Proceedings of the 2023 Winter Simulation Conference
C. G. Corlu, S. R. Hunter, H. Lam, B. S. Onggo, J. Shortle, and B. Biller, eds.

DIGITAL TWINS: FEATURES, MODELS, AND SERVICES

Andrea Matta

Department of Mechanical Engineering
Politecnico di Milano
Via La Masa 1
20156, Milano, ITALY

Giovanni Lugaresi

Department of Mechanical Engineering
KU Leuven
Celestijnenlaan 300
3001 Leuven, BELGIUM

ABSTRACT

This work provides an overview of digital twins, digital replicas of real entities conceived to support analysis, improvements, and optimal decisions. Specifically, it aims to better clarify what digital twins are by pointing out their main features, what they can do to support their related physical twins, and which models they use. An illustrative case together with a few selected application examples is used to better describe digital twins. A discussion on the actual challenges and research opportunities is also reported.

1 INTRODUCTION

With the coming of the Industry 4.0 wave, digital representations of products and manufacturing systems have been considered central for optimizing their development, production, and delivery phases. Digital twins (DTs) are not simply conceived as simulation models of their physical counterparts for offline what-if analysis. They are developed as self-adaptable and empowered decision-makers timely aligned with the dynamics of the real entity. The global DT market size was valued at 8.6 billion USD in 2022 and it is expected to reach 137.7 billion USD by 2030 with a Compounded Average Growth Rate (CAGR) of 42.6% (Fortune Business Insights 2023). According to a recent survey, only around 5% of companies affirm DTs are not part of their digital transformation strategy (Dertien and Macmahon 2022) whereas another 86% consider DTs a crucial solution in their strategy. Also, DTs are subject to international standardization efforts (ISO 23247:2021 2021).

DTs are conceived to mirror physical entities, independently from the domain. As a consequence, the variety of applications surveyed by recent literature is vast (Attaran and Celik 2023). Among these, manufacturing, transportation, agriculture, construction, and healthcare are the major domains of DT applications (Liu et al. 2023). Depending on the application, different types of DTs can be distinguished:

- *Product Digital Twin*. The digital replica mirrors a physical object from its manufacturing phase to its disposal along its whole life cycle. The DT collects and analyzes data collected from manufacturing processes as well as from customers' use to provide valuable feedback to improve the product design phase. Three-dimensional representations of products are relevant to simulate the physical behavior of products in specific situations such as machining processes and disassembly operations.
- *System Digital Twin*. The digital replica mirrors a complex system, i.e. a collection of parts organized for some purpose (Coyle 1997). Examples are production lines, automated warehouses, traffic systems, etc. The main purpose of system DTs is to support decision-makers in improving operational efficiency, effectiveness, and costs. Since the time synchronization of activities and resource availability are the core elements, in general, these DTs do not make larger use of geometrical or physical models.
- *Environment Digital Twin*. The digital replica mirrors an environment or a place. Examples of applications are working environments, entertainment places, etc. The main purpose of place DTs is to provide an immersive environment in which the analyst can better evaluate the physical

979-8-3503-6966-3/23/$31.00 ©2023 IEEE

counterpart. In this case, a link with the *metaverse* exists and would deserve further clarification. Virtual Reality and Augmented Reality are key technologies for developing the models used by this type of DTs (Attaran and Celik 2023).
- *Biological Digital Twin*. The digital replica mirrors a human being, or part of him(her), or any other biological system such as a plant or a fish farm. The main purpose is to provide support in medical and life science domains for alert predictions, surgical operations, environment control, etc. Yet, the development of human DTs is still in the early stages (Ahmed and Devoto 2021; Jimenez et al. 2020).

The variety of functionalities that DTs can potentially offer is very large encompassing different application fields from aerospace to urban traffic. This extremely large spectrum of DT use cases has caused a multitude of scientific contributions and market studies from several disciplines, each one with its own terminology, models, and approaches. The scope of this work is to help readers clarify the basic concepts of DTs as well as their underlying models and possible applications. Particular attention is dedicated to simulation in relation to DTs by discussing its key modeling role and related research challenges.

2 DIGITAL TWIN FEATURES

The DT concept was first introduced by Grieves in the early 2000s. The DT "*is a set of virtual information constructs that fully describes a potential or actual physical manufactured product from the micro atomic level to the macro geometrical level. At its optimum, any information that could be obtained from inspecting a physically manufactured product can be obtained from its digital twin*" (Grieves and Vickers 2017). From this definition, it appears the DT was conceived to support product-related decisions from the beginning-of-life phase to the end-of-life phase in a product life-cycle management approach to close the loop from production, use, and disposal to design phases. Many other definitions have been proposed in recent literature to emphasize specific DT features such as the integration and interconnections of DT's elements, the digital counterpart, the predictive capabilities, and the descriptive power. Barricelli et al. (2019) classified 75 papers according to their DT definitions from manufacturing, aviation, and healthcare domains. Other definitions of DTs that are more customized for manufacturing, aviation, and healthcare domains can be found in Negri et al. (2017), Xiong and Wang (2022), and Croatti et al. (2020), respectively.

2.1 Basic Features

Considering the multi-domain applications of the DT technology and its multiple uses, a definition of DT would hardly satisfy all the requirements. Therefore, it seems more practical to list the fundamental attributes of DTs and discuss their relevance. Figure 1 summarizes the DT features.

Digital. Mirroring physical objects within digital entities to train staff and explore new solutions is not a new concept. Perhaps, the most famous example is the digital environment for simulating the Apollo 13 mission (Barricelli et al. 2019). The increase in computational power, data accessibility, and availability of modeling software environments, have created the ideal conditions to develop digital representations of physical entities affordable with cost and time. This feature is universally recognized by literature, i.e., the DT is a set of instructions coded in a computer program to describe the physical entity's behavior. Digitization allows for exploiting the high calculation speed of computers and for obtaining accurate answers in a short time.

