Proceedings of the ASME 2021 16th International Manufacturing Science and Engineering Conference
MSEC2021
June 21-25, 2021, Virtual, Online

MSEC2021-64438

EDGE COMPUTING ENHANCED DIGITAL TWINS FOR SMART MANUFACTURING

Huiyue Huang
The University of Auckland
Auckland, New Zealand

Xun Xu
The University of Auckland
Auckland, New Zealand

ABSTRACT

Digital Twin is one of the key enabling technologies for smart manufacturing in the context of Industry 4.0. The combination with advanced data analytics and information and communication technologies allows Digital Twins to perform real-time simulation, optimization and prediction to their physical counterparts. Efficient bi-directional data exchange is the foundation for Digital Twin implementation. However, the widely mentioned cloud-based architecture has disadvantages, such as high pressure on bandwidth and long latency time, which limit Digital Twins to provide real-time operating responses in dynamic manufacturing processes. Edge computing has the characteristics of low connectivity, the capability of immediate analysis and access to temporal data for real-time analytics, which makes it a fit-for-purpose technology for Digital Twin development. In this paper, the benefits of edge computing to Digital Twin are first explained through the reviews of the two technologies. The Digital Twin functions to be performed at the edge are then elaborated. After that, how the data model will be used in the edge for data mapping to realize the Digital Twin is illustrated and the data mapping strategy based on the EXPRESS schemas is discussed. Finally, a case study is carried out to verify the data mapping strategy based on EXPRESS schema. This research work refers to ISO/DIS 23247 Automation systems and integration — Digital Twin framework for manufacturing.

Keywords: Digital Twin; Edge Computing; Data Model; EXPRESS

1. INTRODUCTION

In the era of Industry 4.0, physical equipment in a factory is augmented with smart sensors, big data, and Artificial Intelligence (AI) to monitor the entire production process, make decisions autonomously, and achieve smart manufacturing [1]. This demand is particularly urgent because of the unexpected challenges to the manufacturing industry raised by the COVID-19 pandemic. An essential prerequisite for implementing smart manufacturing is to achieve the integration of the cyber-physical space of manufacturing by using state-of-the-art technologies [2]. Cyber Physical System (CPS) is an automated system that enables connection of the physical reality with computing and communication infrastructures by utilizing advanced Information and Communication Technologies [3]. It facilitates physical-cyber integration.

CPS is characterized by a physical asset, its digital replica, and the connection in between. Digital Twin (DT), acting as the digital replica of the physical asset, providing the ability to simulate, predict and optimize the physical asset, is in the center of CPS [4]. Utilizing advanced technologies like big data analytics and AI, DT helps its physical counterpart to be intelligent.

Edge computing is a critical technology for DT implementation. In the work process of DT, not only the data flow from the physical world to the cyber world is required to drive the digital model, but also the data flow in the opposite direction is necessary for the real-time optimization of the process [5]. Digital Thread, a popular technology at the moment, provides a mature solution for the first requirement, as a record of a product or system lifetime [6]. However, how to transmit the decision result of DT to its physical counterpart in real-time is rarely discussed. Besides, shorten the latency of data mapping and improve computation efficiency is a challenge for DT applications. Edge computing is a distributed computing paradigm that provides computing resources and data storage close to physical devices. The computation and network connection capabilities of edge computing allow it to map the physical entities in the cyber world and help with the real-time tasks of DT.

In this paper, how to utilize edge computing in the DT application is illustrated and how to perform data mapping with EXPRESS schema is explained. The next section presents related works. The architecture of edge-based DT is presented in Section 3. Data process and data mapping of DT at the edge are discussed in Section 4. Section 5 presents the case study for the verification of the data mapping method based on EXPRESS schema proposed in Section 4. Section 6 concludes the research and future work.

2. LITERATURE REVIEW

In order to have a clear understanding of the research background, state-of-art, and research gaps, this section reviews related work, including internet and communication technologies within DT, the DT's requirements for computing resources, and edge computing.

2.1 Digital Twin in smart manufacturing

The concept of DT was first mentioned by Michael Grieves in 2003 in his presentation on product lifecycle management at the University of Michigan. It did not receive much attention until NASA, and the U.S. air force adopted it in 2012 [7]. In 2016, DT was defined as "an integrated multi-physics, multi-scale, probabilistic simulation of an as-built system, enabled by Digital Thread, that uses the best available models, sensor information, and input data to mirror and predict activities/performance over the life of its corresponding physical twin" [8]. The under developing standard considers DT as a digital representation of some physical thing, which enables convergence between the physical entity and digital entity at an appropriate rate of synchronization [9]. These definitions propose the following two characteristics of DT: 1) The two-way real-time information exchange between the physical world and the cyber world; 2) DT provides more functions to the physical entity, such as simulation, optimization, and history tracking.

