Digital Twin Implementation for Additive Manufacturing Robotic Cell based on ISO 23247 Standard

João V. A. Cabral, Alberto J. Alvares, and Guilherme C. de Carvalho

Abstract—**Recent developments in the field of Additive Manufacturing have been improving the capabilities of the technique not only to be able to build complex geometry parts layer by layer with different materials, but also including the so-called Industry 4.0 technologies, namely Internet of Things (IoT), big data (BD) and Digital Twins (DT). The combination of these technologies with Additive Manufacturing allows online process monitoring and simulation, along with the cloud storage of the process and geometry data collected during the material deposition. The analysis of such data allows online and post-deposition identification of eventual process instabilities that can lead to quality problems. Considering the above-mentioned concepts, this work presents a DT architecture based on the ISO 23247-Digital Twin Framework for Manufacturing standard. In this sense, an approach of a Digital Twin framework for metal additive manufacturing process in a robotic cell composed of a robotic arm, positioning table and welding machine is presented and validated, focusing on the collection and cloud storage of both geometrical and process data along with near real-time process simulation.**

Link to graphical and video abstracts, and to code: https://latamt.ieeer9.org/index.php/transactions/article/view/8923

Index Terms—**Digital Twins, Additive Manufacturing, MQTT, ISO 23247, Industry 4.0.**

I. Introduction

Recently developed technologies of additive Manufacturing (AM) of metal such as metal fused deposition modeling (FDM), wire-arc additive manufacturing (WAAM) and laser metal deposition (LMD) [1]–[3] have enabled new ways of manufacturing complex parts. With those novelties, new ways of monitoring machine and manufacturing parameters during production also came to light. According to [4], AM, big data and Cyber-physical systems (CPS) are some of the pillars that, along with many other technologies, form the basis for the new industrial revolution known as Industry 4.0 [5].

The advent of Industry 4.0 has revolutionized manufacturing processes through the integration of CPSs, Artificial Intelligence (AI), and Digital Twins (DT). Cyber-physical systems are integrations of computational and physical processes, where embedded computers and networks monitor and control physical processes with feedback loops that allow physical processes to influence computations and vice versa. Integrating machine tools with modern supervisory systems facilitates monitoring and observable parameters based action taking. One of the possibilities that came with AI in Industry 4.0 is the processing and analysis of big chunks of data collected from production chains in real-time or near real-time [6].

Regarding applications of Digital Twins (DTs) in the industry, additive and subtractive manufacturing DTs are among the most common ones [7]. In this sense, there are many possibilities of DT development for AM. Those applications can revolve around product lifecycle, predictive maintenance and mirrored 3D simulation. Such developments can improve the efficiency of production, factory floor planning and administration, and can reduce the waste of feedstock and tools; resulting in better usage of resources and reducing costs of production.

Despite significant advancements regarding DTs in Industry 4.0, a notable gap exists in the seamless adaptation of legacy manufacturing systems to modern Industrial Internet of Things (IIoT) standards, particularly in the context of additive manufacturing. This paper addresses this gap by presenting a novel framework for implementing a Digital Twin (DT) in a robotic additive manufacturing cell, based on the ISO 23247 standard. The proposed solution uniquely leverages low-budget hardware and open-source software, providing a cost-effective approach to modernizing existing manufacturing setups. By demonstrating the practical application of DTs for near real-time monitoring, data analysis, and process optimization, this work contributes to the broader dissemination and adoption of DT technologies in various industrial settings.

One of the main differences between a CPS and a DT is that in the latter, actuation in the mirrored physical system is not mandatory [8]. In fact, there are many assets and legacy systems from Industry 3.0 to be adapted for Industry 4.0 using the concept of DTs that may not be suitable for actuation in feasible time due to hardware and/or software limitations. This situation is experienced in the work presented in this article, in which an adaptation of a legacy manufacturing cell for Industry 4.0 was developed.

In this article, the implementation of a Digital Twin for an Additive Manufacturing robotic cell based on WAAM (Wire Arc Additive Manufacturing) is presented. In the GRACO laboratory at the University of Brasilia, an ABB IRB 2600 robotic arm equipped with a Fronius TTW5500 GTAW (Gas Tungsten Arc Welding) torch, composes with a MW5000 power source and an ABB IRBP A250 positioning table a metal AM robotic cell. The work presented in this paper covers

J. V. A. Cabral, A. J. Alvares and G. C. de Carvalho are with Department of Mechanical Engineering, University of Brasilia, Brasilia, Brazil (e-mails: jvarantescabraltrab@gmail.com, alvares@AlvaresTech.com, and gccarval@unb.br).

the development of a MQTT dataflow using Ethernet TCP/IP to collect data from the ICR5 controller of the ABB robot. For the representational part of the DT, a local Node-RED flow with dashboard and history of data published in cloud Firebase Firestore was also developed; finally, RoboDK software is used for local 3D simulation of material deposition and movement mirroring of the robotic arm and positioning table.

The rest of the paper is organized by the following sections: in section II, a brief state-of-art review is presented with examples of similar implementations and DT architectures published previously; section III describes the basis of the DT architecture developed based on ISO 23247; in section IV the results of the dataflow and interfaces developed are detailed; lastly in the final section V the general conclusions of the work are presented with suggestions and thoughts for future applications regarding the DT developed.