Descriptive. The main purpose of the DT is to provide information and knowledge about the physical entity, e.g., its state, its emerging behavior, its trends, or anything that may help the manager of the physical entity to improve its performance. This feature is particularly relevant when the physical entity cannot be reached (e.g., a space shuttle, or a ship) or it is not easy to extrapolate the information (e.g., a human body or a bulk deformation process at high strain rates), or it necessitates sophisticated models and algorithms for knowledge extraction. Coupling the human ability to conceptualize the problem and its context from visualization of tables of numbers, graphs, and other symbolic information, with the computer speed of

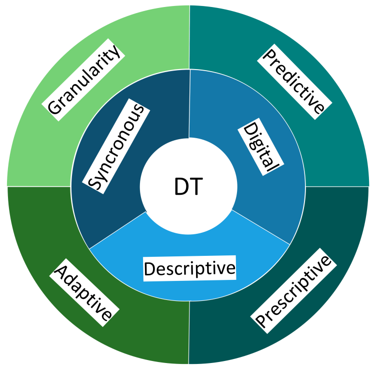

Figure 1: Basic features (inner circle) and advanced features (outer circle) of Digital Twins.

processing information when executing step-by-step operations offers unprecedented opportunities (Grieves 2014). This feature allows DTs to provide functionalities such as status visualization, monitoring, analysis of observed behaviors, diagnosis of malfunctioning, prediction of failures, etc.

Synchronous to physical. The DT should be able to describe the physical entity at any moment. This feature implies the state changes of the physical entity being transferred to the DT. In large and geographically distributed systems, this communication might not be trivial. As far as physical-to-digital communication, significant examples are from heavy industries (e.g., cement, steel, chemical, etc) or power plants, which require continuously updated status of their high-investment equipment. Physical-to-digital communication must be automated as widely recognized by literature, but vice versa there is no unanimous consensus. Indeed, the descriptive information provided by the DT can be used to make a corrective action on the physical entity; this digital-to-physical communication can be asynchronous (i.e., the implementation of the action has a time delay with respect to the decision time), automated (e.g., the cutting parameters of a machine tool are changed after quality inspections) or manual (e.g., the operator starts product changeover in a manufacturing line).

2.2 Advanced Features

In addition to the fundamental features described in the previous section, DTs can be designed also considering advanced attributes as described in the following (see also Figure 1).

Predictive. The alignment with the physical entity state together with the availability of high-speed computational power allows DTs to numerically simulate the future periods under some well-defined scenarios. Simulation results can be used for several purposes such as estimating system performance, checking deliveries, supporting resource allocation, etc. This feature can be very relevant in several contexts, but it is not a strict requirement for a DT. Indeed, for DTs representing single products such as engines, compressors, etc, this feature might not be necessary. On the other hand, it can be relevant in all those situations affected by large complexity and high uncertainty that necessitate experiments with stochastic simulation models, e.g. semiconductor manufacturing systems. One of the first examples of this predictive functionality is the Computer Numerically Control (CNC) of machine tools, which simulates online the tool trajectory to control the velocity and acceleration of machine drives in the scenario defined by the part program. A significant trend is also to use the simulation capabilities of DTs for virtual commissioning of complex systems such as machine tools (Wang et al. 2023) and automotive factories (Mykoniatis and Harris 2021).

Prescriptive. Providing automated feedback to the physical entity in a closed-loop control approach is another advanced feature of DTs. This feature fits especially when the system complexity is high and unmanageable by humans, in these cases, the DT can explore a large number of alternatives and select the best one. Another situation appears when the required feedback time is short. Automation particularly helps when the physical product is difficult to reach or when the decision is repetitive. From this point of view, a programmable logic controller can be considered a primitive DT with simple prescriptive abilities. Lastly, high costs or risks of performance losses may require automated feedback from digital to physical (e.g., in power plants). When a DT does not have a prescriptive feature, it is reasonable to assume that humans will take prescriptive actions after having acquired enough information from the DT.

Adaptive. DTs are normally coupled with physical systems of a certain relevance and costs, which tend to have a long life-cycle during which a lot of changes happen either in the system or in the environment and the context where they operate. DTs should keep their descriptive ability along the whole life-cycle of the physical entity despite changes. Examples are the downgrading of equipment, the increased car traffic in a city, and the opening of a new airport runway. As a consequence, models used to provide DT services might also need updates and dramatic changes to be consistent with the new situation. Automated adaptation of DTs to new system changes and new situations is considered an advanced feature of DTs. Currently, most implementations require manual intervention to the software code, limiting the adaptability of DTs (Tavakoli et al. 2008).

Granularity. Grieves defines DTs able to represent a product from the micro-atomic level to the macro-geometrical level. (Grieves and Vickers 2017). The same consideration can be done for a DT of a factory from the equipment level to the plant level. The fundamental motivation is that DTs are used to support humans for different purposes. Therefore the descriptive ability of DTs must be cross-level encompassing the different fidelities available. Further, DTs must also be consistent with their uses. For instance, if the DT is used for production scheduling on a shop floor, the simulation of tool breakages and machine tool structure deformation does not add much value. Hence, the granularity aspect should be considered an advanced feature, not necessarily available in a DT.

2.3 Digital Models, Digital Shadows, and Digital Twins

A recent trend is to call DT any digital representation of a physical object. This large adoption of the DT term may be a source of confusion and misleading because the functionalities provided by digital models can vary from case to case. This section aims at clarifying the usage of different digital representations.

Since the origins of computer simulation in the ’60s (Tocher 1967), *digital models* are widely used to numerically simulate complex products and systems for the estimation of their key performance measures. Digital models are typically used offline and do not necessarily have to represent existing objects. Indeed, digital models are widely used to evaluate detailed design alternatives during the engineering phases. Further, there is often no automated data flow from the physical entity to the digital model. Vice versa, the results from simulation experiments should help to make and implement decisions on the physical entity. However, this would generally be done manually. In relation to the DT main features described in Figure 1, digital models are digital by definition as well as descriptive. Classical examples of digital model uses are assembly line balancing, layout planning, resource allocation, etc. Another term used in recent literature is *digital factory*. This is related to digital models with extensive use of 3D representations of the whole factory often used for design purposes.

When data management is affordable, the feeding of digital models with data coming from the physical entity becomes natural. This automated physical-to-digital data flow started to be evident in the 80s, with the advent of the Computer Integrated Manufacturing framework (Williams 1989). The term *digital shadow* is widely used in recent literature to emphasize that the digital model is aligned with the physical entity (Kritzinger et al. 2018). This implies that the digital model represents the actual state of the physical entity, so querying the digital shadow or the physical entity is indistinguishable. Many important functionalities can be provided by digital shadows such as state visualization, monitoring, alert prediction, etc. Since the digital shadow follows the physical entity, this last must exist. In relation to the DT main features described in Figure 1, digital shadows are digital, descriptive, and synchronous to physical. Recognizing the status of DTs to digital shadows is an open debate (Bergs et al. 2021) yet and out of the scope of this tutorial. We simply point out that digital shadows own the fundamental features of DTs.