Since the concept DT came into being, data acquisition for the implementation of DT has been studied a lot. Industrial communication protocols are widely used to collect data from physical devices, including Modbus, PROFINET, EtherNet, etc. Other protocols have been reviewed in our previous work [4]. Internet of Things (IoT) is an emerging technology changing the way digital and physical interact [10]. IoT provides the connection and access to the intelligence of DT, working as the backbone of DT [10]. Saad et.al proposed an IoT-based DT of an interconnected microgrids system for the energy cyber-physical systems [11]. IoT can also be used for DT implementation to enhance the resiliency against cyber-attacks [12]. PTC and IBM also provide their DT solution based on their IoT platform (ThingWorx, Watson IoT Platform) [13]. The IoT discussion focuses on the operational phase of the devices, whilst the devices have other lifecycle phases that need to be tracked [14]. A digital thread refers to the communication framework that presents the view of connected data flow and asset data throughout the lifecycle across traditional isolated functional perspectives [15]. It provides an efficient solution for DT data acquisition. However, the other direction of data flow, from DT to its physical counterpart is hardly discussed, especially for the tasks that require real-time response.

According to different physical objects and different purposes, there are various types of DT. Different DTs have different demands for computing resources. As the connotation of DT become border, and the combination with AI and machine learning to provide more functions, the demand for computing resources increases. Meanwhile, the data generated during all phases of the product lifecycle becomes larger, more diverse and complex. The architecture of cloud-based DT was proposed to extend the capabilities of DT. Hu et.al applied a cloud-based DT to reduce the computing resources in the information processing center while keeping good performance [16]. Cloud-computing was adopted to enable the integration of services that existing cloud architecture for the DT of robotics [17]. However, there are three major disadvantages of cloud-based solutions [18]. Firstly, the explosive growth of data can cause network congestion when they are conveyed over the network to the cloud. High bandwidth is therefore required to minimize the network congestion, which will cost a lot. Secondly, the cloud may become unusable when there is no network available. Thirdly, the delayed response of a packet of data travelled from machines and sensors to the cloud and then back to the desired device in a round trip may lead to latency. This affects the real-time synchronization between cyberspace and physical space, thereby affecting the capability of the DT in real-time prediction, analyses, simulation, and collaboration.

In a word, DT is the digital replica synchronized with its physical counterpart, characterized by the bi-directional information exchange. The combination of DT and cloud computing is a popular architecture that can improve the computing power of DT and add intelligence to DT. However, cloud computing-based DT architecture has the disadvantages of highly dependent on network conditions and difficult to handle real-time tasks.

2.2 Edge computing

Edge computing describes an infrastructure that is physically located close to or is integrated to the data source, as a micro-cloud service that can provide the core capabilities of networking, data analyzing, storing and utilizing close to or inside the edge devices [19]–[21]. Edge computing is a key enabler of Industrial IoT (IIoT), since it allows reacting to process data in real-time by extending communication, storage and processing capabilities from the cloud to the peripherals of production systems, and it has the ability to pre-process data before sending them to the cloud [22]. By doing this, edge computing can greatly relieve the pressure of network bandwidth and the burden of centralized data processing. At the same time, it reduces the transmission delay and privacy leakage occurred in cloud computing, and improves the response speed and reliability of services [22].

In conclusion, edge computing allows DT to meet the requirement of real-time for lightweight, intelligent manufacturing, and the reasonable allocation of computing resources. To address the above issues, an architecture for the implementation of edge computing-based DT is presented in Section 3.

3. EDGE COMPUTING-BASED DIGITAL TWIN

Recently, edge computing has been adopted for DT applications to solve some typical problems. However, there is not much research on systematic analysis of the role edge computing plays. This article focuses on efficient information mapping between DT and its physical entity based on edge computing. In this section, the application architecture of edge computing-based DT is presented, and the DT functions performed at the edge are discussed.

3.1 The architecture of edge computing-based DT

DT is the digital representation located in the cyber world that reflects the physical entity status in real-time. The implementation of a DT is an encapsulation of a software object or model that mirrors the physical object and its behaviors. DT possesses powerful data analysis capabilities supporting functions like real-time simulation, online optimization, and predictive maintenance to improve equipment performance. In addition, it provides an interactive interface for operators to monitor and issue instructions. To be a high-performance DT, integrating cloud and edge computing resources rationally is necessary. Overall, the functions that consume more computing and storage resources and have low real-time requirements are arranged in the cloud, while edge computing is used for data fusion that reduce the pressure for both cloud and data transmission, data transformation to mirror the physical entity in the cyber world, and functions that require high real-time performance. The application architecture of edge computing-based DT is shown in Fig. 1, which is consists of three parts.

