation

A set of tests were conducted to test the developed solution, where various combinations of connected assets and product-requested skills were specified. The product comprises a carrier, and it is possible to add two colour pieces (red, blue, green, or yellow), one close to the other.

In addition to the colour pieces, the system also includes two skills that can be made available at this educational kit station, called Screw and Drill, which in this case, do not affect the product, but only activate the station in different ways. The robot or the operator can place the colour pieces on their respective conveyors where they are connected. After the product is routed to the position where they are, the product can request the RA, which abstracts the physical asset, to perform the desired skill.

During the system execution, the DT will always have a faithful representation of the assets available in the physical layer and, respectively, in the MCPPS. When these are connected, it is possible in the DT to configure which colours each asset can provide at any given moment. For example, the operator can only add blue or green pieces, and the robot can add yellow or red. This configuration is sent to the MCPPS, which will control the system according to the settings specified by the DT user. The product used in the demonstration combines a carrier and two-colour pieces that are added to this same carrier. In Fig. 9, it is possible to see how the user configures the available colours by selecting them.

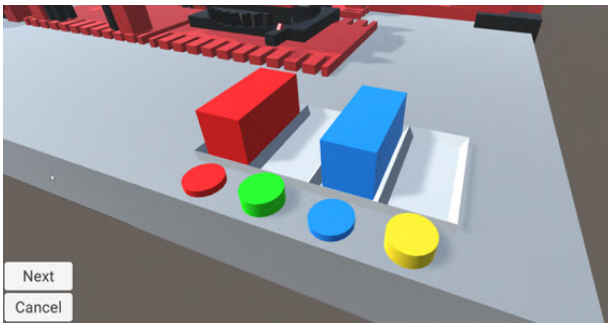

Fig. 9 – Configuration of the Available Pieces (Skills) in a Specific Station.

A. *Trigger an execution*

After the configuration phase, and whenever the user desires to launch and configure one product, the DT user specifies which colours to add to the product and on which side each piece should be placed. Moreover, the operator can add the skills offered by the station and specify the skills' order to be executed, as shown in Fig. 10. To configure the production process, the user drags and drops the skills in the desired order.

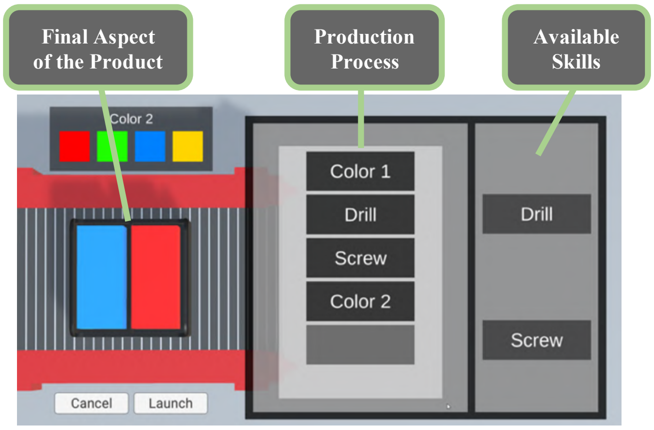

Fig. 10 – Product Configuration View.

B. *Reconfiguration process*

The process of plugging a physical asset into the developed environment was extensively tested, and as described, it begins with the physical plug of the hardware. In the case of the robot, the two stations where the plug can occur have a presence sensor, which, whenever activated, launches an RA to abstract the robot and communicate the presence of the robot to the DT. In the case of the human operator, when an operator would start working at one of the available stations, they would select the station where they would begin operating on the HMI, as shown in Fig. 11. As shown in Fig. 6, the operator can work on the left or the right side of the system.

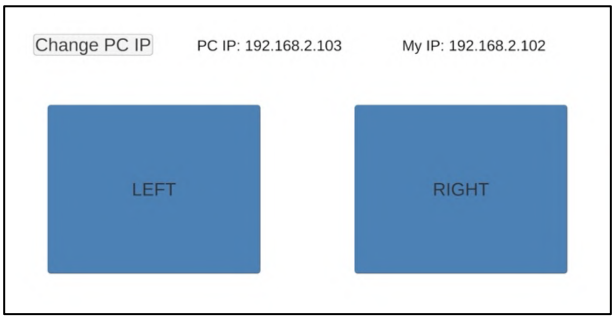

Fig. 11 – HMI Ready for the Operator to Select the Station to Start Working.

As previously explained, after the plug, the colours the new asset can place on the product were configured through the configuration buttons on the DT. Then, when the product was launched using the configurator shown earlier, it was routed to the operator or robot at each process step, as expected. In the case of the robot, it added the piece autonomously, and in the case of the operator, whenever a piece was requested, they received a notification on the HMI, as shown in Fig. 12.

V. Conclusion

In conclusion, this research paper addresses the imperative need for manufacturing systems to adapt to the evolution of Industry 4.0, characterised by the growing demand for customised products and flexible production methodologies. The emergence of CPPS as a crucial innovation highlights the industry's efforts to enhance functionality and manage complex tasks effectively through virtual representations of production resources.

