

Digital Twin Experiments Focusing Virtualisation, Connectivity and Real-time Monitoring

Flávia Pires^{*†}, Victória Melo[‡], João Almeida[‡], Paulo Leitão^{*}

^{*} Research Centre in Digitalization and Intelligent Robotics (CeDRI), Instituto Politécnico de Bragança, Campus de Santa Apolónia, 5300-253 Bragança, Portugal

Email: {fpires, pleitao}@ipb.pt

[†] Faculty of Engineering - University of Porto, Rua Dr. Roberto Frias, 4200-465 Porto, Portugal

[‡] Federal University of Technology – Parana (UTFPR)

Email: victoriac.a.melo25@gmail.com, almeidaj@alunos.utfpr.edu.br

Abstract—Industry 4.0 is re-shaping the manufacturing world, and amongst the several associated emerging methods and technologies, Digital Twin is becoming a popular approach both in industry and academia. However, the lack of knowledge about the characteristics, functionalities, best practices and benefits that it can provide, especially for small and medium enterprises, constraints its wider adoption. The use of real applications and demonstrators can contribute to exposing stakeholders to these new and innovative technologies and approaches, showing the applicability of Digital Twins. This paper presents several experiments in implementing Digital Twin for automation scenarios, considering different technologies and functionalities, namely in terms of virtualisation, connectivity and monitoring. Lessons learnt and challenges are also provided as a result of the experimental implementations.

Keywords: Digital Twin, Virtualisation, Connectivity, Monitoring.

I. INTRODUCTION

The introduction of Industry 4.0 [1] has been embraced by several countries around the world through the development and implementation of different initiatives, such as "Industrie 4.0" in Germany, "Society 5.0" in Japan, "Industrie du Futur" in France, "Industria Conectada 4.0" in Spain, "Made in China 2025" in China, "Smart Industry" in the Netherlands and "Industrial Internet of Things" in USA [2]. These programs are driving the manufacturing world towards the adoption of emerging technologies, e.g., Internet of Things (IoT), Big data, artificial intelligence, virtual reality and collaborative robots, under the Cyber-Physical Systems (CPS) technological umbrella. Also in this digital transformation, the Digital Twin emerges as one of the most promising and innovative technologies for solving key challenges in the development of a smart manufacturing environment [3].

The Digital Twin importance has been growing in both academia and industry fields. Figure 1 illustrates the evolution of the interest in the Digital Twin concept along the years, taking into consideration the scientific papers retrieved from the Scopus repository that contains the "Digital Twin" term in the title, abstract and keywords [4]. The analysis of the curve allows concluding the growing of the number of publications related to Digital Twin along the decade, with a particularly significant improvement in the last three years.

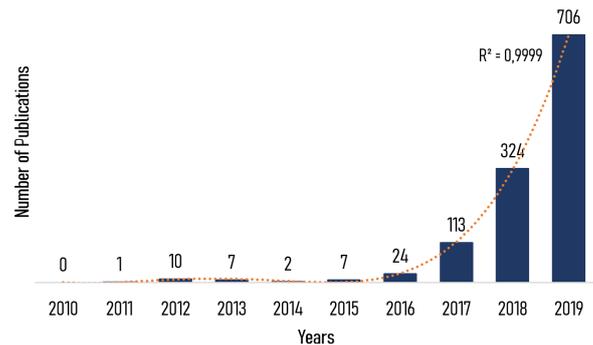


Figure 1: Evolution of the scientific interest in Digital Twin.

On the other hand, several companies are starting to use Digital Twin to improve their businesses, e.g., General Electric, Siemens, IBM, CISCO and Bosch [5]. This growing interest can work as a cascade effect for the small and medium enterprises (SME) to be aware of the potential benefits that the Digital Twin can bring to their business.