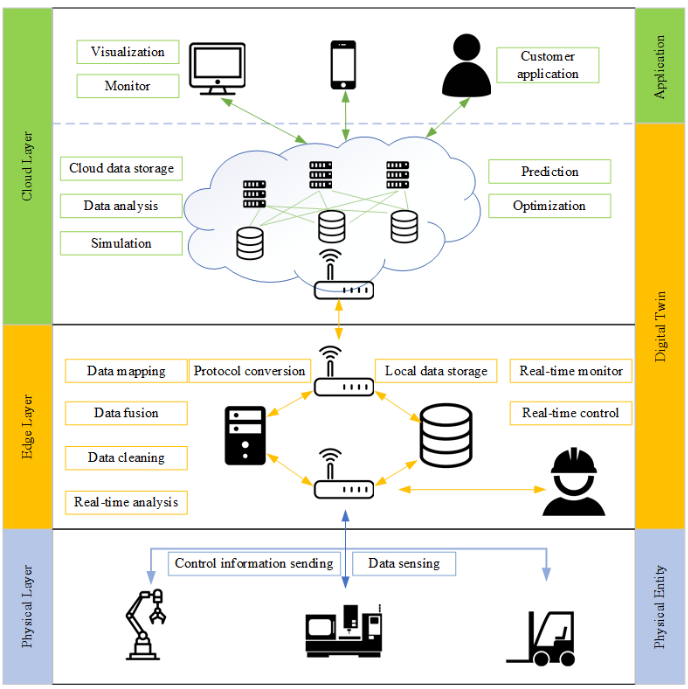

FIGURE 1: THE ARCHITECTURE OF EDGE COMPUTING-BASED DT

1) Physical Entity

Equipment, sensors for data collection, and adapters to provide standard network communication ability are the physical entities located in the physical layer. Because of the diversity of equipment types in the shopfloors, a standard protocol for data transmission is needed. OPC UA and MTConnect are the widely used protocols for field data collection and communication. OPC UA (OPC Unified Architecture) makes data available by providing communication protocol for industrial automation, while MTConnect makes data actionable by offering a vocabulary dictionary that ensures communication among devices [23].

2) Digital Twin

As shown in Fig. 1, DT spans two layers, the edge layer and the cloud layer. Data fusion from various manufacturing resources is an important step for data acquisition and intelligent control in the management of dynamic manufacturing processes. Data cleansing, data mapping and protocol conversion are essential for DT implementation. With multi-source data gathered, data fusion is performed by the edge devices to extract features and monitor the equipment status. Data models, including EXPRESS data model, XML (eXtensible Markup Language) data model, are located in the edge devices for data transformation to mirror the physical entity. Meanwhile, the computing resources provided by edge devices can also support DT to complete some real-time tasks. After processing by the edge devices, the network pressure is reduced, and the data obtained in the cloud is formatted and can be used directly. A mirror of the physical entity is created in the cloud layer. Benefiting from a large number of computing resources, the cloud can perform complex algorithms like AI and machine learning, and support advanced decision-making functions.

3) Application

The consumers and clients can establish a connection with DT through the Internet. The consumers and clients can be any person, applications, and other DTs that need support. Data visualizing can be performed by taking data from the DT. Other further data analysis functions can also be realized through the data provided by DT.

3.2 Edge computing functions analysis

Edge computing is an indispensable technology for DT implementation due to its unique advantages. In this section, the DT data processing performed in the edge device will be discussed in detail.

1) Physical Entity

Data cleansing refers to the process of evaluating, validating, and correcting data to provide efficient, accurate and effective data [24]. In the data collection process, each sensor aggregates the collected data and then transmits the data to the edge device. However, in the actual data collection scenario, data collected from various resources are of poor quality which will affect the following data analysis result. Data cleansing provides a better data quality by error removing, inconsistencies resolving and data transforming [25]. Data anomaly detection is the foundation for data cleansing to identify the errors within the raw data set. It is a challenge to complete anomaly detection in real-time as the amount of raw data increases. Data cleaning at the edge can reduce the pressure on network transmission to the cloud. In addition, the edge device is closer to the underlying network, which makes it easier to collect data.