3 DIGITAL TWIN ELEMENTS

According to the conceptual five-dimension model proposed by Tao and Zhang (2017), a DT can be developed on top of different types of models that encompass five elements, namely physical entity, virtual or digital representations of the system, data, connections, and services as represented in Figure 2. This model can effectively convey the main topics that regard DTs, outlining the role of each dimension.

3.1 Physical Dimension

The physical entity represents the real-world physical or perceived system, for instance, a manufacturing system or a machining process that is dynamically connected via a communication or integration medium. In (Tao and Zhang 2017), the physical entity is summarily defined as a set of objects composing the physical entity. However, the knowledge about the physical entity might be far from perfect, and, for this reason, the DT with the acquired sensor data may be used to improve such knowledge. The physical entity model is, therefore, the model of the physical entity with the current knowledge. The model can be any representation of its components and relationships. A variety of models can be used to describe the physical entity, from the least informative ones such as a simple list of components and sub-components to a more formal one such as a class diagram, an Entity Relationship Graph, or the percentages of the material chemical composition. This model can be improved when the knowledge and understanding of the physical entity increase.

3.2 Virtual or Digital Dimension

Depending on the requirements, the virtual entity can be composed by using either discrete events simulation, continuous simulation, or hybrid simulation (Robinson 2014). As already mentioned in section 2, the prediction capability can also be provided by analytical models or simple formulas. However, in this case, it would not be worth considering it as a DT.

In general, multiple digital models can be used to describe the physical entity, each dedicated to a specific physical entity's behavior. The choice of the model should depend on the purpose of use and, therefore, to which service the model will be dedicated. For example, a machine reliability model can be used to schedule maintenance operations, whereas a material flow model will be used to predict the system service level. Another example is a simple model providing the maximum load of a product based on the minimum section and the nominal ultimate tensile strength of the material. In contrast, more sophisticated models would represent the product with FEM equations, material properties, and a solid model. Simulation models offer deep descriptions of the physical counterpart with insights to support decision-making, and predicting anomalies or future failures (VanDerHorn and Mahadevan 2021).

The virtual representations in DTs are mainly of two types: *model-based* providing structural information about the physical entity, and *model-free* representing what was observed in the past. DTs can use both types depending on the service requirements. The following subsections explain the main differences.

3.2.1 Model–based

Model-based digital twins rely on models to describe the behavior of the physical entity (e.g., physics-based models, Discrete Event Simulation (DES) models, analytical models). For example, physics models are typically expressed by mathematical equations that describe the physical laws governing the system's behavior. The related variables can, therefore, incorporate data from sensors to calibrate the model and make accurate predictions. An example of a simple model is Taylor's formula which provides the increased age of a cutting tool based on the cutting speed used by the machine spindle (Mills and Redford 1983). A more advanced model would be a FEM model to simulate the tool wear. Taking a production line as an example of a physical entity, the production rate of the system in the shift can be given by a simplified model that takes into account the status of the bottleneck machine. A more detailed model would simulate the operations until the end of the shift to estimate the number of parts produced in the remaining period.

Model-based digital twins are particularly useful for systems with well-defined physical laws and behavior, such as mechanical or electrical systems, as well as systems with a systematic structure, such as discrete-part manufacturing systems. Of course, model-based digital twins are computationally demanding and generally require much more input information and longer response times. To lighten these virtual models, meta-modeling approaches can be used to simplify the model into a simpler and lighter model by fitting equations from simulation output data. Examples of meta-models, also known as surrogate models, are linear regression, or other types of non-linear regression such as Kriging, Kernel, neural networks, etc. Such lighting techniques can be used either to simplify deterministic models or stochastic models.

3.2.2 Model–free

Model-free digital twins can rely on data-driven algorithms to detect patterns, anomalies, and correlations in data collected from the physical entity. These DTs do not require an explicit model of the system and rely on the data generated by it to provide valuable insights. For instance, a model-free DT can be equipped with machine learning algorithms to identify process parameters that affect its throughput or yield and construct recommendations to improve its efficiency. A granularity feature allows the use of different models with different detail levels, accuracy, costs, and times.

3.3 Service Dimension

The functionalities provided by DTs are embedded in services accessible by humans or other physical and digital objects. Service types can be different depending on the specific application. For example, optimization methods can be combined with simulation methods in simulation-optimization models. In this approach, the service module would utilize techniques such as gradient-based or greedy algorithms to search for the best possible solution. For each type of service, a proper model is needed to be defined. Section 5 presents a selection of relevant services of DTs.

3.4 Data Dimension

In order to accurately reflect the behavior of the real-world entity or to enhance its performance, a DT must be equipped with the ability to automate data acquisition (Dittmann et al. 2021). There are three possible data sources in a DT: the real-time (dynamic) data stream input from the physical entity, the baseline (static) data that was used in building the model for the first time, and the human expert knowledge (optional). DTs use various models and techniques to manage, fuse, and process the extensive and diverse data gathered from the physical entity. The most commonly used data models are relational data models, object-oriented data models, hierarchical data models, etc. More advanced data models such as knowledge graph models can allow for rich and flexible data representation to connect data and infer reasoning among diverse data (Esser and Fahland 2021). Techniques are used to provide feedback mechanisms for decision-making and control of the physical entity. Common examples are data visualization, streaming processing, time-series analysis, batch-oriented processing, and security analytics (Damjanovic-Behrendt and Behrendt 2019).

3.5 Connection Dimension

The bi-directional communication between the physical entity and the DT uses communication protocols. These protocols must reflect the synchronization models adopted for communication. Simple synchronization models are time-based protocols reading/sending a stream of data every constant time, or event-based communicating any state change. What information and data to read/send is also part of the connection models. Sending large amounts of data can create high latency. On the contrary, limiting communication can cause misalignment between DT and the physical entity (Tan and Matta 2022). Depending on the complexity of the physical entity and the geographical distribution of its components and devices, the connection dimension can change significantly. For instance, a product DT such as an engine has connection protocols very different from an automated warehouse DT.

4 DIGITAL TWIN ARCHITECTURE

Once the main features and models of a DT have been defined for its scope, they must be reproduced in software components that actuate its behavior. To do so, a proper architecture is needed. Despite the architecture can be built in accordance with the 5-dimension model (section 3), it will reasonably be adapted to the specific use cases. We can identify some common principles for its design. Additional notes are added to guide researchers in choosing architectural solutions for the development of DTs in research laboratories.