Despite the growing popularity of the Digital Twin, there is a lack of knowledge about the characteristics, functionalities and benefits that it can provide, especially for SMEs, that constraints its wider adoption. The use of real implementations as demonstrators is one suitable way to expose stakeholders to these new and innovative technologies and approaches, showing the applicability, benefits and best practices of implementing Digital Twin solutions. According to [6], the transfer of knowledge and benefits should be made through the demonstration platforms with real examples of implementation as is the case of learning factories. This could contribute to a faster adoption by the industry sector and boost the research being done in the field.

Having this in mind, this paper provides several experiments in implementing Digital Twin for automation scenarios, considering different technologies and functionalities, namely in terms of virtualisation, connectivity and monitoring. An analysis of the lessons learned during the experimental implementation is also provided.

The rest of the paper is organised as follows. Section II provides an overview of the Digital Twin concept, application areas, benefits and challenges. Section III presents experiments

related to implementing Digital Twin focusing the virtualisation and connectivity, and Section IV presents an experiment related to the real-time monitoring. Section V summarises the challenges and lessons learned during the development of the experiments. Finally, Section VI rounds up the paper with the conclusion and points out the future work.

II. A BRIEF OVERVIEW OF DIGITAL TWIN

The origins behind the concept of the Digital Twin were associated to Michael Grieves [7], which proposed in 2002 the concept for the creation of a Product Lifecycle Management (PLM) based on the constituent elements of today's Digital Twin, as is the real world, the virtual world and their interconnection to allow the data flow between the two worlds [7]. Lately on 2010, NASA (National Aeronautics and Space Administration) defined the Digital Twin as *"an integrated multi-physics, multi-scale, probabilistic simulation of a vehicle or system that uses the best available physical models, sensor updates, fleet history, etc., to mirror the life of its flying twin. It is ultra-realistic and may consider one or more important and interdependent vehicle systems"* [8]. From this point on, the concept has evolved and transposed the barriers of the aerospace into the manufacturing sector.

Taking into consideration the different definitions that can be found in the literature, it is possible to establish a general definition of Digital Twin as *"the digital copy of a physical object or system, that is connected and shares functional and/or operational data"* [9]. The collected data posteriorly supports the execution of simulation and data analytics allowing the optimisation and enhancement on the physical object [9].

The development of a general Digital Twin can be linked to an architectural model, e.g., illustrated in Figure 2, that comprises five phases, namely virtualisation, connectivity, monitoring, decision support and control.

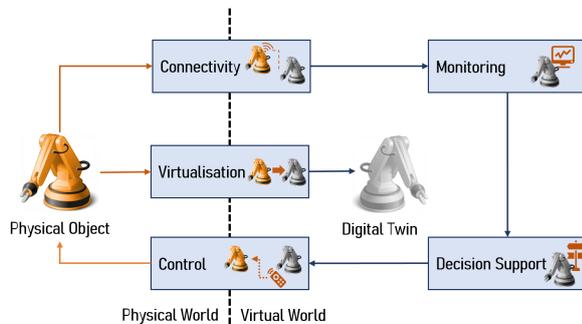


Figure 2: Architectural model for Digital Twin applications.

The first phase comprehends the virtualisation of physical object or system, e.g., a product, a machine or a process. The creation of the virtual model can follow various approaches, e.g., the development of CAD models, taking in consideration that the models will describe the physical object, supporting the execution of simulation and data analysis. The second phase comprehends the establishment of the communication between the physical part and the Digital Twin, and vice-versa, allowing the real-time data collection to feed the virtual

model. For this purpose, several industrial communication protocols are usually used, clustered according to the industrial networks categories, namely Fieldbus networks (e.g., PROFIBUS and Modbus), Ethernet-based networks and Wireless networks (e.g., wifi, Bluetooth and Zigbee) [10]. The third phase comprises the implementation of real-time monitoring mechanisms through the use of, e.g., statistical analysis or control rules, immersed in dashboard software applications, e.g., the Node-RED. The fourth phase includes the development of the decision support capability, which may include the data acquisition and cleaning, the data storage, the data processing and the domain knowledge, complemented with the use of simulation platforms. These tasks are supported by the application of artificial intelligence algorithms to produce knowledge, prediction, optimisation and options that are sent back to the physical system. Finally, the last phase is related to the control of operations that can be applied automatically by the Digital Twin into the physical system when the trust level is adequate, being possible to find the use of augmented and virtual reality to support the faster and easier implementation of defined actions. These five phases can also be viewed as different levels of implementation for the Digital Twin, as well as working in a cycle aiming the continuous improvement.