2) Storage and privacy

Collecting and taking advantage of collected data is the essence of DT, and at the same time brings security risks. Since more business-sensitive data is stored in the cloud, privacy leakage is becoming one of the most serious concerns. Security vulnerabilities in many IoT applications indicate that IoT applications may put physical systems at risk. Edge computing offers a platform for privacy computing and storage service. In the context of edge computing-based DT, sensitive data are stored in the local, edge and cloud respectively according to their characteristics, and raw data is hidden and cannot be obtained from a part of the data blocks [26]. There are various comprehensive security architecture designed based on edge computing architecture, such as distributed firewalls, intrusion detection systems, authentication and authorization algorithms [27].

3) Service and response

As the amount of data generated by the physical system increased, the pressure on network data transmission is increasing. Whilst DT in the cloud is always located far from the physical devices and network bandwidth is limited, which makes it difficult to handle real-time tasks which usually involve monitoring the status of the equipment and reacting to the situation in real-time (e.g., equipment anomaly detection). The key components for real-time task handling include data processing, computing, situation monitoring, and control. Edge computing resources are allocated for real-time tasks and execution to address the problem of processing delays, response delays, and bandwidth shortage.

4) Data mapping

In a data warehouse, data mapping is the process of creating connections between the two data models (source and target) [28]. In the context of DT, data mapping refers to the synchronous mapping of physical data and virtual data model [29]. Data mapping of DT mainly contains two parts, the establishment of data association between two data models and data synchronization. The descriptive model is important for data association, as it provides the digital representation of a physical entity and the basis of DT functions. An accurate descriptive model helps DT better reflect the physical entity. There are two types of data synchronization, real-time synchronization and no-real-time synchronization. The core of the real-time synchronization between DT and its physical counterpart is to connect the digital model controller with the physical PLC (Programmable Logic Controller) [29]. For the no-real-time synchronization, it can be done with an intermediate database. Edge computing adapts well with two types of data synchronization by providing sufficient computing resources for real-time synchronization and local database for no-real-time synchronization.

3.3 Summary

In general, edge computing brings computation resources and data storage closer to the location of data being gathered. The adoption of edge computing on the one hand can make real-time tasks do not suffer latency issues, and reduces the data transmission pressure by processing real-time data before transmitting to the cloud layer on the other hand. The responsiveness, agility and privacy of DT will be enhanced by edge computing [30].

4. DATA MAPPING WITH EXPRESS SCHEMA

DT is the digital replica located in the cyber world of the physical entity. The data collected from the physical world mapping to the cyber world will drive DT to update in real-time, keeping it synchronized with the physical counterpart. According to the previous discussion, a descriptive data model is important for data mapping. In this section, a data mapping strategy based on EXPRESS language is proposed for DT implementation of a machine tool.

4.1 EXPRESS

ISO standard STEP (Standard for the Exchange of Product Model Data) was defined for the information transfer during the product lifecycle from design to manufacturing. EXPRESS and its graphical representation EXPRESS-G were defined as information model specification language in STEP, which makes it more suitable for DT modeling in manufacturing [31]. EXPRESS includes object-oriented and procedural concepts for the description of information models [31]. Implementing an EXPRESS model on a database repository is feasible, which allows it to help realize the data mapping of DT in edge devices.

4.2 EXPRESS schema for machine tool

An EXPRESS schema designed for the DT of machine tool is applied for the illustration of the working mechanism of data mapping with EXPRESS. The corresponding EXPRESS-G diagram of the machine tool DT is shown in Fig. 2. The structure of the schema is based on the Digital Twin framework for manufacturing, a draft ISO standard [32]. The information includes static and dynamic features. The attribute that remains constant throughout the machining process is the static feature of the equipment, such as identification, characteristics, schedule, and description. The attribute that is changeable during the machining process is a dynamic feature, including status, location, report, and dynamic description.

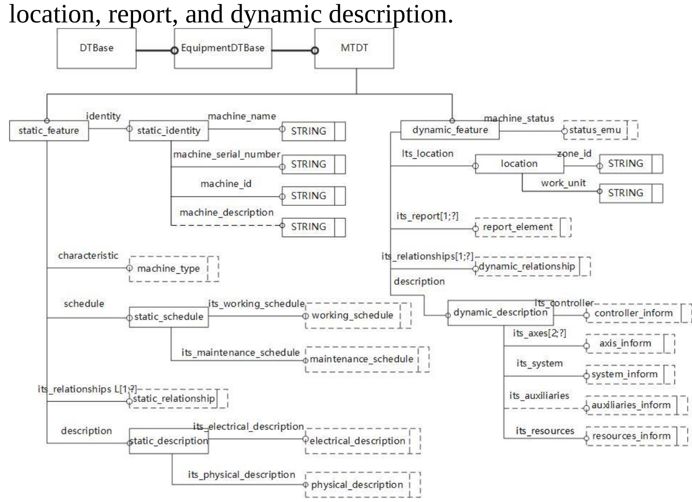

FIGURE 2: EXPRESS-G DIAGRAM OF THE MACHINE TOOL DT

4.3 Data mapping in the edge

Data mapping refers to the process of translating the data in one schema into data conforming to the schema of another [33]. In order to make the data mapping process automated and accurate, mapping transformation scripts based on the source schema and target schema should be generated. The process of data mapping with EXPRESS schema is shown in Fig. 3. It can be divided into two steps, mapping transformation scripts generation and data transform.