4.1 Communication Technologies

Communication technologies can be categorized into low-power wide-area networks (LPWAN) and short-range networks. LPWANs include, among others, cellular networks (e.g., 4G and 5G), and long-range wide-area networks (Karagiannis et al. 2015). Short-range networks are used where the devices have low power need to be connected locally, and require a gateway for wide-area connectivity. They include, among others, RFID, Bluetooth, and Wireless Local Area Networks (Kurose and Ross 2013). For research laboratories, short-range networks are recommended for ease of deployment and maintenance. Implementation examples in research laboratories for DTs can be found in (Zhou et al. 2021; De Marchi et al. 2022).

Network protocols are used to establish communication between the physical devices and the digital components. For IP-compliant communication, typical choices include Hyper-Text Transfer Protocol (HTTP), which has a relatively higher latency, energy, and memory consumption (Naik 2017), or Data Distribution Service (DDS), usually more suitable for real-time applications given its low latency and higher scalability. Non-IP-compliant devices can be connected using protocols such as the Message Queue Telemetry Transport (MQTT) or OPC-UA, among others. Each of the protocols will manifest its advantages depending on the specific application. For instance, despite it does not guarantee high security, MQTT has a simple and lightweight nature and a wide range of applications for embedded systems (Naik 2017). For research laboratories, MQTT has proven to be a flexible and easy-to-deploy solution, also because of the availability of open-source software libraries (Zhou et al. 2021).

4.2 Communication Protocols

The choice of communication protocol depends on a variety of factors such as the type and number of sensors required, the distance among them, and the control system adopted. Local Area Network (LAN) is a general-purpose networking technology that can be used for a wide range of applications, including industrial control systems. However, its use in industrial environments may require additional measures to ensure reliability and security. This solution is normally adopted in research laboratories for DT technologies (Lugaresi et al. 2021). Profinet is a high-speed, real-time Ethernet-based protocol that is well-suited for use in complex automation systems that require high data transfer rates and precision control. Modbus is a widely used, open-source protocol that is relatively simple and inexpensive, making it a good choice for smaller systems or where cost is a primary consideration.

4.3 Database Components

The application-specific data requirements of the DT drive the implementation of related software components. For instance, the database-related components of a DT encompass the internal repositories or any data storage mechanism that houses data. Cloud-based databases offer benefits such as accessibility, scalability, processing power, and efficiency in data transfer (Alam and El Saddik 2017). Local-based data management may be preferred to ensure data security. Also, hybrid approaches offer compromises (VanDerHorn and Mahadevan 2021).

4.4 Service Components

Other specific components might need to be developed to perform specific functions and implement the services of the DT. The service components can be in the cloud or in the local network for security reasons. They can be grouped into unique components or kept separated, the latter facilitating the use of services developed or provided by different companies. An example of a service component is a scenario manager used for conducting *what-if* analyses based on the outcomes of DT predictive models. Specifically, if a decline in production performance is detected over a certain period, forward-looking simulation models might be initiated to assess the effectiveness of alternative policies under the latest parameter settings. The scenario manager would then gather the experimental results and convert them into a set of instructions.

5 DIGITAL TWIN SERVICES

Within its life cycle, a DT relates to services that can be divided into two main sets: (1) *internal services* guarantee the capability of a DT to provide its functionalities all along its expected lifespan, and (2) *external services* that provide useful benefits for its physical counterpart. All the services make use of the five-dimension model. The next sections further elaborate on the most important services for each set.

5.1 Internal Services

Figure 3 shows a simple representation of a DT life cycle, highlighting the different roles of internal and external services. The following paragraphs further elaborate on relevant internal services.

5.1.1 Model Creation and Update

The DT uses virtual entity models to describe the real entity. These models are generated from the knowledge described in the physical entity models. Model generation must be compliant with their use, i.e., coarse and light models will be generated for rapid responses whereas accurate and heavy models are created for more detailed analysis where the response rapidity is less relevant. Model generation and update can be supported within the same DT environment or it can be done externally. In the first case, the user creates and updates the virtual model using the same constructs present in the DT and offered by the service; in the latter case, the service is just a model upload in the DT architecture. In both cases, the creation and update are executed manually by experts. However, most physical entities are dynamic and may change with a high frequency. Hence, the digital constructs need to be able to adapt to always represent the physical entity. We may identify several types of adaptation: (1) model structure, which refers to the adaptation of structural components, equations, and relationships describing the real entity;(2) model level, which refers to the tuning of the model, i.e. the possibility to exclude, from the digital representation, the components that do not significantly contribute to the system description with respect to a particular goal and (3) model parameters, which refer to the adjustment of the input data model to reflect the current conditions. The model update belongs to the synchronization cycle that routinely aligns the state, validates the model, updates the model and possibly the prediction, and checks if the identified prescription can still be implemented.

5.1.2 State Alignment

The synchronization service has the scope of aligning the states of physical and digital entities. This service is needed even in case the virtual model is a perfect representation of the system because there might be incorrect data and stochasticity. For instance, the number of parts in a queue might vary due to the production rates of downstream and upstream processes. If a short-term performance estimation or prediction via the DT services is of interest, it is essential to start all experiments considering the current buffer level. The synchronization service retrieves the necessary data to correctly represent the system state. An ideal real-time shadowing is practically unachievable for complex systems (Tan and Matta 2022), hence the alignment of the system state would typically be replaced by either a cadenced service (i.e., each fixed time period) or by an event-based trigger as decided with the design of the connection models, this introduces the next internal service, model validation. Synchronization needs a set of traces and a correct system logic to operate (i.e., the real part must be aligned with its virtual counterpart).

5.1.3 Model Validation

Validation refers to checking whether the DT is up to date and aligned with the physical entity, in terms of its capability to correctly represent the system behavior (e.g., output performance metrics, parameter profiles). Online validation guarantees that if an unpredictable change occurs in the physical entity, the DT will be capable of replicating that change as well (Lugaresi et al. 2022; Hua et al. 2022). This service is labeled as a synchronization service because it is necessary for detecting large deviations not imputable to stochastic noise.

5.1.4 Prediction Update

Upon the alignment of DT with the real entity, a new prediction must be elaborated starting from the aligned state. This prediction is generally accomplished using performance evaluation techniques either model-based such as simulation or model-free such as neural networks.

5.1.5 Prescription Check

This service closes the synchronization cycle by checking the validity of the DT prescription (Aydt et al. 2014). For instance, while the DT is elaborating on selecting the control action, the physical entity keeps operating and the identified action could be no more feasible.

5.2 External Services

The services offered externally by DTs are vast. The following part describes the most relevant ones classified according to the enabling features.