In terms of benefits, Rashed et al. [11] identify eighth benefits when using the Digital Twin technology, namely as real-time remote monitoring and control, higher efficiency and safety, predictive maintenance and scheduling, scenario and risk assessment, better intra and inter-team synergy and collaboration, more efficient and informed decision support system, customisation of products and services, and better documentation and communication.

However, several research challenges still need to be addressed to improve its adoption in an Industry 4.0 environment. Lu et al. [10] refer seven research challenges to be solved in the nearby future, namely the definition of an architecture pattern, the establishment of requirements for the communication latency according to the application, the definition of data collection mechanisms, the establishment of standards to maintain the capabilities of flexibility, interoperability and scalability, the inherent functionalities of the Digital Twin that are being applied in practice, the creation of management models able to be adapted throughout time, and finally, the introduction of humans in Digital Twin applications. This last challenge brings to light two other important challenges, i.e. the integration of the human as an integral part of the production system and the trust of the human in the decisions provided by the Digital Twin [9].

Along with the security issues that arise due to the connection with several endpoints for collecting data, another important challenge is to convince the stakeholders about the potential benefits of adopting this approach, which can be overcome by their exposition to Digital Twins experiments, exploring different technologies and functionalities. Having this in mind, several experiments are reported focusing on the implementation of some Digital Twins phases, namely virtualisation, connectivity and monitoring.

III. EXPERIMENTS RELATED TO VIRTUALISATION AND CONNECTIVITY

In this section, two Digital Twin applications will be presented regarding the virtualisation and connectivity, using different technologies and software applications.

A. Virtualisation of the UR3 Robot

One of the emergent technologies associated to Industry 4.0 is collaborative robotics, where the robot shares the same work-space with humans during the execution of work operations, e.g., assembly tasks. The first case study considers an industrial inspection station capable to analyse console displays, by using an UR3 collaborative robot.

The construction of the Digital Twin for the UR3 robot comprises its virtualisation in the V-REP robotic simulation package that already provides the 3D model of the robot. Figure 3 illustrates the UR3 robot working in the physical world and its virtual model running in the V-REP environment.

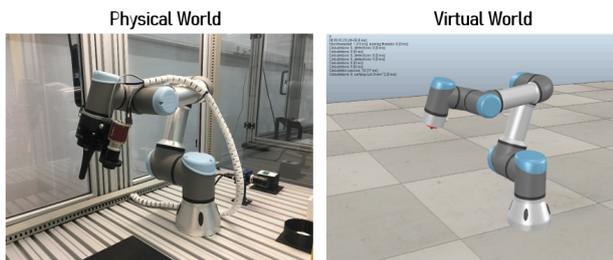


Figure 3: UR3 robot in the real world (left), UR3 virtual model in V-REP (right).

The virtualisation by using a virtual model provided by the simulation software is much faster and easier than having to build a virtual model from scratch. The virtualisation of the workstation structure, developed by using the SolidWorks software is very computational heavy to be run in the V-REP environment and was not considered in this work.

After having the virtual model available in the simulation environment, a crucial issue is to establish the connectivity with the physical asset in order to collect real-time data and feed the virtual model (allowing the visualisation, animation and simulation of the virtual model according to the real execution of the system in the physical world). In this case, the connectivity was established by using the Modbus TCP/IP protocol through the development of a customised Java application (see [9] for more details), which allows connecting remotely to the robot, as long as the computer and the robot are in the same network. This application also performs some conversion tasks between the physical and virtual worlds. The representation of the joints of the real and virtual robot is different, being necessary to make an additional calculation before sending the real data to the virtual model.