II. Literature Review

According to a recent definition published in a technical report by the NIST in Lin et. al. [9], Digital Twins are defined as active agent of digital transformation by enabling digital representations, analysis, process optimization and simulation of products and systems along their lifecycle. In this sense, DTs must be highly interoperable with their real counterparts in order to achieve accurate insights from the monitored systems. Along this section, an overview of the ISO 23247 standard will be presented. Aside of that, a few articles regarding similar Digital Twins applications for additive manufacturing will be presented.

A. Manufacturing Framework for DTs - ISO 23247

The ISO 23247 standard [8], published in 2021, was formulated with the objective of composing guidelines for the development of a framework for developing Digital Twin applications in the context of industrial manufacturing in Industry 4.0. The framework determined by the norm is composed of 4 domains, each representing a different application layer. Each predicted domain provides services and applications to be used by subsequent domains. In order to achieve this, between subsequent domains there is always a communication protocol that enables exchange of data [10]. The four domains listed in the norm are (in order): Observable Manufacturing Element (OME), Data Collection and Device Control Entities (DCDCE), Digital Twin Entity, Digital Twin User Entity.

B. GTAW Metal Deposition Process

Similar to the GMAW (Gas Metal Arc Welding) process, in the GTAW metal deposition process a non-consumable electrode is used , in order to achieve that, both the electrode and the molten pool are protected by an inert gas that comes out of the torch [11]. Gases commonly used in this process are gas argon and helium.

C. Related Works

Throughout the development process of this work, many related works in the literature were consulted and used as an inspiration to the current project. Those works are related to machinetools DT development and DTs applied to both additive and subtractive manufacturing processes. A recent state-of-the-art review [9] highlights dozens of articles and projects related to DT development in Industry 4.0, those applications are related to many fields of parts manufacturing and analysis of processes such as thermal profiling, cloud DT monitoring, AI applied to big data generated by DTs, etc.

Another prominent review article in [12] presents more than a hundred articles focusing on the steps necessary to implement a DT application successfully since the collection of data passing through several interface layers to promote data visualizations and human-machine interfaces. Although some architectures in the cited article have similar attributes to the ones predicted by the ISO 23247 norm, there is no direct citation to the use of the norm on those applications. This demonstrates the need to confirm the applicability of the recent ISO 23247 standard on new DT frameworks that facilitate the repeatability of this type of application.

Cruz et. al. [13] presents an implementation of a Cyber-Physical System based on the IEC 61499 standard applied to a package classification system. In the cited work, a virtual model of the real plant is present to simulate the activation of cylinders.

Kim et. al. [14] proposes a DT architecture based on the ISO 23247 framework standard, the proposed solution is planned to be applied on large-scale AM systems. One of the suggested features is an anomaly detection system based on OPC UA applied to the WAAM process, fulfilling one of the predicted applications of the ISO 23247 in Industry 4.0, quality management applied to the product's and system's lifecycle, without implementation.

Liu et. al. [15] presents a comprehensive data management application applied to DT-enabled AM systems. The collaborative workstation solutions presented in the paper include quality measurement of processes, product design, process planning, manufacturing monitoring, etc. Those solutions may be applied both on local and on cloud/edge edge Digital Twins.

Alvares et al. [16] presents a monitoring and process defining solution related to metal AM. In the paper, a robotic manufacturing cell is implemented by integrating a Meltio LMD nozzle with a Kuka robot. This workstation is then integrated with another solution by the same authors [17] to create a Digital Twin of the observed manufacturing cell using proprietary software Grashopper and Rhino, with 3D simulation in Kuka.Sim and dashboard in Node-RED.

Finally, a previous work at the GRACO laboratory was used as inspiration for the basis of the Digital Twin architecture and general development of the applications shown in this current work. As presented in [10], a Digital Twin architecture also based in the ISO 23247 was developed for a Haas machining center. In this previous work, MQTT and MTConnect dataflows were established, being the MQTT flow the main development, containing also local and cloud dashboards and 3D movement simulations. History of data was also present in an IBM Cloudant database.

When comparing the present work to other related works, it is still evident that there are not many implemented DT