5.2.1 Descriptive-based Services

A DT has the capability to monitor the condition of its physical counterpart in real-time because of the continuous flow of information from the physical entity. For example, the use of machine tool vibration data can be used to generate a real-time *health score* of the resource. Monitoring services can be implemented as micro-services or smaller applications that allow the DT to monitor different aspects of the physical entity (Damjanovic-Behrendt and Behrendt 2019). These services include state visualization, tracing, performance metrics, alerting (i.e. detecting and isolating problems), as well as dashboards.

Virtual commissioning involves creating a digital representation of a system in order to test it before construction. DTs offer a promising approach to virtual commissioning since they can provide a realistic simulation environment. These operations reduce the risk of errors and improve the efficiency of the commissioning process. Also, DTs can be used to train operators and maintenance personnel, allowing them to gain familiarity with the system before it is built. In order to exploit DTs for virtual commissioning, the physical entity and its components must be modeled including the functional relationships of sensors and actuators, to be able to simulate their behavior in a virtual environment. A DT must then be constructed and integrated with the control system software, allowing it to interact with the simulated system. Hence, referring to Figure 1, the synchronous feature of a DT for virtual commissioning is referred to as a digital-to-digital scenario.

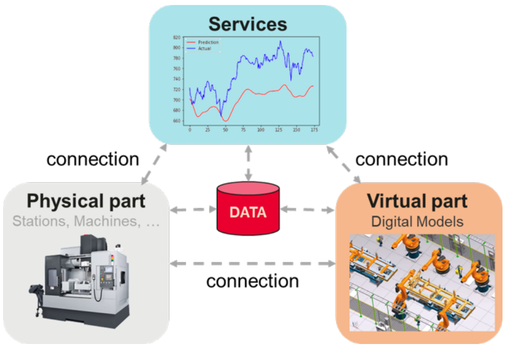

Figure 2: DT concept (adapted from Tao and Zhang, 2017): DTs need models for each dimension.

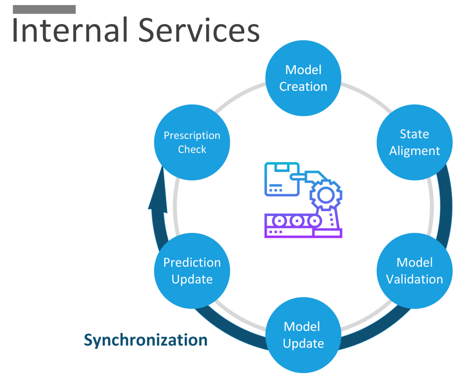

Figure 3: Internal services of DTs.

5.2.2 Prediction-based Services

A DT includes digital models that enable predictions and forward-looking analyses. For instance, the real-time state of a manufacturing system can feed a discrete event simulation model and use it to estimate the future production performance starting from the current conditions (Monostori et al. 2016). It is common to use such services in combination with optimization algorithms. Indeed, simulation-optimization approaches use simulation experiments to construct the search space.

Prognosis services are designed to identify potential issues with the entity before they occur, enabling us to take corrective actions. This is typically done by exploiting advanced analytics and machine learning algorithms, which use data to identify patterns and anomalies in a system's behavior. For example, a DT of a production line might analyze data from sensors installed on each station of the line to detect any deviations from nominal operating conditions (e.g., chip geometry, and temperature profiles). Then, the DT service can use real-time data to feed a simulation model and predict when a particular piece of equipment is likely to fail or require maintenance. Prognosis services can also be used to optimize manufacturing processes by continuously identifying opportunities for process improvement.

5.2.3 Prescription-based Services

A prescription service provides recommendations on how to improve the performance of the real entity, based on the results of experiments performed on the DT. For example, the identification of bottlenecks and inefficiencies in a production process can be done through analyses of digital replicas. A prescription service must then provide counteractions for how to adapt or reconfigure the production line toward improvements. Alternatively, shop-floor data can be used to identify when a station is not working at peak efficiency (prognosis), followed by a prescription for how to adjust the station settings to optimize energy use (e.g., tool changeover, lower spindle speed). Other examples of prescriptive-based services are for planning and control of processes and systems or optimization of some relevant variables.

6 ILLUSTRATIVE EXAMPLE

In this section, a simple production system with its DT architecture is used as an illustrative example. The physical entity is a two-station lab-scale manufacturing system available at Politecnico di Milano, illustrated in Figure 4a. The system is built with LEGO Mindstorm (Lugaresi et al. 2021). The processing times on both stations follow triangular distributions with parameters $(3,8,5)$ and $(2,5,3)\,s$, respectively. Both buffers can hold up to 8 units and 12 pallets are circulating. The blocking after-service (BAS) policy is applied. This setting reproduces manufacturing system dynamics, such as blocking, deadlocks, and stochastic behaviour. Next, each of the five types of DT dimensions is described, with models and enabled services.

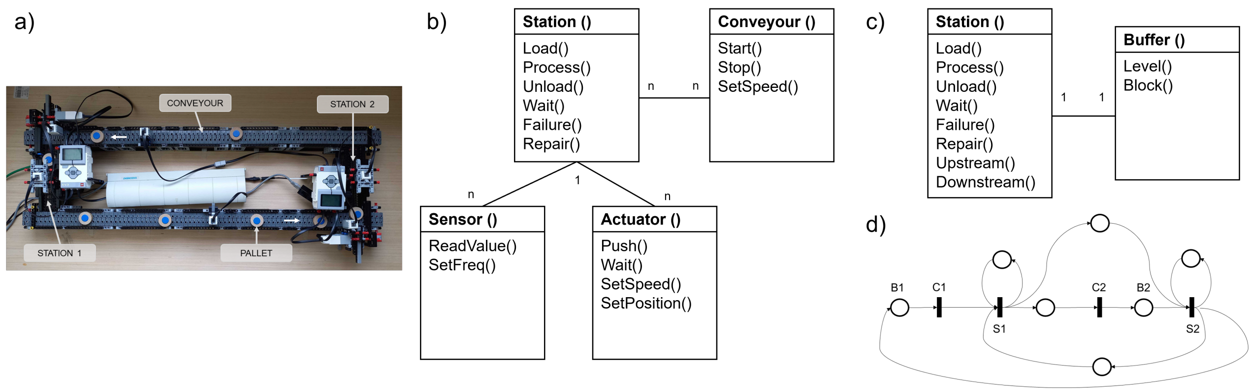

Figure 4: Illustrative example – a) 2-station closed-loop lab-scale system, b) class diagram model of the physical entity, c) class diagram model, and d) Petri Net model of the virtual entity.