B. Virtualisation of a Fischertechniks Cell

The second experiment is related to a Fischertechniks manufacturing system that consists of four cells, called Punching A, Punching B, Indexed A and Indexed B. These cells have

photoelectric sensors and DC motors applied to the conveyors, drills and pushers. The manufacturing system is controlled by a Schneider Modicon M340 PLC, and an ABB IRB1400 robot is used to manipulate the parts around the cells.

Instead of what happened for the virtualisation of the UR3 robot, in this case, a pre-designed virtual model is not available, being necessary to build the virtual model from the scratch and import these models from the simulation environment. For this purpose, the 3D virtual models for each one of the manufacturing cells were designed using the SolidWorks software. After creating the 3D models, these were exported to the V-REP simulation environment using the URDF format. In this case, it was necessary to develop Lua scripts in the V-REP environment to perform the linear displacement of the track blades to simulate the real movement of the conveyors and the movement of the parts along the conveyors synchronised with the physical system. Figure 4 illustrates the real Fischertechniks cells and their virtual models running in the V-REP environment.

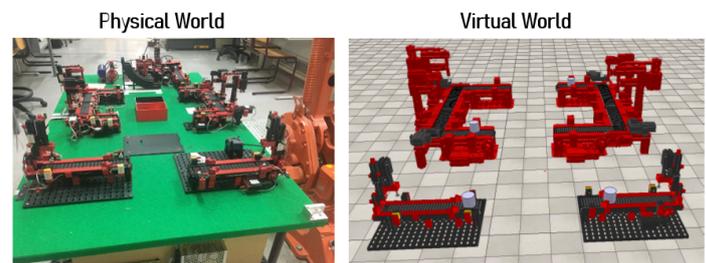


Figure 4: Fischertechniks system in the physical world (left), Fischertechniks virtual model in V-REP (right).

Since the virtual 3D model was built from scratch, the virtualisation task, in this case, was very time consuming, particularly due to the need to measure and model every single component of the manufacturing system. Also challenging was the need to create dependencies between the physical system and the virtual model in order to perform the task in a synchronised manner.

In terms of connectivity, a JAVA data acquisition application was developed, being responsible for the real-time collection of the conveyor system data, namely sensors and actuators, through the use of the Modbus TCP/IP protocol to retrieve data from the Modicon M340 PLC. The communication between this application and the virtual model was performed using the Remote API function available in the V-REP simulation environment, allowing to send the data from the physical system to the created virtual model.

The stated drawbacks represent research opportunities to be followed as future work, in order to improve the operation of the Digital Twin of the UR3 and the Fischertechniks cells.

IV. EXPERIMENTS RELATED TO VIRTUALISATION AND REAL-TIME MONITORING

This section describes a Digital Twin experiment that is performing the dynamic and real-time monitoring of the health condition of a self-organised conveyor transfer system.

A. Description of the Case Study

The third case study is related to a modular and self-organised conveyor system that uses multi-agent systems (MAS) to regulate its operation in a distributed manner. This system comprises a sequence of cyber-physical components, where each physical part consists of a conveyor belt, a DC 24V motor and two photoelectric sensors used to detect parts in the input and output positions of the conveyor belt, while the logical part is performed through an intelligent agent running in a Raspberry Pi, as illustrated in Figure 5.

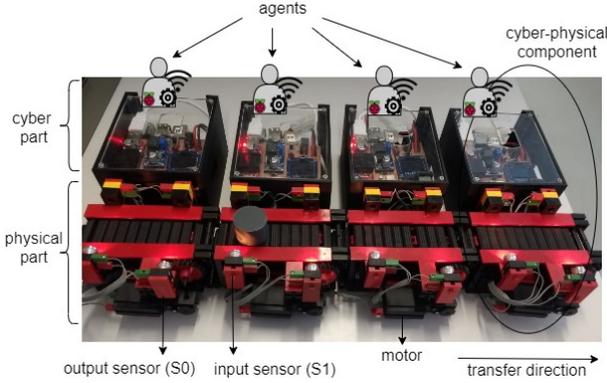


Figure 5: Cyber-physical self-organised conveyor system.