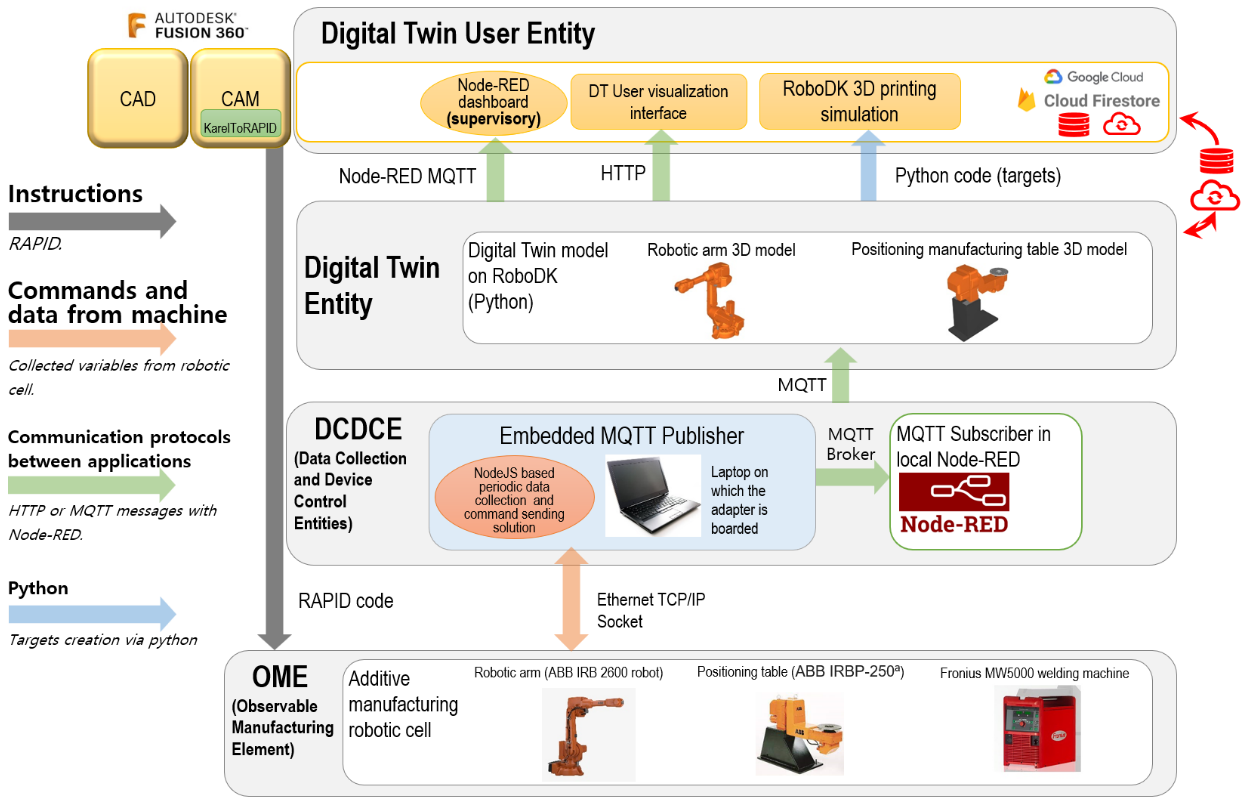


Fig. 1. Digital Twin framework base architecture based on ISO 23247.

frameworks using the norm ISO 23247 as a standard for the development of DT domains [9], [18], [19], especially when considering applications related to additive manufacturing.

III. Digital Twin Architecture Based on ISO 23247

The proposed framework for the Digital Twin development present in this paper is shown in figure 1. In the image, all the domains provided by the ISO 23247 standard are present. All of those work synchronously to provide services and applications to the neighboring domains. The applications provided by the architecture include dashboard of parameters visualization, 3D simulation of mirrored movement of real assets and also history of variables stored in cloud database. Regarding the domains, in the following subsections each one of them is detailed relating to the architecture and planned development.

A. Observable Manufacturing Element

As evidenced in the first domain in figure 1, the OME in the DT implementation corresponds to the assets to be monitored in the lab, in this case the robotic arm ABB IRB 2600, the positioning table ABB IRBP A250 and the Fronius MW5000 GTAW power source that is integrated to the robotic arm. In order to collect data from the robot controller ABB ICR5, an Ethernet cable was used to link the port already installed in the controller to a personal computer (laptop).

The ICR5 controller present in the lab has a RobotWare (RW) 5.13.0 operational system. Because of limitations related to this legacy operational system version, a recent solution related to data collection developed by ABB called Robot Web Services (RWS) wasn't available since it requires RW 6.0 or higher. To overcome this limitation, another solution present in ABB manuals is socket messaging, which was used as a basis for information exchange and parameter extraction.

Starting from an Ethernet TCP/IP socket messaging archetype, a client socket was established in RAPID procedures and code snippets to be interpreted by the ICR5 controller. Parallel to that, a server TCP/IP socket was implemented in a JavaScript (JS) module, being run on a local NodeJS runtime environment.

To collect the parameters from the robotic arm and positioning table, built-in functions in the RAPID code were used. Regarding the signals from the Fronius MW5000 welding machine, a LocalNet to DeviceNet converter was used in order to monitor those, parameters gathered include welding current, voltage, wire feeding speed and welding electric arc status.

B. Data Collection and Device Control Entities - DCDCE

The DCDCE domain corresponds to the basis for the MQTT dataflow implemented, it is crucial that the MQTT adapter works synchronously with other modules aiming the correct operation of the entire DT architecture. Being the bridge between the OME and the Digital Twin Entity, is in this layer that the collected data begins to be organized.

As established by the MQTT standard [20], the Publisher-Subscriber messaging pattern is implemented as a service to be used by the following domains. In this work, three modules were developed in JS corresponding to three steps of data transmition based on MQTT:

TABLE I
COLLECTED VARIABLES IN RAPID CODE AND CORRESPONDING MQTT TOPICS

Variable	Collection method	Variable description	Corresponding MQTT topic
Status	On initializing the client socket module in a RAPID program, the socket sends to the adapter's server socket via TCP/IP a "Status;ONLINE" message; when ending the socket module, the message "Status;OFFLINE" is sent.	Describes if the adapter is online and connected to the ICR5 controller.	"abb/irb2600/status"
Mode	When the client socket is active, in each cycle of messages sent, the controller sends a TCP/IP message with the content "auto", since the client socket module only works in the automatic execution mode of the controller.	Actual execution mode of the ICR5 controller (automatic).	"abb/irb2600/mode"
Execution	Name of the program defined in the RAPID code.	Name of the program being executed.	"abb/irb2600/execution"
Tool	RAPID function CTool() which returns the spatial and inertial parameters of the actual tool.	Describes the actual working tool: displacement between the coupler and the TCP, orientation and inertial parameters if necessary.	"abb/irb2600/tool"
Cwobj	RAPID function CWobj() which returns the positional variables of the current work object.	"Current work object", concerns the actual variables of the workstation (local positional reference), which are used by the robot's controller to calculate the global movement variables.	"abb/irb2600/cwobj"
Speed	An output signal in the ICR5 controller was defined to obtain the system output variable "soTCPSpeed".	Actual TCP linear speed.	"abb/irb2600/speed"
Mspeed	RAPID function MaxRobSpeed().	Maximum linear velocity of the robot in mm/s.	"abb/irb2600/mspeed"
RSpeed	Obtained through the RAPID function CSpeedOverride().	Instantaneous positioning table's rotation speed in percentage.	"abb/irb2600/rspeed"
Welding	Obtained through the Fronius welding machine I/O signal through the ICR5 controller DeviceNet adapter, digital output signal "doFr1ArcOn".	Defines the values "TRUE" or "FALSE" for the instantatneous state of the deposition process, in terms of present or absent electric arc.	"abb/irb2600/welding"
WVoltage	Obtained through the I/O monitoring signal of the Fronius machine by the ICR5 controller "aiFr1WeldingVoltage".	Actual welding voltage in volts.	"abb/irb2600/wvoltage"
WCurrent	Obtained through the I/O monitoring signal of the Fronius machine by the ICR5 controller "aoFr1Power".	Instantaneous welding current in amperes.	"abb/irb2600/wcurrent"
WSpeed	Obtained through the I/O monitoring signal of the Fronius machine by the ICR5 controller "aoFr1WireSpeedWfi".	Current wire feed speed in mm/s.	"abb/irb2600/wspeed"
PosRotation	Instantaneous rotation of external axes of the robot obtained through the "extax" parameters, which are obtained through the return of the built-in RAPID function CRobT() ("current robot target").	Rotation of the external axes of the robot in degrees (rotation of the positiong table's "arm" and plate).	"abb/irb2600/posrotation"
CRobT	Translational ("trans") and rotational ("rot") attributes of the actual "target" of the robot obtained through RAPID function CRobT().	Actual position variables ("target") of the robot, Cartesian coordinates and orientation of the TCP in quaternion notation.	"abb/irb2600/crobt" "abb/irb2600/x_pos" "abb/irb2600/y_pos" "abb/irb2600/z_pos" "abb/irb2600/rorient"
Timestamp	A timestamp used for reference generated in the JavaScript code and published by the MQTT publisher.	Instantaneous timestamp for reference of the end of the current cycle of variables transmission by the controller.	"abb/irb2600/timestamp"

1) Ethernet module: in this module a Ethernet TCP/IP server socket is implemented in order to receive data from the ICR5 controller via ethernet cable. The data transmission occurs based on handshakes, in which the server confirms each message received to the TCP/IP client socket in the ICR5 controller in order to signalize the availability of receiving the next message;
2) Broker module: the broker module creates a TCP/IP broker with Aedes handler in order to pass along messages published by the publisher reaching the subscribers of each topic;
3) Publisher module: the publisher module initiates instances of the first two modules, using the ethernet module to collect data via socket TCP/IP from the ICR5 controller and also using the broker to deliver messages to the subscribers of each topic. In each cycle of data collection, messages related to each topic are published.

It is in the publisher module that the data collected from the controller in messages are divided in topics, each parameter's information is processed and then assigned to a topic. The relation between the parameters, RAPID functions and topics are detailed in section IV.

C. Digital Twin Entity

In the Digital Twin Entity there are two MQTT flows developed, each corresponding to a service used by an application of the following domain, the DT User Entity. In those flows, data is structured and processed to enable data visualization, 3D movement simulation and data storage. It is important to notice that both flows make use of the MQTT adapter (publisher, broker and ethernet adapter) present in the DCDCE domain.

The first and main flow makes use of the Node-RED framework in order to organize and process messages received by subscribers of the topics published by the MQTT publisher module. The messages are then united in a single JSON format document to be stored in a cloud Firebase Firestore database. Documents stored in the cloud database serves as a history of collected variables from the AM robotic cell and may be used to generate graphs and to analyse processes.

The second developed flow serves as a basis for the 3D simulation of mirrored movement of the real robotic cell. In this flow the 3D models of the ABB IRB 2600 robot, ABB IRBP A250 positioning table and GTAW torch are present in order to enable the 3D view of the movement simulation of those components in the Digital Twin User Entity domain.

D. Digital Twin User Entity

Finally in the Digital Twin User Entity domain are the Human-Machine Interfaces (HMI) that make use of the resources of the services made available by the Digital Twin Entity and DCDCE domains. Those HMIs correspond to the users of the dataflows based off the DCDCE MQTT adapter.

The HMI applications include near real-time parameters visualization through Node-RED dashboard and RoboDK 3D simulation of real assets digital models. Both applications will be further explained in the following section.

IV. Results of Implemented DT Architecture

As stated in section III, the experimental setup of the Digital Twin testing is composed by the ABB IRB 2600 robotic arm, the ABB IRBP A250 positioning table and the Fronius MW5000 welding machine with GTAW torch attached to the robot's manipulator. Data collected from the Fronius welding machine is mapped by the local DeviceNet adapter embedded in the ICR5 controller, in this sense, positional data of the robotic arm and positioning table and also material deposition process data collected through the ICR5 controller are passed to a local MQTT adapter via Ethernet cable and TCP/IP sockets. This adapter is embedded on a laptop and is used as the main component of the DCDCE domain of the DT and composes the basis for the dataflows of data storage, visualization and 3D simulation.