1) Mapping transformation scripts generation

A mapping is an expression to describe how the data in one format is related to the data in another format. The transformation script, generated from a set of mappings that is extracted from the schemas of both data source and target, translates a set of data instances of the source into the target schema data representation. The EXPRESS schema provides detailed information to generate the mapping set about the DT, which is the target format in the process of data mapping in DT implementation, such as the data type of the elements and value range.

There are two categories of mapping according to the data processing method during data transform, syntax mapping and semantic mapping.

- *Syntax mapping*: It is a kind of 'word-to-word translation', where the sampled data can be directly mapped to the target data model without further calculation or analysis. Take the attribute to present machine tool status for example. The machine tool status information gathered from the physical entity can be transformed to conform to the corresponding attribute of the DT model through syntax mapping.
- *Semantic mapping*: In semantic mapping, the mapping transformation scripts need an unambiguous and semantic understanding of both source and target data model. Therefore, it can handle the DT model attributes that are not provided directly by the physical entity, through the calculation and analysis of the collected data. The amount of computing resources required for the semantic mapping of each attributes is different. Some attributes in the DT model needs to be obtained by a simple calculation. For example, many CNC systems do not directly provide the data of axis acceleration. To get the value of the axis acceleration, a differential operation needs to be performed on a set of the actual speed value collected from the CNC system. There are also some more complicated situations. In predictive maintenance, anomaly detection is a key process. The result of the anomaly detection can help the DT updated upwards, and provide countermeasures to the physical device downwards. To get the result, the anomaly detection model needs to be derived from real-time data, which will take up more computing resources. Hence, the semantic mapping for DT implementation requires not only a certain amount of computing resources, but also the ability to allocate computing resources flexibly.

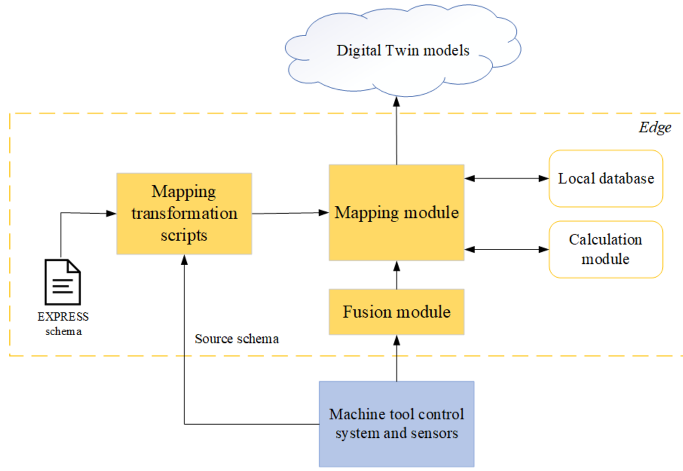

FIGURE 3: DATA MAPPING STRATEGY WITH EXPRESS SCHEMA

2) Data transform

With the mapping transformation scripts, data transform can be performed for the data mapping in the edge. As shown in Fig. 3, raw data are gathered from the machine tool controller system and various sensors. After data fusion, the generated clean and accurate data will be sent to the mapping module for data mapping with the help of mapping transformation scripts. Edge computing provides sufficient computing resources for the running of scripts, and its storage capacity can meet the different real-time requirements of DT for synchronization. For real-time synchronization, the edge device will sand the result of data mapping directly to the cloud. For no-real-time synchronization, the local database will store some mapping results to reduce the pressure of network transmission.

5. CASE STUDY

In order to verify the proposed data mapping method, a case study for a 3-axis milling machine based on edge-computing is presented. In the case study, the machine tool is connected to the DT through MTConnect, which is a widely used communication protocol for machine tools. The implementation of MTConnect is based on HTTP as the transport protocol and XML as the language for encoding [34]. The architecture model of MTconnect is shown in Fig. 4. The client application obtains information of the equipment from the agent by sending MTConnect requests. There are four types of MTConnect requests that can be issued by the client to ask for different information from the equipment. The *probe* request is to acquire the machine tool metadata. The *current* request is to get the current state of the machine tool. The *sample* request is to request a series of data values from the buffer in the agent. The *asset* request is to get information related to MTConnect assets that have been published to the agent.