This system aims to transport a piece from a starting point to an ending point through a self-organised set of modular conveyors. The transfer among the conveyors should respect some precedences, e.g., considering the first two conveyor modules, the second conveyor starts its motor when the part arrives at sensor S0, and the first only stops its motor when the part arrives at sensor S1.

The use of MAS with self-organisation capabilities deployed in a CPS for a modular conveyor system allows achieving scalability and the dynamic system reconfiguration, enabling to remove, add or swap conveyor modules on-the-fly, i.e. without the need to restart, stop or reprogram the system. The agent-based system for this conveyor system was developed using the JADE framework [12] and deployed in Raspberry Pi single-board computers, being the communication among them performed through a WiFi network [13].

The Digital Twin for this conveyor system aims to monitor its operation and enables the earlier detection of failures based on the real-time data collection using IoT technologies.

B. Virtualisation and Connectivity

The virtualisation of the described conveyor system uses a Node-Red dashboard application, where the physical system is virtually modelled and represented to allow the dynamic visualisation and monitoring of its condition state.

The data regarding the conveyor system operation, namely the motor and sensors states, the timestamps in the transfer of parts, the vibration on the three-axis, the battery level and the current in the motor, is collected by using the MQTT (Message Queuing Telemetry Transport) protocol, which stands out for

its ease use in IoT solutions and follows the publish-subscribe schema. As illustrated in Figure 6, each conveyor module sends the required data by posting it into specific topics in the MQTT broker, which will forward the received data to the clients that had subscribed that topics, in this case, the Node-Red dashboard application.

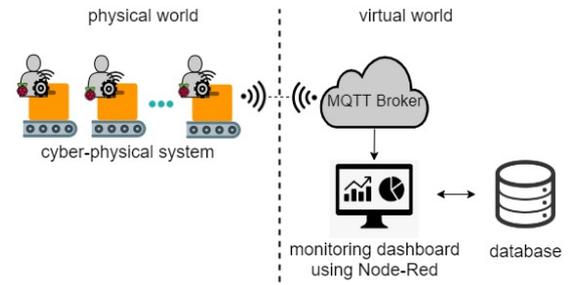


Figure 6: Virtualisation approach for the conveyor system.

The default configuration of the MQTT protocol allows the easy observation of the messages being exchanged by using a sniffer since the IP address of the broker is known. Also, the Node-Red platform is not secured, and if knowing the IP host, it is possible to access the editor and deploy changes. Knowing this security weakness, user authentication was deployed to secure the access to the Node-Red editor and to deny anonymous connections to the broker and signing up allowed clients. The Secure Sockets Layer (SSL) and Transport Layer Security (TLS) protocols were used to encrypt the communications by acquiring keys and certificates, making the Node-Red server to work over HTTPS and ensuring that the broker only receives messages from the real system.

Figure 7 represents the virtual model of the conveyor transfer system using the Node-Red platform, where the collected data, as well as the aggregated statistical information, is visualised in different menus.

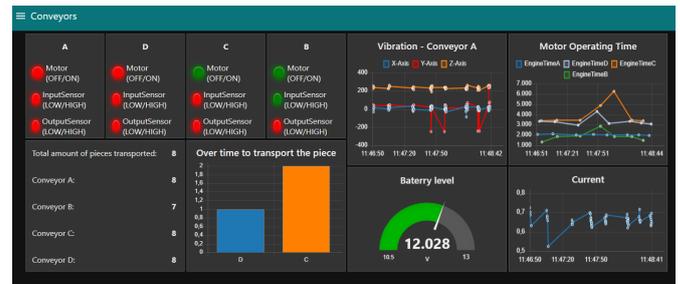


Figure 7: Dashboard for the virtual model of the conveyor system supporting the real-time monitoring.