Throughout many tests of synchronizing the real robotic cell with its Digital Twin, it was determined that the shortest period of data collection that doesn't compromise the synchronization, data integrity and correct operation of the manufacturing cell was 100 milliseconds. This time interval corresponds to a near real-time setup of data collection, which doesn't invalidate the 3D simulation of the robotic cell and the planned parameters visualization through dashboard.

Using ABB RAPID built-in functions as a basis for the data collection module (TCP/IP client socket in RAPID code), 19 MQTT topics were used for the DT mirroring of the real robotic cell. Each related to a parameter collected through the Ethernet socket communication established with the ABB IRC5 controller. Those parameters relate to the monitored process (GTAW deposition of material) and positional aspects of the robotic cell. The relationship among data collection methods, observed parameters and corresponding MQTT topics are listed and detailed in table I.

With respect to the RAPID programs adapted through the client socket module, those algorithms used for testing were developed in a previous unpublished project developed at the GRACO laboratory using a solution called "KarelToRAPID".

The implemented DT architecture serves 2 dataflows, one related to the Node-RED dashboard and data storage, and the other used as basis for 3D movement simulation. Both of the dataflows makes use of the data published by the MQTT publisher and are connected to the same MQTT broker. The real robotic cell along with the dataflows are evidenced in the diagram shown in figure 2. In the following subsections, each flow will be explained along with its related applications.

A. 1st Flow: Local Node-RED with Cloud Data Storage

The first and main flow of the DT architecture is embedded on a local Node-RED server which makes use of all the 19 topics published by the MQTT broker in DCDCE listed in I. Each topic's message is received by a subscriber in the Node-RED flow, when all messages of the cycle of variables collection are obtained, those messages are then processed structured into a single message to be stored in the cloud Firebase Firestore database. Another application that makes use of the same subscribers is a local dashboard of parameters visualization embedded into the Node-RED flow.

The Node-RED dataflow representational diagram is shown in figure 2. Data collected from the ICR5 controller and stored in cloud may also be used to analyse processes and generate graphs, this feature is evidenced in the graphs generated by data collected from the Firebase Firestore in figure 3.

From the figure, one can infer the layers of deposition in graph 3a, simulated via a virtual controller in RobotStudio, and relate them to the cyclic X and Y movement in graph 3b.

B. 2nd Flow: 3D Simulation in Local RoboDK Server

One of the objectives of the DT architecture 1 was to create an application capable of mirroring the movements of the robotic cell in near real-time. To achieve that, local subscribers connected to the MQTT broker were created in the RoboDK Python environment using the Paho MQTT Python library.

In order to better understand the configuration of the DT robot's kinematics, it's important to explain the data representation of the real robot's kinematics in RAPID code. The common movement data representation in RAPID to be unraveled in messages and topics by the MQTT dataflow is called "robtarget" (robot target). This so called "target" represents the Tool Center Point (TCP) current position, which is related to the tip of the tool attached to the robotic manipulator. The "robtarget" is divided in 4 parts, detailed bellow.

- **trans**: translation, related to the Cartesian position (x, y and z) of the TCP. This position of the target is used to calculate the rotation angle of the robot's joints in the inverse kinematics of the movement;
- **rot**: rotation, describes the current orientation of the tool in the form of a quaternion, [*q1*, *q2*, *q3*, *q4*];
- **robconf**: related to the axes configuration of the robot, joints configuration in zero position;
- **extax**: external axes, describes the movement of external mechanical units in the form [*eax_a*, *eax_b*, *eax_c*, *eax_d*, *eax_e*, *eax_f*]. In the case of the contemplated robotic cell, the external axes are related to the positioning table's movement. The axis *eax_b* describes the rotation of the arm on which the plate is positioned whilst *eax_c* describes the rotation of the plate itself (both in degrees).

Another important aspect of the kinematics' representation is the "wobjdata" or work object data, this variable used in the RAPID code describes the shift from the global zero position to a local workstation zero position, in this case, the center position of the plate on the positioning table. This new referential is used for calculating targets and the current TCP position, so the global position is subtracted by the referential "wobjdata" position. Another referential argument used in the RAPID code is the "tooldata", which which refers to the displacement and reorientation between the tool connection frame and the TCP.

Movement's data representation in RoboDK is also done in targets, but the quaternion notation of ABB's robot is not