According to the process of data mapping strategy with EXPRESS schema in Fig. 3, the EXPRES schema that describes the data model of DT and the probe file that describes the metadata of the machine tool are analyzed to generate the mapping module. Then the real-time data of the machine tool is collected based on the MTConnect protocol and the analysis module maps these real-time data to the cloud layer of DT accordingly. A database can be used to store the data of DT. The structure of the case study is shown in Fig. 5.

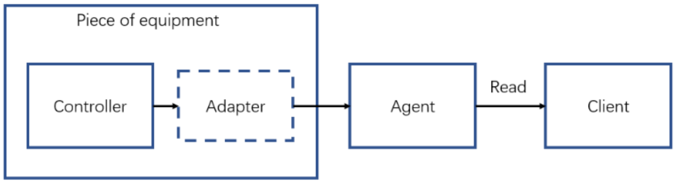

FIGURE 4: MTCONNECT ARCHITECTURE MODEL (ADAPTED FROM [34])

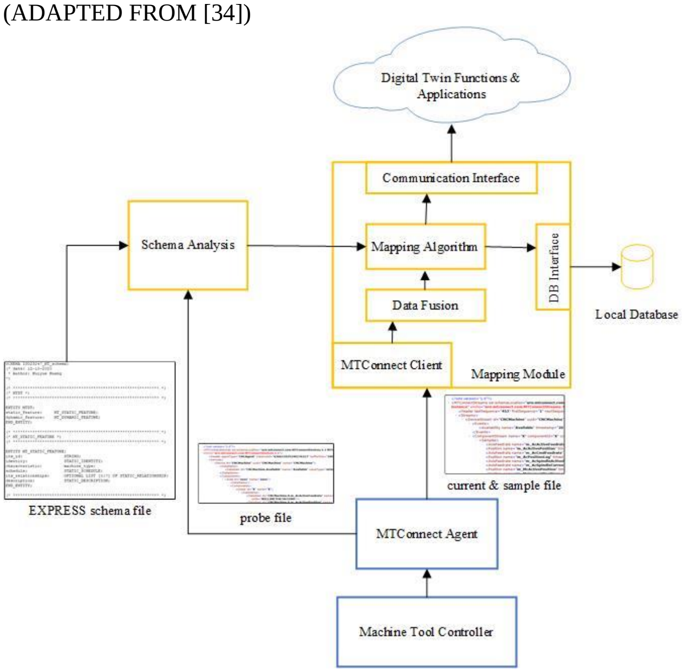

FIGURE 5: STRUCTURE OF THE CASE STUDY

To summarize, the adoption of edge computing in the implementation of DT reduces the pressure on network transportation, providing useful information to the cloud layer of DT. However, the case study is limited to functions such as data mapping in the edge layer of DT for verification; other functions are to be developed.

6. CONCLUSIONS

This paper systematically illustrated the application of edge computing in DT implementation which would be meaningful in responding to a pandemic like COVID-19. The research mainly focuses on the following aspects. First, an application architecture of edge-based DT is proposed. Second, the functions of edge computing in the implementation of DT are discussed; these include data fusion, data storage and privacy, real-time tasks processing, and data mapping. Third, the data mapping strategy based on the EXPRESS schema is discussed and a case study is carried out to verify the method. However, there are still many works that need to be done to fulfill the study of edge computing-based DT. Future work will focus on these aspects: 1) study the communication mechanism between the edge and the cloud within DT; 2) research on data mapping from DT to the physical counterpart.

ACKNOWLEDGEMENTS

The authors are grateful to Dr. Yuqian Lu for his guidance and Tang Ji for his help. Huiyue Huang is supported by China Scholarship Council.

REFERENCES

[1] “Industry 4.0: Fourth Industrial Revolution Guide to Industrie 4.0,” i-SCOOP. Available: https://www.i-scoop.eu/industry-4-0/. [Accessed: 13-Nov-2020].

[2] Xu, Zhangyang; Zhang, Yanqi; Li, Haoyuan; Yang, Weijing and Qi, Quan. “Dynamic resource provisioning for cyber-physical systems in cloud-fog-edge computing.” *Journal of Cloud Computing* Vol. 9 No. 1 (2020): pp. 32. DOI 10.1186/s13677-020-00181-y.

[3] Jazdi, N. “Cyber physical systems in the context of Industry 4.0.” *2014 IEEE International Conference on Automation, Quality and Testing, Robotics*. pp. 1-4. DOI 10.1109/AQTR.2014.6857843.