On the top left of the dashboard there are four displays showing the current status of the conveyor modules, with LEDs indicating if the motor is operating (green colour) or not (red colour), and if the input and output sensors are activated (green colour) or not (red colour). The letters are indicating the sequence of the conveyor modules in the physical conveyor system, being dynamically updated when a change in the sequence of conveyor modules occur.

The data from the accelerometer was collected to monitor the conveyor system condition and its behaviour when it is working, and the values acquired for the X, Y and Z axes are published in the vibration chart. In the same manner, the motor operating time chart shows the time in seconds that the motor of each conveyor module works to transport a piece.

The bottom left presents the total amount of transported parts, regardless of how many conveyors are working, and the total amount of transported parts for each conveyor. If the time to transport the part in a conveyor is higher than the normal, it will be indicated in the bar chart, that shows how many parts had problems in each conveyor during a period of work. The battery level is presented in a graph with defined maximum and minimum values, making easier to perceive when it should be changed so that the system's operation is not affected. The last chart monitors the current values, i.e., how much current is being consumed during the system operation.

C. Real-time Monitoring

The collected data is analysed in real-time to monitor the health condition of the conveyor system and its correct behaviour, and also to allow the detection of abnormal situations, preferably in advance. For this purpose, an alert system was developed to warn risk situations related to the functioning of the conveyor system. Some of these alerts are generated by applying process control methods, such as Nelson rules [14]. These rules use the mean value (\bar{X}) and the standard deviation (σ) to determine if a measured variable is out of control or presents a trend that shows the variable is near to be out of control, as illustrated in Figure 8.

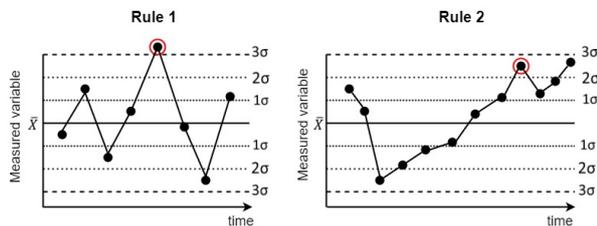


Figure 8: Rules to detect outliers and trends in the conveyor operation behaviour.

Figure 8 presents two rules used to detect situations where the conveyor is in an out of control condition. Rule 1 is used to detect an outlier in the operation of the conveyor and is triggered when a value of the variable being analysed is greater than 3σ . In this way, the average values and standard deviations were calculated for the variables of current, battery, vibration in the z-axis and for the maximum time to transport the part in a conveyor considering the normal system operation. In these cases, the outlier will be detected if the current value exceeds by 12,84% the average value, 18,065% for the acceleration on the z-axis and 32% for the time to transport the piece. However, for the battery, it was observed that when reaching the value of 10.5V (3σ), the voltage supplied was no longer sufficient to maintain the system functioning. Thus, two alerts were created concerning

its level, the first being “Low Battery” when it reaches 11.5V (1σ) and the second “Battery is too low” when it reaches 11V (2σ). When alerts are generated, a pop-up menu appears on the dashboard, indicating which measure is out-of-control.

Rule 2 allows identifying a continuous growth trend in values of the measured variable, i.e. six successive increasing values in a row, allowing to detect in advance possible failures. For the alerts generated from this rule, the variables of current, vibration and motor operating time were used. Whenever a value of one of these variables is added to the database, an analysis is performed with the last six values recorded, checking whether there is an increasing or decreasing trend. If a trend is verified, the alert is generated on the dashboard.

Also, when the virtual system identifies the occurrence of a swap in the physical conveyor system, an alert is displayed and the sequence is updated in the dashboard. The generated alerts are also sent by emails to the system manager, notifying the occurred problem.

V. CHALLENGES AND LESSONS LEARNED

Several challenges and lessons learned regarding the development of a functional Digital Twin have been identified during the development of the previously described experiments.