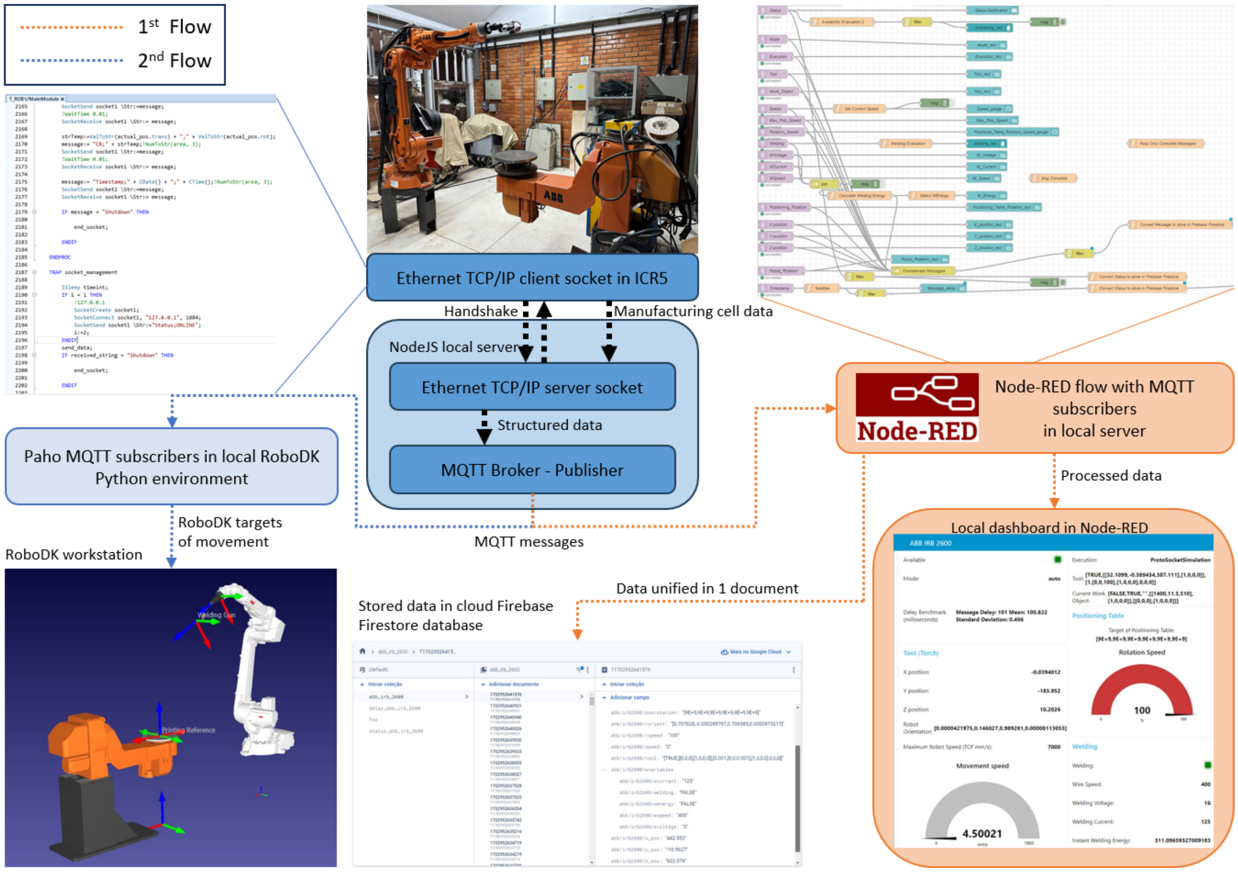


Fig. 2. Developed MQTT flows based on the planned Digital Twin framework architecture.

used by standard in RoboDK. Thus, a built-in RoboDK Python API function is used to convert the robot's orientation from quaternion to a rotational matrix (*quaternion_2_pose()*).

For the 3D assets used in the digital workstation model, standard ABB IRB 2600, ABB IRBP A250 and GTAW torch models present in the RoboDK library were used, highlighted in figure 2. To mirror the robotic cell's movement, MQTT subscribers are used to obtain data published through the broker, the topics related to the 3D simulation are the following:

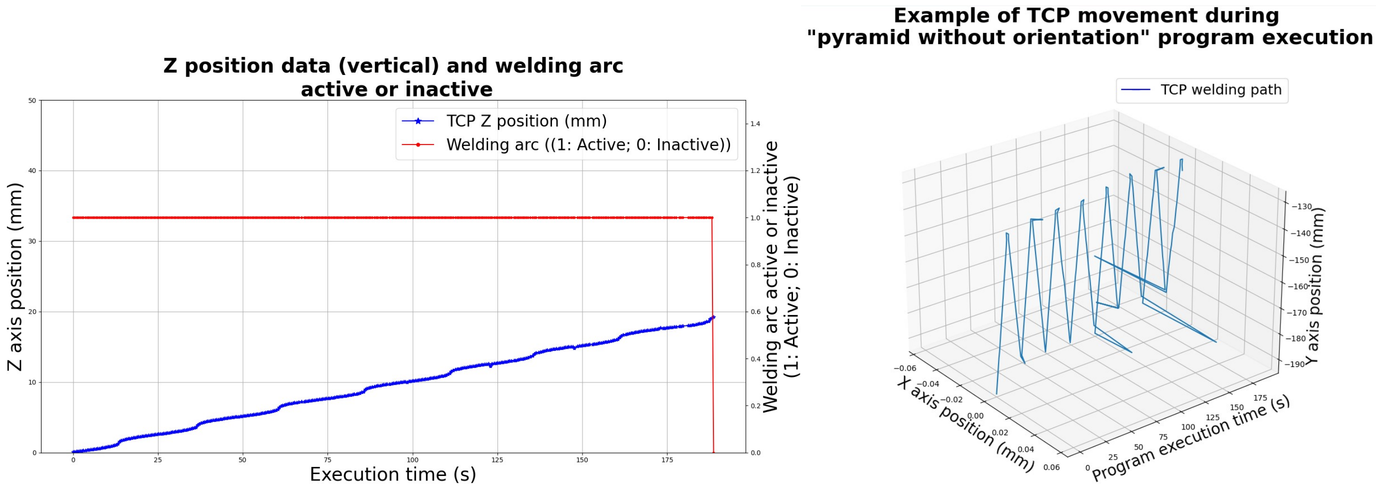
- **"abb/irb2600/cwobj"**: current work object or global referential to the local workstation (middle of the plate on the positioning table) that is used as a offset to the movement simulation;
- **"abb/irb2600/crobt"**: Cartesian parameters (x, y and z) and orientation (quaternion) of the TCP present in the current robot target. The orientation of the TCP is converted into a rotational matrix to be interpreted by the built-in RoboDK movement functions;
- **"abb/irb2600/speed"**: current linear speed of the TCP, used to adjust the speed of the robot's movement in the simulation;
- **"abb/irb2600/welding"**: states whether the welding electric arc is active or not. If it is, Robodk material deposition spray simulation function "SPRAY_ON" is activated, if not, "SPRAY_OFF" is activated;
- **"abb/irb2600/posrotation"**: rotation of the external axes in degrees, the used external axes are *eax_b* and *eax_c*, describing rotations around the x (arm of the positioning table) and z (plate on the arm) axes respectively.