[4] Lu, Yuqian; Liu, Chao; Wang, Kevin I-Kai; Huang, Huiyue and Xu, Xun. “Digital Twin-driven smart manufacturing: Connotation, reference model, applications and research issues.” *Robotics and Computer-Integrated Manufacturing* Vol. 61 (2020) pp. 101837. DOI 10.1016/j.rcim.2019.101837.

[5] Grieves, Michael and Vickers, John. “Digital Twin: Mitigating Unpredictable, Undesirable Emergent Behavior in Complex Systems.” *Transdisciplinary Perspectives on Complex Systems: New Findings and Approaches*. Springer International Publishing, Cham (2017): pp. 85-113.

[6] “ The Difference Between a Digital Thread and a Digital Twin,” Challenge Advisory. https://www.challenge.org/insights/digital-twin-and-digital-thread/ (accessed Nov. 14, 2020).

[7] Tao, Fei; Qi, Qinglin; Wang, Lihui and Nee, A. Y. C. “Digital Twins and Cyber–Physical Systems toward Smart Manufacturing and Industry 4.0: Correlation and Comparison.” *Engineering* Vol. 5 No. 4 (2019): pp. 653-661. DOI 10.1016/j.eng.2019.01.014.

[8] Kraft, Edward M. “The Air Force Digital Thread/Digital Twin - Life Cycle Integration and Use of Computational and Experimental Knowledge.” *54th AIAA Aerospace Sciences Meeting*. American Institute of Aeronautics and Astronautics (2016).

[9] “ISO/DIS 23247-1 Automation systems and integration — Digital Twin framework for manufacturing — Part 1: Overview and general principles,” ISO. https://www.iso.org/cms/render/live/en/sites/isoorg/contents/data/standard/07/50/75066.html (accessed Nov. 28, 2020).

[10] “Why IoT is the Backbone for Digital Twin.” https://www.ptc.com/en/blogs/corporate/iot-digital-twin (accessed Nov. 15, 2020).

[11] Saad, Ahmed; Faddel, Samy and Mohammed, Osama. “IoT-Based Digital Twin for Energy Cyber-Physical Systems:

Design and Implementation." *Energies* Vol. 13 No. 18 (2020): pp. 4762. DOI 10.3390/en13184762.

[12] Saad, Ahmed; Faddel, Samy; Youssef, Tarek and Mohammed, Osama A. "On the Implementation of IoT-Based Digital Twin for Networked Microgrids Resiliency Against Cyber Attacks." *IEEE Transactions on Smart Grid* Vol. 11 No. 6 (2020): pp. 5138-5150. DOI 10.1109/TSG.2020.3000958.

[13] Fuller, Aidan; Fan, Zhong; Day, Charles and Barlow. "Digital Twin: Enabling Technologies, Challenges and Open Research." IEEE Access Vol. 8 (2020): pp. 108952-108971. DOI 10.1109/ACCESS.2020.2998358.

[14] Canedo, Arquimedes. "Industrial IoT lifecycle via digital twins." *Proceedings of the Eleventh IEEE/ACM/IFIP International Conference on Hardware/Software Codesign and System Synthesis.* Article No.: 29 pp. 1. New York, NY, USA, October 2016. DOI 10.1145/2968456.2974007.

[15] Leiva, Conrad. "Demystifying the Digital Thread and Digital Twin Concepts." Available: https://info.ibaset.com/hubfs/Demystifying_the_Digital_Thread_and_Digital_Twin.pdf. (accessed Nov. 14, 2020).

[16] Hu, Liwen; Nguyen, Ngoc-Tu; Tao, Wenjin; Leu, Ming C; Liu, Xiaoqing Frank; Shahriar, Md Rakib and Sunny, S. M. Nahian Al. "Modeling of Cloud-Based Digital Twins for Smart Manufacturing with MT Connect." *Procedia Manufacturing* Vol. 26 (2018): pp. 1193-1203. DOI 10.1016/j.promfg.2018.07.155.

[17] Hoebert, Timon; Lepuschitz, Wilfried; List, Erhard and Merdan, Munir. "Cloud-Based Digital Twin for Industrial Robotics." *Industrial Applications of Holonic and Multi-Agent Systems. HoloMAS 2019. Lecture Notes in Computer Science* Vol. 11710. Springer, Cham (2019): pp. 105-116. DOI 10.1007/978-3-030-27878-6_9.

[18] Brecher, C.; Buchsbaum, M. and Storms, S. "Control from the Cloud: Edge Computing, Services and Digital Shadow for Automation Technologies." *2019 International Conference on Robotics and Automation (ICRA)*. Montreal, QC, Canada. 20-24 May 2019. DOI 10.1109/ICRA.2019.8793488.