Firstly, the virtualisation of physical systems is simpler if pre-designed virtual models can be reused or even easily integrated into simulation environments. This was shown in the first case study where the use of the V-REP simulation environment provided a pre-designed 3D model of the UR3 robot. On contrary, the second case study clearly shows that when these virtual models are not available and need to be developed from the scratch, the virtualisation is more complex and time-consuming (with the need to develop the geometric and dynamics models). In this case, despite using the same simulation environment, the V-REP software only provides 3d models for robots and not for the work stations considered in the manufacturing system. However, in both cases, it is necessary to proceed with the implementation of some conversion methods to completely adapt the virtual model to the physical system, e.g., the motion positions. In terms of virtualisation, another important issue is the computational power of the platform running the 3D models, since it strongly affects the visualisation, animation and simulation of the models and provokes delays in the synchronisation with the physical system. This was clearly illustrated in the first case study, where the model for the entire inspection station structure, developed in SolidWorks, was heavy and the computational platform where the virtual models are hosted has not enough computational power to run them properly.

Secondly, in terms of connectivity, the use of standard communication protocols, such as Modbus, strongly simplifies the interoperability between the physical object and the Digital Twin, contributing for a simpler, faster and more transparent development process. This was exemplified in the two first case studies that used the Modbus TCP/IP protocol. In the same manner, the connectivity in the third case study was performed by using the MQTT protocol, which contributes

to developing loosed coupled solutions, enhancing the connectivity between the two worlds. In case of systems using proprietary protocols or old controller systems, the difficulty to implement the connectivity between the physical system and the Digital Twin is complex. This was illustrated in the second case study, where the connection with the ABB IRB1400 robot was very difficult to be implemented due to the old version of the robot controller, which provided limited access to data, i.e. only related to the working state (ON/OFF). In both approaches, cybersecurity issues focusing the data privacy and integrity assumes a crucial role, being necessary to implement simple mechanisms, e.g., authentication and encryption, combined with other more complex mechanisms based on the use of ML algorithms, e.g., to detect attack vectors.

In terms of the real-time monitoring, the collected data can be analysed to supervise the condition operation and to detect anomalies and performance degradation in advance, supporting the diagnosis, prediction and optimisation of the system operation to anticipate possible risks. This was illustrated in the third case study, where a Node-Red dashboard application displays the real-time system data, as well as statistical information, and supports the dynamic monitoring of the condition operation of the conveyor system by generating alerts through the implementation of control rules. At this level, important challenges are related to the difficult to identify the correlation among the measured variables, and to the limitation of some software environments that are appropriate for the virtualisation and simulation but presents weaknesses regarding the visualisation and monitoring processes. This was identified in the first two case studies, where the use of the V-REP software was adequate to create the virtualisation of the robot and to perform the simulation of the virtual model but presents drawbacks regarding the effective and user-friendly visualisation and monitoring of the collected information.

VI. CONCLUSIONS AND FUTURE WORK

Digital Twin is becoming popular in the context of Industry 4.0, allowing the creation of digital copies of physical systems that are connected and share the functional and operational data, allowing the monitoring, diagnosis, prediction and optimisation of the system operation to anticipate possible risks or condition changes. One current challenge is the convincement of stakeholders of the potential and benefits of such an approach, by exposing them to evidence of real implementations.

This paper describes the implementation of the Digital Twin approach in three case studies, using different technologies and focusing different functionalities, namely the virtualisation, connectivity and real-time monitoring. The first case study is related to the virtualisation of a UR3 robot and the second one to the virtualisation of a Fischertechnik's manufacturing system, both using the V-REP software, but the first using a 3D model provided by the software package and the second one using a 3D model developed in SolidWorks. In both situations, the connectivity was performed via the Modbus TCP/IP protocol. The third case considers the virtualisation of

a self-organised conveyor system aiming the real-time monitoring of its health condition, using a Node-Red dashboard application, being the connectivity performed by using the MQTT protocol. This last experiment also considers cybersecurity issues related to the use of IoT technologies during the data collection, suggesting the implementation of simple solutions to prevent potential security threats.

