Journal of Manufacturing Systems 66 (2023) 56–70

Contents lists available at ScienceDirect

Journal of Manufacturing Systems

journal homepage: www.elsevier.com/locate/jmansys

ELSEVIER

Technical paper

A multi-access edge computing enabled framework for the construction of a knowledge-sharing intelligent machine tool swarm in Industry 4.0

Chao Zhang [a,b], Guanghui Zhou [a,b,*], Jingjing Li [a], Fengtian Chang [c], Kai Ding [c], Dongxu Ma [a]

[a] School of Mechanical Engineering, Xi'an Jiaotong University, Xi'an 710049, China
[b] State Key Laboratory for Manufacturing Systems Engineering, Xi'an Jiaotong University, Xi'an 710054, China
[c] School of Construction Machinery, Chang'an University, Xi'an 710064, China

ARTICLE INFO

Keywords:
Multi-access edge computing
Digital twin
Intelligent machine tool
Industry 4.0
Knowledge sharing

ABSTRACT

Developing intelligent machine tools has been front and center for manufacturing enterprises to take a step towards intelligent manufacturing in Industry 4.0, which has attracted increasing attention from both academics and industry. Nevertheless, most current approaches focus on the construction of a single digital twin machine tool with limited intelligence due to the lack of data and knowledge accumulated by that machine tool for decision-making support. Consequently, this paper integrates digital twin with multi-access edge computing (MEC) and proposes a novel framework for the construction of a knowledge-sharing intelligent machine tool swarm that supports the secure knowledge sharing across the authorized machine tools in the swarm with ultra-low latency performance. Then, three key enabling methodologies of the framework are introduced from the perspective of digital twin machine tool swarm construction, knowledge-based cloud brain learning, and MEC-enhanced system deployment. Finally, a prototype system is implemented, where its application examples and evaluation experiments demonstrate the feasibility and effectiveness of the proposed approach.

1. Introduction

The in-depth integration of advanced manufacturing technologies with new generation information technologies, such as edge computing, cloud computing, and digital twins, is guiding another paradigm shift in manufacturing, generally known as the fourth industrial revolution [1,2]. The ultimate goal of Industry 4.0 is to achieve the digitalization, networking and intelligence of modern manufacturing systems [3]. Developing an intelligent manufacturing system has been regarded as the key to establish the competitive advantages for manufacturing industry of major countries in the context of Industry 4.0 [4]. As the foundation of manufacturing industry, the level of intellectualization of machine tools has a critical influence on the implementation of the intelligent manufacturing system [5,6]. Therefore, it is of great significance for manufacturing enterprises to take a step towards intelligent manufacturing by developing intelligent machine tools that take advantage of in-depth fusion of cyber-physical systems to achieve a smart and flexible machining process in order to maximize the product quality and throughput, while reducing cost [7,8].

Digital twin (DT), defined as "an integrated multi-physics, multi-scale, probabilistic simulation of a vehicle or system that uses the best available physical models, sensor updates, fleet history, etc., to mirror the life of its flying twin" [9], has been regarded as an effective solution for implementing cyber-physical fusion of machine tools towards intelligent manufacturing [10]. DT has become one of the world strategic technology trends, which attracts more and more attention from both industry and academics. Nowadays, DT has achieved initial success in the construction of a single smart machine tool with a certain degree of intelligence in, for instance, status perception [11], cutting parameter optimization [12], time-varying error control [13], and predictive maintenance [14]. However, DT application is still in its infancy. How to construct an intelligent and autonomous machine tool is still made difficult by the following unaddressed issues. Firstly, DT modelling is a fundamental but crucial step for the construction of an intelligent machine tool, where, however, a systematic effective modelling method with a quantitative evaluation metric for the construction of an intelligent machine tool is still lacking. Secondly, the cyber-physical fusion of a single machine tool could hardly reach high-level intellectualization restricted by the limited data and knowledge for decision-making support. Thirdly, most DT-based applications require ultra-low latency

* Corresponding author at: School of Mechanical Engineering, Xi'an Jiaotong University, Xi'an 710049, China.
E-mail addresses: superzc@xjtu.edu.cn (C. Zhang), ghzhou@mail.xjtu.edu.cn (G. Zhou), ljj1588@stu.xjtu.edu.cn (J. Li), ftchang@chd.edu.cn (F. Chang), kding@chd.edu.cn (K. Ding), dongxu_ma@stu.xjtu.edu.cn (D. Ma).

https://doi.org/10.1016/j.jmsy.2022.11.015
Received 23 June 2022; Received in revised form 26 October 2022; Accepted 23 November 2022
Available online 7 December 2022

performance, which is still challenging for the current DT-based machine tool system.