6.1 Physical Dimension

The system is a closed-loop production line. It is assumed that unlimited parts to be processed are available and that the pallet loading and unloading phases are very short. Each station processes one pallet at a time. If a failure occurs, the pallet is held for an additional amount of time in the station until it is repaired. The conveyors bring pallets from one station to the other and operate as buffers. A station cannot download parts until the corresponding downstream buffer is full. The class diagram in Figure 4b represents the physical entity model developed from the most recent knowledge about the system.

6.2 Virtual or Digital Dimension

The virtual entity of the illustrative example consists of a DES model of the system. The model is described using the Petri Net formalism, in which the main transitions correspond to the processing of the stations and the places represent the pallets' locations (i.e., in the stations, on the conveyors). See also Figure 4d. Further, the discrete event simulation model is coded in Simpy. A Unified Modeling Language (UML) class diagram model is used to describe the main elements of the model, i.e. stations and the buffers in Figure 4c. The DES model does not need to exactly correspond to the physical model. Indeed, approximations and abstractions are normally introduced in the modeling phase. For instance, transportation time from one station to another is modeled using a dummy station with deterministic processing time, differently from the knowledge described in section 6.1. The simulation model as a virtual entity enables useful services, such as predicting the end-of-the-day performance of the system with current settings, and diagnosis, when simulation experiments are included in a simulation-optimization approach.

6.3 Data Dimension

Data are collected from the physical entity and stored in a real-time InfluxDB database. Data are modeled using UML class diagrams, in which each class represents a table. Such a setting enables real-time monitoring services on raw data. For instance, the last pallet processed by each station can be retrieved via a query to the database component. Also aggregated data resulting from data analyses and predictions are stored in the same database (e.g., throughput, system time).

6.4 Connection Dimension

The connections between all components are achieved by means of the Secure Shell Host (SSH) protocol, and the system is controlled using messages exploiting the MQTT protocol. With this architecture, it is possible to exchange information and implement actions on the system online. Each message is structured

Table 1: Illustrative example: role of digital twin components and enabled services.

Dimension	Components	Models	Enabled Services
Physical	Closed-loop 2-station lab-scale model built with LEGO® components, including sensors and actuators	Class Diagram Model	Demonstrative System Dynamics
Virtual	DES Model in Simpy	Petri Net Model, Class Diagram Model	Prediction, Diagnosis
Data	Real-time Database in InfluxDB	Class Diagram Model	Real-time Monitoring
Connections	Message-based infrastructure using MQTT	Message Model	Prescription (actuators)
Services	Internal services (creation, state alignment, validation, update); External Services (remaining cycle time prediction, throughput estimation, system time estimation)	Synchronization Validation Generation Statistical Model	Prediction, Prognosis (what-if)

following a specific data model. For instance, a message indicating an activity starting in a station is written as the dictionary $\{$*"activity" : s, "id" : id, "ts" : time, "tag" : "s"* $\}$, where *s* is a variable indicating the station number, *id* indicates the identifier of the pallet, *time* the event time-stamp in UNIX format, and *tag* is a string indicating the activity performed in the station. Specific messages can be modeled to indicate the collection of specific data, or controls to the physical actuators, hence enabling the online prescription of corrective actions. Further examples are available in (Lugaresi et al. 2021).

6.5 Service Dimension

For reasons of space, we only consider the case in which a production plan already exists and, at a certain moment, Station 2 undergoes degradation. This station becomes slower with its processing time following a new triangular distribution with parameters $(9,14,11)$. The DT assesses and evaluates counteractions to manage the described situation.

A model generation internal service is deployed implementing the methodology described in (Lugaresi and Matta 2021). The result is a discrete event simulation model in Simpy. The simulation model is validated online using the technique described in (Lugaresi et al. 2022), and synchronized to the last known system state, via the respective internal services. A scenario manager component exploits the virtual entity to perform a what-if analysis. For this simple case, the analysis is conducted on two alternatives: (1) *do nothing*, keep producing at a slower pace and repair the station at the end of the shift; (2) *react*, stop the plant to allow repairing activities and then continue with the production pace before the slow-down. The alternatives are evaluated as two separate simulation experiments are executed in order to determine which scenario maximizes the production output until the rest of the shift. In this case, the prediction results in an average 165 ± 3 parts in the first scenario, and 209 ± 3 parts in the second one. Hence, the second scenario is selected and the prescription is sent for implementation.

7 APPLICATION EXAMPLES

In this section, three noteworthy use cases from the literature are selected and analyzed in light of the identified DT characteristics in this paper. Table 2 collects these use cases and summarizes their main features. Leng et al. (2021) built the DT of a warehouse to support packing and storage assignment optimization. The DT gathers real-time information from a physical warehouse product-service system and applies it to a digital model. The DT incorporates a joint optimization model to efficiently manage both stacking and storage assignments of the warehouse in a timely manner. With real-time data, the joint optimization model is able to make periodic optimal decisions, which are then verified through a simulation engine. The authors developed and tested a prototype DT using a tobacco warehouse product-service system as a case study.

Table 2: Description of selected use cases with a significant DT application.

Use Cases		Leng et al. (2021)	Zhang et al. (2021)	Son et al. (2021)
Features	Basic	*all*	*all*	*all*
	Advanced	Prescriptive	Predictive, prescriptive	Predictive
Dimensions	Physical	Warehouse	Job-shop	Automotive Plant
	Virtual	3D model, simulation	Optimization Model (MILP)	Information models: product, process, plan, plant, and resources
	Service	Packing Optimization, Coordination, Storage Assignment	API interfaces: availability prediction, disturbance detection, performance evaluation	Production Planning, Performance Prediction
	Data	n.a.	ODBC, JDBC interfaces	Service-oriented
	Connection	Event-based	MTConnect, OPC-UA, TCP/IP	Web-based

Zhang et al. (2021) proposed a methodology that involves the integration of a DT to enhance the dynamic scheduling process in a machining job shop that produces hydraulic valves. The DT is exploited within a production planning and control context. The physical entity is given an initial production plan. The synchronization service updates in real-time the resource availability. Then, a scheduling component elaborates the new production plan in a rolling approach.

Son et al. (2021) introduced a DT to predict if an automotive product can be manufactured according to a predefined schedule in the presence of abnormal scenarios. The authors' approach includes an information model purposely designed to represent the main objects involved in automotive body production lines. The DT is developed jointly with other components to create an integrated web-based manufacturing platform. The authors used real production planning data to verify the DT performance through dedicated experiments.