All of those topics are used to simulate the robotic cell's movement through every cycle of parameters published by the MQTT publisher. Tests of mirrored 3D movement were made collecting data from the robotic cell via Ethernet TCP/IP client socket during the execution of RAPID programs. A short demonstration video of the DT can be seen in [22].

Comprising not only the monitoring of the metal AM robotic cell, the project developed in this work represents a validation of the new ISO 23247 norm for Digital Twins in manufacturing, not commonly referenced in state-of-the-art projects. Also, the implementation presented can be used to enable many applications such as fault detection, predictive maintenance and efficiency gain based on AI-enabled analysis.

V. Conclusions

As a result of this work's implementation, a Digital Twin architecture for a metal AM robotic cell was presented. The Digital Twin was composed of an MQTT dataflow through the DT domains established by the ISO 23247 standard. The MQTT protocol was selected by its reliability and fast communication, and also because it is a protocol predicted by the ISO 23247 standard. About the norm that was used as a basis for the DT architecture, each domain predicted by the standard was implemented following its principles and rules. This is evidenced by the implementation of services and



(a) (b)

Fig. 3. Graphs of X, Y and Z position of TCP during deposition simulation in RAPID program "pyramid without orientation" [21] ran in RobotStudio. Data was retrieved from the cloud Firestore database.

applications in each domain, interconnected by communication protocols evidenced in figure 1.

One of the challenges faced during the development of this project was the adaptation of a manufacturing cell that was originally conceived based on protocols and concepts originated from Industry 3.0. In this sense, the main contribution of this work is enabling applications common to Industry 4.0 using low-budget hardware/software to adapt old assets. In this way, implementations like the one presented in this article may contribute to the dissemination of DT applications by using open-source software to enable them.

In future works, it is intended to implement more advanced 3D visualizations based on VR/AR, which are often proposed as part of digital twin implementations [23]. Another update to the DT would be making the MQTT Node-RED dashboard available in cloud, enabling data analysis and surveillance at a distance. Furthermore, tests manufacturing real metallic parts are planned, thus making it possible to analyze the energetic performance of the metal AM robotic cell and quality measurement of the parts manufactured.

References

[1] A. Haleem and M. Javaid, “Additive manufacturing applications in industry 4.0: A review,” *Journal of Industrial Integration and Management*, vol. 04, 08 2019. doi: 10.1142/S2424862219300011

[2] S. Chandrasekaran, “Chapter 2 - functionally graded materials for marine risers**this chapter is co-authored by dr. hari sreenivasan, assistant professor, school of petroleum technology, pandit deendayal petroleum university, gandhinagar. india.” in *Design of Marine Risers with Functionally Graded Materials*, ser. Woodhead Publishing Series in Civil and Structural Engineering, S. Chandrasekaran, Ed. Woodhead Publishing, 2021, pp. 25–58. [Online]. Available: https://doi.org/10.1016/B978-0-12-823537-9.00004-9

[3] P. Ghosal, M. C. Majumder, and A. Chattopadhyay, “Study on direct laser metal deposition,” *Materials Today: Proceedings*, vol. 5, pp. 12 509–12 518, 2018. doi: https://doi.org/10.1016/j.matpr.2018.02.232

[4] P. Dutta Pramanik, B. Mukherjee, S. Pal, B. Upadhyaya, and S. Dutta, *Ubiquitous Manufacturing in the Age of Industry 4.0: A State-of-the-Art Primer*, 11 2019, pp. 73–112. [Online]. Available: https://doi.org/10.1007/978-3-030-14544-6_5

[5] B. P. Santos, F. C. Santos, and T. M. Lima, “Industry 4.0: an overview,” 07 2018.

[6] R. S. Peres, X. Jia, J. Lee, K. Sun, A. W. Colombo, and J. Barata, “Industrial artificial intelligence in industry 4.0 - systematic review, challenges and outlook,” *IEEE Access*, vol. 8, pp. 220 121–220 139, 2020. doi: 10.1109/ACCESS.2020.3042874

[7] M. Singh, R. Srivastava, E. Fuenmayor, V. Kuts, Y. Qiao, N. Murray, and D. Devine, “Applications of digital twin across industries: A review,” *Applied Sciences*, vol. 12, no. 11, 2022. doi: 10.3390/app12115727

[8] ISO 23247-1, “Automation systems and integration — digital twin framework for manufacturing — part 1: Overview and general principles,” 2021.