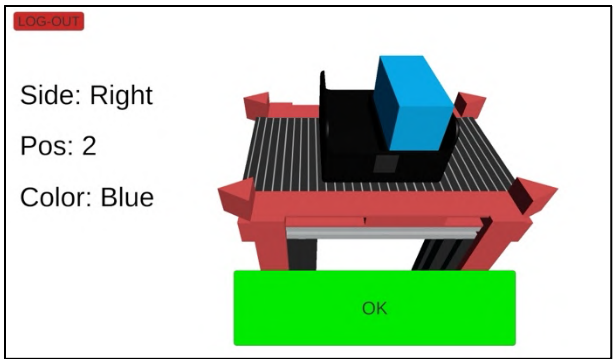

Fig. 12 – HMI Whenever a Task Needs to be Performed by the Operator.

Leveraging MAS within distributed CPPS has shown promise in fostering collaborative manufacturing environments, improving efficiency, and enabling adaptability to market fluctuations and disruptions.

The proposed DT framework serves as a top layer, providing a logical representation of the current state of the physical system and the MCPPS responsible for controlling and optimising execution. Through three layers - Physical Assets, MCPPS, and DT - the framework offers a comprehensive approach to modelling and managing manufacturing systems in a dynamic and reconfigurable environment.

Revisiting the research question and hypothesis proposed at the beginning of the article, it is now possible to state that through the proposed solution, the operator can better understand what is happening in the system, due to the visual component of the factory environment. Additionally, the operator can have some control (at a macro level) of the system through the reconfiguration of each module, setting certain boundaries for system execution. In this way, the authors believe that these measures can indeed increase human confidence in using such systems, thereby enhancing their adoption in real environments.

The engineering process involves studying the physical layer, designing and implementing the MCPPS, and creating the DT using tools such as Blender and Unity.

Tests conducted are promising in their effectiveness in handling various combinations of connected assets and product-requested skills. The DT ensures a representation of physical assets and the MCPPS, allowing for configuration and monitoring of the manufacturing process. Despite the complexity, the proposed solution offers a promising approach to addressing control challenges and easing human interaction within manufacturing ecosystems.

Although the work contributes to the design of DTs that can be useful for the adoption of Modular CPPS, especially with Plug-and-Produce capabilities, there are still some limitations. Due to the unpredictable nature of a Modular CPPS, it is not possible to know in advance all the modules that might be connected to the system. Thus, it will be necessary to create a system capable of handling any module, even those not considered at the start of execution, that may be added later.

In addition to this limitation, the rigidity at the transportation system level can also be identified. In the proposed work, the transportation system does not change (e.g., new conveyors or new layouts). If we want to create a flexible environment at this level as well, it is necessary to advance the work in this area.

The future work will necessarily have to focus on resolving these two limitations, that is, ensuring that the environment can receive new modules, even if they were not initially considered, and creating a mechanism that allows for the creation of a transportation layout, which could be, for example, a combination of transportation components (conveyors, mobile robots, or even operators).

Future work can also focus on improving the interfaces and exploring additional communication interfaces to further increase the capabilities and applicability of the framework in various industrial environments, as well as its application in a real production scenario.

ACKNOWLEDGMENT

The authors acknowledge the contributions of the Portuguese FCT-Strategic program UIDB/00066/2020.

M.A. thanks the Fundação para a Ciência e Tecnologia (FCT), Portugal, for the Ph.D. Grant .

REFERENCES

[1] L. M. Camarinha-Matos, A. D. Rocha, and P. Graça, 'Collaborative approaches in sustainable and resilient manufacturing', *Journal of Intelligent Manufacturing 2022*, pp. 1–21, Dec. 2022, doi: 10.1007/S10845-022-02060-6.

[2] L. Monostori *et al.*, 'Cyber-physical systems in manufacturing', *CIRP Annals*, vol. 65, no. 2, pp. 621–641, Jan. 2016, doi: 10.1016/J.CIRP.2016.06.005.

[3] L. Ribeiro, 'The Design, Deployment, and Assessment of Industrial Agent Systems', *Industrial Agents: Emerging Applications of Software Agents in Industry*, pp. 45–63, Jan. 2015, doi: 10.1016/B978-0-12-800341-1.00003-6.

[4] P. Leitao, S. Karnouskos, T. I. Strasser, X. Jia, J. Lee, and A. W. Colombo, 'Alignment of the IEEE Industrial Agents Recommended Practice Standard With the Reference Architectures RAMI4.0, IIRA, and SGAM', *IEEE Open Journal of the Industrial Electronics Society*, vol. 4, pp. 98–111, 2023, doi: 10.1109/OJIES.2023.3262549.

[5] L. Ribeiro, A. Rocha, A. Veiga, and J. Barata, 'Collaborative routing of products using a self-organizing mechatronic agent framework—A simulation study', *Comput Ind*, vol. 68, pp. 27–39, Apr. 2015, doi: 10.1016/J.COMPIND.2014.12.003.