8 FINAL REMARKS AND RESEARCH CHALLENGES

This work has summarized the current knowledge about DTs with the scope of making the readers aware of the recent essence of this paradigm. DTs are not a single technology but a set of technologies integrated to provide specific services in relation to a product, a system, a place, or a human. The consequence is an increase in complexity that represents its main limitation. DTs coupled with simple products or systems would be too expensive to develop and maintain for the expected benefits. This work has two main limitations. The first is the space limit that forced us to summarize concepts that would deserve more discussion. For instance, the cybersecurity of DTs is a relevant problem that is attracting the attention of researchers and practitioners. The second is the bias of the authors, who are more involved in system DTs rather than product, place, and human DTs.

Research is also needed to tackle the challenges that are currently limiting the DTs. Integration challenges arise from the complexity of managing existing DTs once they are operational. Although the interaction between the physical and digital worlds is a crucial aspect of DTs, few studies have explored this topic in depth. Thus, there is a need to develop techniques specifically tailored to address the challenge of physical-digital alignment. Also, the digital-to-physical alignment is not clear, a prescription selected by DT from solving an optimization problem might not be implementable anymore in reality because this last evolved during the optimization task execution. This is particularly important for achieving level 5 of the DT evolution framework, which involves building and managing DTs at the federated level and integration with information systems (e.g., ERP, MES). However, there is currently a lack of clarity on how this will be achieved. Overconfidence in DTs might be the cause of trusting simplistic models that are not adequate for the complexity of the real entity. How to continuously validate the models used by a DT during its operation is another challenge. Model validation techniques are currently used offline and may require limiting assumptions, specific experiments, and long times. Online validation of digital twin models is an open field for novel approaches. When a DT model is not valid, a modification must be introduced to re-align the twins. How to modify the used models is a challenge. Techniques from Artificial Intelligence can be used to fit the behavior of the real entity or a single part of it. More advanced approaches have the potential to automatically discover from data new knowledge about the real entity and use it for improving the DT models. Several studies have proposed DT architectures that are largely domain- and technology-specific. The use cases, applications, and domains vary extensively, together with the proposed application-specific architectures. The development of a DT remains dependent on the requirements of its intended applications, with several related complex choices (e.g., communication standards, data management system).

ACKNOWLEDGMENTS

This work was partially supported by the Horizon Europe Project ”Auto-Twin” under Grant Agreement No. 101092021.

REFERENCES

Ahmed, H., and L. Devoto. 2021. “The Potential of a Digital Twin in Surgery”. *Surgical Innovation* 28(4):509–510.

Alam, K. M., and A. El Saddik. 2017. “C2PS: A Digital Twin Architecture Reference Model for the Cloud-based Cyber-physical Systems”. *IEEE Access* 5:2050–2062.

Attaran, M., and B. G. Celik. 2023. “Digital Twin: Benefits, Use Cases, Challenges, and Opportunities”. *Decision Analytics Journal* 6:100165.

Aydt, H., W. Cai, and S. John Turner. 2014. “Dynamic Specialization for Symbiotic Simulation-based Operational Decision Support Using the Evolutionary Computing Modelling Language (ECML)”. *Journal of Simulation* 8(2):105–114.

Barricelli, B. R., E. Casiraghi, and D. Fogli. 2019. “A Survey on Digital Twin: Definitions, Characteristics, Applications, and Design Implications”. *IEEE Access* 7:167653–167671.

Bergs, T., S. Gierlings, T. Auerbach, A. Klink, D. Schraknepper, and T. Augspurger. 2021. “The Concept of Digital Twin and Digital Shadow in Manufacturing”. *Procedia CIRP* 101:81–84. 9th CIRP Conference on High Performance Cutting.

Coyle, R. G. 1997. “System Dynamics Modelling: a Practical Approach”. *Journal of the Operational Research Society* 48(5):544.

Croatti, A., M. Gabellini, S. Montagna, and A. Ricci. 2020. “On the Integration of Agents and Digital Twins in Healthcare”. *Journal of Medical Systems* 44:1–8.

Damjanovic-Behrendt, V., and W. Behrendt. 2019. “An Open Source Approach to the Design and Implementation of Digital Twins for Smart Manufacturing”. *International Journal of Computer Integrated Manufacturing* 32:366–384.

De Marchi, M., B. G. Mark, N. Angeli, E. Rauch, and D. T. Matt. 2022. “Using RFID and Industrial IoT in Manual Assembly for Data Collection and Processing in Learning Factories”. *Available at SSRN 4074692*.

Dertien, S., and C. Macmahon. 2022. “State of Digital Twin 2022”. *White Paper*:1–12.

Dittmann, S., P. Zhang, A. Glodde, and F. Dietrich. 2021. “Towards a Scalable Implementation of Digital Twins-A Generic Method to Acquire Shopfloor Data”. *Procedia CIRP* 96:157–162.

Esser, S., and D. Fahland. 2021. “Multi-dimensional event data in graph databases”. *Journal on Data Semantics* 10(1-2):109–141.

Fortune Business Insights 2023. “Digital Twin Market Report”. https://www.fortunebusinessinsights.com/digital-twin-market-106246, accessed 12.05.2023.

Grieves, M. 2014. “Digital twin: Manufacturing Excellence Through Virtual Factory Replication”. *White paper*:1–7.

Grieves, M., and J. Vickers. 2017. *Digital Twin: Mitigating Unpredictable, Undesirable Emergent Behavior in Complex Systems*, 85–113. Cham, Switzerland: Springer International Publishing.

Hua, E. Y., S. Lazarova-Molnar, and D. P. Francis. 2022. “Validation of Digital Twins: Challenges and Opportunities”. In *Proceedings of 2022 Winter Simulation Conference*, edited by B. Feng, G. Pedrielli, Y. Peng, S. Shashaani, E. Song, C. G. Corlu, L. H. Lee, E. P. Chew, T. Roeder, and P. Lendermann, 2900–2911. Piscataway, New Jersey: Institute of Electrical and Electronics Engineers, Inc.

ISO 23247:2021 2021. “Automation Systems and Integration — Digital Twin Framework for Manufacturing — Part 1: Overview and General Principles”. Standard, International Organization for Standardization, Geneva, Switzerland.

Jimenez, J. I., H. Jahankhani, and S. Kendzierskyj. 2020. *Health Care in the Cyberspace: Medical Cyber-Physical System and Digital Twin Challenges*, 79–92. Cham, Switzerland: Springer International Publishing.