These experiments demonstrated the potentialities and benefits of using the Digital Twin concept, ensuring remote monitoring, predictive maintenance, higher efficiency and security. Future work will be devoted to extending the developed functionalities in these experiments, namely incorporating simulation and data analysis algorithms to support the decision support phase, aiming to improve the system's performance through the optimisation of the operating variables or adaptation of the system behaviour to face the condition changes.

ACKNOWLEDGEMENT

This work has been supported by FCT – Fundação para a Ciência e Tecnologia within the Project Scope UIDB/05757/2020. The author Flávia Pires thanks the Fundação para a Ciência e Tecnologia (FCT), Portugal for the Ph.D. Grant SFRH/BD/143243/2019.

REFERENCES

- [1] H. Kagermann, W. Wahlster, and J. Helbig, "Recommendations for implementing the strategic initiative INDUSTRIE 4.0," *Final report of the Industrie 4.0 WG*, no. April, p. 82, 2013.
- [2] M. Proctor and J. Wilkins, *4.0 sight digital industry around the world*, 2018.
- [3] F. Tao, S. Member, H. Zhang, A. Liu, and A. Y. C. Nee, "Digital Twin in Industry: State-of-the-Art," *IEEE Transactions on Industrial Informatics*, vol. 15, no. 4, pp. 2405–2415, 2019.
- [4] Elsevier, "Scopus," 2019. [Online]. Available: <https://www.elsevier.com/solutions/scopus>
- [5] "Top 10 Digital Twin Vendors for 2019," 2019. [Online]. Available: https://www.em360tech.com/business_agility/tech-features-featuredtech-news/top-10-digital-twin-vendors/
- [6] T. H. Uhlemann, C. Schock, C. Lehmann, S. Freiberger, and R. Steinhilper, "The Digital Twin: Demonstrating the potential of real time data acquisition in production systems," in *Proceedings of the 7th Conference on Learning Factories*, vol. 9, 2017, pp. 113–120.
- [7] M. Grieves and J. Vickers, "Digital Twin: Mitigating Unpredictable, Undesirable Emergent Behavior in Complex Systems," in *Transdisciplinary Perspectives on System Complexity: New Findings and Approaches*, 2017, no. August, pp. 85–113.
- [8] E. Negri, L. Fumagalli, and M. Macchi, "A review of the roles of Digital Twin in CPS-based production systems," *Procedia Manufacturing*, vol. 11, no. June, pp. 939–948, 2017.
- [9] F. Pires, A. Cachada, J. Barbosa, A. P. Moreira, and P. Leitão, "Digital Twin in Industry 4.0 : Technologies , Applications and Challenges," in *Proceedings of the IEEE International Conference on Industrial Informatics (INDIN'19)*, 2019, pp. 721–726.
- [10] Y. Lu, C. Liu, K. I.-K. Wang, H. Huang, and X. Xu, "Digital twin-driven smart manufacturing: Connotation, reference model, applications and research issues," *Robotics and Computer-Integrated Manufacturing*, vol. 61, p. 101837, 2020.
- [11] A. Rasheed, O. San, and T. Kvamsdal, "Digital twin: Values, challenges and enablers," 2019.
- [12] F. Bellifemine, G. Caire, and D. Greenwood, *Developing Multi-Agent Systems with JADE*. Wiley, 2007.
- [13] P. Leitão, J. Barbosa, G. Funchal, and V. Melo, "Self-organized Cyber-Physical Conveyor System using Multi-agent Systems," *Accepted to be published in the International Journal of Artificial Intelligence*, 2020.
- [14] L. S. Nelson, "The Shewhart Control Chart-Tests for Special Causes," *Journal of Quality Technology*, vol. 16, no. 4, pp. 237–239, 1984.