[19] Pushpa, J. and Kalyani, S. A. "Chapter Three - The fog computing/edge computing to leverage Digital Twin." *Advances in Computers*. Elsevier (2020). pp. 51-77.

[20] Chen, Baotong; Wan, Jiafu; Celesti, Antonio; Li Di; Abbas, Haider and Zhang, Qin. "Edge Computing in IoT-Based Manufacturing." *IEEE Communications Magazine* Vol. 56 No. 9 (2018): pp. 103-109. DOI 10.1109/MCOM.2018.1701231.

[21] Qi, Qinglin; Zhao, Dongming; Liao, T. Warren and Tao, Fei. "Modeling of Cyber-Physical Systems and Digital Twin Based on Edge Computing, Fog Computing and Cloud Computing Towards Smart Manufacturing." *ASME 2018 13th International Manufacturing Science and Engineering Conference* Paper No.: MSEC2018-6435, V001T05A018. College Station, Texas, USA. 18-22, 2018. DOI 10.1115/MSEC2018-6435.

[22] Bao, Jinsong; Li, Jie and Zhang, Jie. "Real-time task processing method based on edge computing for spinning CPS." *Frontiers of Mechanical Engineering* Vol. 14 No. 3 (2019): pp. 320-331. DOI 10.1007/s11465-019-0542-1.

[23] Immerman, Graham. "What is the Difference Between MTConnect and OPC UA?" https://www.machinemetrics.com/blog/what-is-the-difference-between-mtconnect-and-opc-ua (accessed Nov. 20, 2020).

[24] Loshin, David. "15 - Parsing and Standardization." *The Practitioner's Guide to Data Quality Improvement*. Morgan Kaufmann, Boston (2011): pp. 261-278.

[25] Ridzuan, Fakhitah and Zainon, Wan Mohd Nazmee Wan. "A Review on Data Cleansing Methods for Big Data." *Procedia Computer Science* Vol. 161 (2019): pp.731-738. DOI 10.1016/j.procs.2019.11.177.

[26] Mei, Yaxin; Wang, Tian and Ma, Ying. "Edge-based Differenital Big Data Processing for Sensor-Cloud Systems." *2019 IEEE International Conference on Signal, Information and Data Processing (ICSIDP)*. INSPEC Accession Number: 19892250: pp. 1-6. Chongqing, China, 11-13 Dec. 2019. DOI 10.1109/ICSIDP47821.2019.9173096.

[27] Sha, Kewei; Yang, T. Andrew; Wei, Wei and Davari, Sadegh. "A survey of edge computing-based designs for IoT security." *Digital Communications and Networks* Vol. 6 No. 2 (2020): pp. 195-202. DOI 10.1016/j.dcan.2019.08.006.

[28] Shahbaz Ul Haq, Qamar. "Chapter 1 – Introduction." *Data Mapping for Data Warehouse Design*. Morgan Kaufmann, Boston (2016): pp. 1.

[29] Zheng, Yu; Yang, Sen and Cheng, Huanchong. "An application framework of digital twin and its case study." *Journal of Ambient Intelligence and Humanized Computing*. Vol. 10 No. 3 (2019): pp. 1141-1153. DOI 10.1007/s12652-018-0911-3.

[30] Zhang, Tong; Li, Yikai and Chen, C.L. Philip. "Edge computing and its role in industrial internet: methodologies, applications, and future directions." *Information Sciences*. Vol. 557 (2021): pp. 34-65. DOI 10.1016/j.ins.2020.12.021.

[31] Arnold, Florian and Podehl, Gerd. "Best of Both Worlds – A Mapping from EXPRESS-G to UML." *The Unified Modeling Language. «UML»'98: Beyond the Notation*. Springer, Berlin, Heidelberg (1999): pp. 49-63.

[32] "ISO/DIS 23247-3 Automation systems and integration — Digital Twin framework for manufacturing — Part 3: Digital representation of manufacturing elements," ISO. https://www.iso.org/cms/render/live/en/sites/isoorg/contents/data/standard/07/87/78744.html (accessed Nov. 28, 2020).

[33] Bellahsene, Zohra; Bonifati, Angela; Duchateau, Fabien and Velegrakis, Yannis. "On Evaluating Schema Matching and Mapping." Schema Matching and Mapping. Springer, Berlin, Heidelberg (2011): pp. 253-291.

[34] "MTConnect Standard Part 1.0 – Overview and Fundamentals Version 1.6.0," MTConnect Institute, https://docs.mtconnect.org/MTC_Part_1-0-Overview_and_Fundamentals_1-6-0.pdf (accessed Feb. 20, 2021).