Karagiannis, V., P. Chatzimisios, F. Vazquez-Gallego, and J. Alonso-Zarate. 2015. “A Survey on Application Layer Protocols for the Internet of Things”. *Transaction on IoT and Cloud computing* 3(1):11–17.

Kritzinger, W., M. Karner, G. Traar, J. Henjes, and W. Sihn. 2018. “Digital Twin in Manufacturing: A Categorical Literature Review and Classification”. *IFAC-PapersOnLine* 51(11):1016–1022.

Kurose, J., and K. Ross. 2013. "Multimedia networking". *Computer networking: a top-down approach, 6th edn. Pearson, New Jersey*:588–592.

Leng, J., D. Yan, Q. Liu, H. Zhang, G. Zhao, L. Wei, D. Zhang, A. Yu, and X. Chen. 2021. "Digital Twin-driven Joint Optimisation of Packing and Storage Assignment in Large-scale Automated High-rise Warehouse Product-Service System". *International Journal of Computer Integrated Manufacturing* 34(7-8):783–800.

Liu, X., D. Jiang, B. Tao, F. Xiang, G. Jiang, Y. Sun, J. Kong, and G. Li. 2023. "A Systematic Review of Digital Twin About Physical Entities, Virtual Models, Twin Data, and Applications". *Advanced Engineering Informatics* 55:101876.

Lugaresi, G., V. V. Alba, and A. Matta. 2021. "Lab-scale Models of Manufacturing Systems for Testing Real-time Simulation and Production Control Technologies". *Journal of Manufacturing Systems* 58:93–108.

Lugaresi, G., S. Gangemi, G. Gazzoni, and A. Matta. 2022. "Online Validation of Simulation-Based Digital Twins Exploiting Time Series Analysis". In *Proceedings of 2022 Winter Simulation Conference*, edited by B. Feng, G. Pedrielli, Y. Peng, S. Shashaani, E. Song, C. G. Corlu, L. H. Lee, E. P. Chew, T. Roeder, and P. Lendermann, 2912–2923. Piscataway, New Jersey: Institute of Electrical and Electronics Engineers, Inc.

Lugaresi, G., and A. Matta. 2021. "Automated Manufacturing System Discovery and Digital Twin Generation". *Journal of Manufacturing Systems* 59:51–66.

Mills, B., and A. Redford. 1983. "Manufacturing of Engineering Materials". *Applied Science*:20.

Monostori, L., B. Kádár, T. Bauernhansl, S. Kondoh, S. Kumara, G. Reinhart, O. Sauer, G. Schuh, W. Sihn, and K. Ueda. 2016. "Cyber-physical Systems in Manufacturing". *Cirp Annals* 65(2):621–641.

Mykoniatis, K., and G. A. Harris. 2021. "A Digital Twin Emulator of a Modular Production System Using a Data-driven Hybrid Modeling and Simulation Approach". *Journal of Intelligent Manufacturing*:1–13.

Naik, N. 2017. "Choice of Effective Messaging Protocols for IoT Systems: MQTT, CoAP, AMQP and HTTP". In *2017 IEEE international systems engineering symposium (ISSE)*, 1–7. IEEE.

Negri, E., L. Fumagalli, and M. Macchi. 2017. "A Review of the Roles of Digital Twin in CPS-based Production Systems". *Procedia Manufacturing* (11):939–948.

Robinson, S. 2014. *Simulation: the Practice of Model Development and Use*. London, United Kingdom: Bloomsbury Publishing.

Son, Y. H., K. T. Park, D. Lee, S. W. Jeon, and S. Do Noh. 2021. "Digital Twin–based Cyber-physical System for Automotive Body Production Lines". *The International Journal of Advanced Manufacturing Technology* 115:291–310.

Tan, B., and A. Matta. 2022. "Optimizing Digital Twin Synchronization in a Finite Horizon". In *Proceedings of 2022 Winter Simulation Conference*, edited by B. Feng, G. Pedrielli, Y. Peng, S. Shashaani, E. Song, C. G. Corlu, L. H. Lee, E. P. Chew, T. Roeder, and P. Lendermann, 2924–2935. Piscataway, New Jersey: Institute of Electrical and Electronics Engineers, Inc.

Tao, F., and M. Zhang. 2017. "Digital Twin Shop-floor: a New Shop-floor Paradigm Towards Smart Manufacturing". *Ieee Access* 5:20418–20427.

Tavakoli, S., A. Mousavi, and A. Komashie. 2008. "A Generic Framework for Real-time Discrete Event Simulation (DES) Modelling". In *Proceedings of the 2008 Winter Simulation Conference*, edited by S. J. Mason, R. R. Hill, L. Mönch, O. Rose, T. Jefferson, and J. W. Fowler, 1931–1938. Piscataway, New Jersey: Institute of Electrical and Electronics Engineers, Inc.

Tocher, K. D. 1967. *The art of simulation*. English Universities Press.

VanDerHorn, E., and S. Mahadevan. 2021. "Digital Twin: Generalization, Characterization and Implementation". *Decision Support Systems*:113524.

Wang, J., X. Niu, R. X. Gao, Z. Huang, and R. Xue. 2023. "Digital Twin-driven Virtual Commissioning of Machine Tool". *Robotics and Computer-Integrated Manufacturing* 81:102499.

Williams, T. J. 1989. "A Reference Model for Computer Integrated Manufacturing (CIM)". *International Purdue Works*.

Xiong, M., and H. Wang. 2022. "Digital Twin Applications in Aviation Industry: a Review". *The International Journal of Advanced Manufacturing Technology* 121(9-10):5677–5692.

Zhang, M., F. Tao, and A. Nee. 2021. "Digital Twin Enhanced Dynamic Job-shop Scheduling". *Journal of Manufacturing Systems* 58:146–156.

Zhou, J., Y. Wang, and Y. Q. Chua. 2021. "Real-Time OEE for Industry 4.0 Learning and Practice Training". In *Proceedings of the 2021 Conference on Learning Factories (CLF)*.

AUTHOR BIOGRAPHIES

ANDREA MATTA is Full Professor at Politecnico di Milano, where he currently teaches integrated manufacturing systems and manufacturing processes. His research area includes analysis, design, and management of manufacturing and healthcare systems. He is Editor in Chief of the Flexible Services and Manufacturing Journal. His email address is andrea.matta@polimi.it.

GIOVANNI LUGARESI is Assistant Professor at the Department of Mechanical Engineering of KU Leuven (Belgium). He works on production planning and control, process mining, and robust optimization. His email is giovanni.lugaresi@kuleuven.be.