[6] J. Barbosa, P. Leitão, E. Adam, and D. Trentesaux, 'Behavioural validation of the ADACOR2 self-organized holonic multi-agent manufacturing system', *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, vol. 9266, pp. 59–70, 2015, doi: 10.1007/978-3-319-22867-9_6/COVER.

[7] L. Ribeiro, J. Barata, M. Onori, C. Hanisch, J. Hoos, and R. Rosa, 'Self-organization in automation - The IDEAS pre-demonstrator', *IECON Proceedings (Industrial Electronics Conference)*, pp. 2752–2757, 2011, doi: 10.1109/IECON.2011.6119747.

[8] J. Dias-Ferreira, L. Ribeiro, H. Akillioglu, P. Neves, and M. Onori, 'BIOSOARM: a bio-inspired self-organising architecture for manufacturing cyber-physical shopfloors', *J Intell Manuf*, vol. 29, no. 7, pp. 1659–1682, Oct. 2018, doi: 10.1007/S10845-016-1258-2/FIGURES/10.

[9] P. Leitão, N. Rodrigues, J. Barbosa, C. Turrin, and A. Pagani, 'Intelligent products: The grace experience', *Control Eng Pract*, vol. 42, pp. 95–105, Sep. 2015, doi: 10.1016/J.CONENGPRAC.2015.05.001.

[10] P. Leitão, A. W. Colombo, and S. Karnouskos, 'Industrial automation based on cyber-physical systems technologies: Prototype implementations and challenges', *Comput Ind*, vol. 81, pp. 11–25, Sep. 2016, doi: 10.1016/J.COMPIND.2015.08.004.

[11] A. Rocha *et al.*, 'An agent based framework to support plug and produce', *Proceedings - 2014 12th IEEE International Conference on Industrial Informatics, INDIN 2014*, pp. 504–510, Nov. 2014, doi: 10.1109/INDIN.2014.6945565.

[12] N. Antzoulatos, E. Castro, L. de Silva, A. D. Rocha, S. Ratchev, and J. Barata, 'A multi-agent framework for capability-based reconfiguration of industrial assembly systems', *Int J Prod Res*, vol. 55, no. 10, pp. 2950–2960, May 2017, doi: 10.1080/00207543.2016.1243268.

[13] P. Danny, P. Ferreira, N. Lohse, and K. Dorofeev, 'An Event-Based AutomationML Model for the Process Execution of Plug-and-Produce' Assembly Systems', *Proceedings - IEEE 16th International Conference on Industrial Informatics, INDIN 2018*, pp. 49–54, Sep. 2018, doi: 10.1109/INDIN.2018.8471955.

[14] K. Dorofeev, S. Profanter, J. Cabral, P. Ferreira, and A. Zoitl, 'Agile Operational Behavior for the Control-Level Devices in Plug&Produce Production Environments', *IEEE International Conference on Emerging Technologies and Factory Automation, ETFA*, vol. 2019-September, pp. 49–56, Sep. 2019, doi: 10.1109/ETFA.2019.8869208.

[15] P. Leitão, 'Agent-based distributed manufacturing control: A state-of-the-art survey', *Eng Appl Artif Intell*, vol. 22, no. 7, pp. 979–991, Oct. 2009, doi: 10.1016/J.ENGAPPAI.2008.09.005.

[16] M. Palma da Silva Viola Parreira Rocha, D. Rogério Salema Puga Leal Arguente, D. José Barata de Oliveira Vogal, and D. António Carlos Bárbara Grilo, 'Risk of employing an evolvable production system', 2016, Accessed: Feb. 16, 2024. [Online]. Available: https://run.unl.pt/handle/10362/16431

[17] G. R. Rolo, A. D. Rocha, J. Tripa, and J. Barata, 'Application of a Simulation-Based Digital Twin for Predicting Distributed Manufacturing Control System Performance', *Applied Sciences 2021, Vol. 11, Page 2202*, vol. 11, no. 5, p. 2202, Mar. 2021, doi: 10.3390/APP11052202.

[18] J. Lee, B. Bagheri, and H. A. Kao, 'A Cyber-Physical Systems architecture for Industry 4.0-based manufacturing systems', *Manuf Lett*, vol. 3, pp. 18–23, Jan. 2015, doi: 10.1016/J.MFGLET.2014.12.001.

[19] C. Zhuang, J. Liu, and H. Xiong, 'Digital twin-based smart production management and control framework for the complex product assembly shop-floor', *International Journal of Advanced Manufacturing Technology*, vol. 96, no. 1–4, pp. 1149–1163, Apr. 2018, doi: 10.1007/S00170-018-1617-6/METRICS.

[20] A. Di. Rocha, J. Tripa, D. Alemao, R. S. Peres, and J. Barata, 'Agent-based plug and produce cyber-physical production system - Test case', *IEEE International Conference on Industrial Informatics (INDIN)*, vol. 2019-July, pp. 1545–1551, Jul. 2019, doi: 10.1109/INDIN41052.2019.8972169.