[9] T. Shen and B. Li, “Digital twins in additive manufacturing: a state-of-the-art review,” *The International Journal of Advanced Manufacturing Technology*, Feb 2024. doi: 10.1007/s00170-024-13092-y

[10] J. V. A. Cabral, E. A. R. Gasca, and A. J. Alvares, “Digital twin implementation for machining center based on iso 23247 standard,” *IEEE Latin America Transactions*, vol. 21, no. 5, p. 628–635, Apr. 2023. doi: 10.1109/TLA.2023.10130834

[11] R. Singh, “3 - welding and joining processes,” in *Applied Welding Engineering (Third Edition)*, third edition ed., R. Singh, Ed. Butterworth-Heinemann, 2020, pp. 157–186.

[12] P. Stavropoulos, “Digitization of manufacturing processes: From sensing to twining,” *Technologies*, vol. 10, no. 5, 2022. doi: 10.3390/technologies10050098. [Online]. Available: https://www.mdpi.com/2227-7080/10/5/98

[13] E. Monroy Cruz, L. R. García Carrillo, and L. A. Cruz Salazar, “Structuring cyber-physical systems for distributed control with iec 61499 standard,” *IEEE Latin America Transactions*, vol. 21, no. 2, p. 251–259, Sep. 2022. doi: 10.1109/TLA.2023.10015217

[14] D. Kim, G. Shao, and G. Jo, “A digital twin implementation architecture for wire + arc additive manufacturing based on iso 23247,” 2022-08-20 04:08:00 2022. doi: https://doi.org/10.1016/j.mfglet.2022.08.008

[15] C. Liu, L. Le Roux, C. Körner, O. Tabaste, F. Lacan, and S. Bigot, “Digital twin-enabled collaborative data management for metal additive manufacturing systems,” *Journal of Manufacturing Systems*, vol. 62, pp. 857–874, 2022. doi: https://doi.org/10.1016/j.jmsy.2020.05.010

[16] A. Alvares, I. Lacroix, M. Maron, and B. Figueroa, “Robotic additive manufacturing by laser metal deposition in the context of industry 4. 0: Manufatura aditiva robotizada por deposição de metal a laser no contexto da indústria 4. 0,” *Concilium*, vol. 23, no. 23, p. 79–103, dez. 2023. doi: 10.53660/CLM-2571-23U13

[17] A. J. Alvares, I. Lacroix, M. A. d. L. Maron, and B. S. Figueroa, “Desenvolvimento de uma célula de manufatura aditiva robotizada baseada no processo deposição de metal à laser usando arame de soldagem,” *Peer Review*, vol. 5, no. 21, p. 17–39, out. 2023. doi: 10.53660/1102.prw2664

[18] Z. Huang, Y. Shen, J. Li, M. Fey, and C. Brecher, "A survey on ai-driven digital twins in industry 4.0: Smart manufacturing and advanced robotics," *Sensors*, vol. 21, no. 19, 2021. doi: 10.3390/s21196340

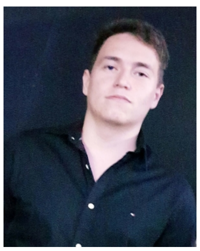
[19] H. Mu, F. He, L. Yuan, P. Commins, H. Wang, and Z. Pan, "Toward a smart wire arc additive manufacturing system: A review on current developments and a framework of digital twin," *Journal of Manufacturing Systems*, vol. 67, pp. 174–189, 2023. doi: https://doi.org/10.1016/j.jmsy.2023.01.012. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0278612523000237

[20] O. MQTT, "Mqtt: The standard for iot messaging," https://mqtt.org/.

[21] A. J. Alvares, G. C. De Carvalho, and J. V. A. Cabral, "Mqtt adapter for abb icr5 controller," https://tinyurl.com/5n8hfamy, 2024.

[22] J. V. A. Cabral, "Abb irb 2600 digital twin demonstration," https://youtu.be/UGdrRNocmDg, 2024.

[23] H. Li, X. Shi, B. Wu, D. R. Corradi, Z. Pan, and H. Li, "Wire arc additive manufacturing: A review on digital twinning and visualization process," *Journal of Manufacturing Processes*, vol. 116, pp. 293–305, 2024. doi: https://doi.org/10.1016/j.jmapro.2024.03.001. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S152661252400210X



João Vítor A. Cabral received the B.S. degree in Mechatronics Engineering from the University of Brasilia. He is currently pursuing his MSc. degree in Mechatronic Systems. Member of the Innovation Group on Industrial Automation (GIAI in Portuguese), his research interests concern industrial automation, Internet of Things, communication protocols, Industry 4.0 and Digital Twins.

Alberto J. Alvares is a Titular Professor at the University of Brasilia, working in Mechatronics Engineering. He graduated in Mechanical Engineering from the University of Brasília (1986), a Master in Mechanical Engineering from the Federal University of Santa Catarina (1990), Doctorate in Mechanical Engineering from the Federal University of Santa Catarina (2005) and Post-doctorate from the University of Jaume in Spain (2017).

Guilherme C. de Carvalho received the Aeronautical Mechanical Engineering from the Instituto Tecnológico de Aeronáutica (1989), a master's degree in Mechanical Engineering from the University of Brasília (1993) and a doctorate (Ph.D.) in Welding Technology - University of Cranfield (1997). He is currently Associate Professor at the University of Brasília.