To bridge the gaps, this paper proposes a novel multi-access edge computing (MEC) enabled framework for the construction of a knowledge-sharing DT machine tool (DTMT) swarm towards intelligent manufacturing. The proposed framework could take full advantage of DT for cyber-physical fusion, DTMT swarm for knowledge accumulation and sharing, and MEC for data security assurance and system latency performance improvement. Then, three key enabling methodologies for MEC-enhanced knowledge-sharing DTMT swarm are introduced from the perspective of DTMT construction and evaluation, knowledge-based cloud brain learning, and MEC-enhanced system deployment and operation. Finally, a prototype system is implemented, where its application examples and evaluation experiments demonstrate the feasibility and effectiveness of the proposed approach.

The remainder of the paper is organized as follows. Section 2 introduces the research background and motivation behind the paper. In Section 3, we propose a novel MEC-enabled framework for the construction of a knowledge-sharing intelligent DTMT swarm. Section 4 introduces three key enabling methodologies of the DTMT swarm. Section 5 develops a DTMT swarm prototype based on the proposed framework and technologies. Section 6 presents a case study of the prototype to show the effectiveness of the approach. The conclusion and future work are found in Section 7.

2. Background

This section presents the research background behind the paper, including DT concept and its application in machine tools, and multi-access edge computing. Then, the research gap and motivation of the paper are summarized.

2.1. DT machine tools

The ultimate goal of Industry 4.0 is to develop an intelligent manufacturing system that takes advantage of the orchestration of physical and digital processes within a manufacturing system to create a hyper-flexible, self-adapting manufacturing capability [3]. A DT machine tool takes advantage of in-depth fusion of cyber-physical systems to achieve a smart and flexible machining process, thus serving as the key building block of the intelligent manufacturing system that helps manufacturing enterprises to take a step towards intelligent manufacturing [15,16].

Nowadays, DT machine tool has achieved initial success in the area of status perception [11], cutting parameter optimization [12], time-varying error control [13], predictive maintenance [14], etc. For status perception, Zhao et al. [17] introduced a DT-based on-line machine tool monitoring method, which could estimate the machining status with the accessed data from the CNC system; Tong et al. [18] designed a DT-driven intelligent machine tool, which could acquire multi-sensors data based on MTConnect and monitor condition of the machine tool. For cutting parameter optimization, Zhao et al. [19] presented a DT-enhanced surface roughness stabilization method that could make the machined surface quality stable; Zhou et al. [20] constructed a DT-based five-dimensional model to conduct a comprehensive optimization on a centrifugal impeller considering its machining process and aerodynamic performance. For time-varying error control, Liu et al. [13] proposed a method of the time-varying error prediction and compensation for the movement axis of the CNC machine tool based on DT, to improve the accuracy stability of the hole pitch. For predictive maintenance, Qiao et al. [14] combined DT with a bi-directional gate recurrent unit for machining tool condition prediction; Luo et al. [21] proposed a hybrid predictive maintenance method of CNC machine tool driven by DT, which could fuse milling multi-domain model reflecting the actual operating conditions; Wei et al. [22] proposed a consistency retention method for CNC machine tool DT model, which could realize the update of DT model with the performance attenuation of the machine tool; Mourtzis, Angelopoulos, and Panopoulos [23] designed a edge-computing platform that promoted the utilization of 5G cellular networks for improving equipment predictive maintenance.

The above pioneering works provide a new insight into the conceptual understanding of DT machine tools. Nevertheless, the construction and application of DT-based intelligent machine tools are still in its infancy, which remains several unaddressed issues: (1) a systematic effective modelling method with matched quantitative evaluation metric for the construction of an intelligent machine tool is still lacking; (2) the cyber-physical fusion of a single machine tool could hardly reach high-level intellectualization restricted by the limited data and knowledge for decision-making support; (3) how to ensure the data-security and low-latency performance of the current DT machine tool system is still challenging.

2.2. Multi-access edge computing

Edge computing is an emerging computing infrastructure that bridges the gap between cloud and things by distributing edge nodes and providing storage resources close to end-users or devices, thus processing time-sensitive data near the data generation source at the network edge [24]. Edge computing brings several advantages for DT-based industrial intelligence, such as reduced latency, increased bandwidth, and improved data security [23,25]. MEC is a typical edge computing architecture that is characterized by ultra-low latency and high bandwidth as well as real-time access to radio network information that can be leveraged by applications, while providing a strong data security mechanism.

MEC was firstly proposed by ETSI (European Telecommunications Standards Institute) ISG (Industry Specification Group) in 2014, in an effort to create a standardized and open environment that allows the efficient and seamless integration of applications from vendors, service providers, and third-parties across multi-vendor multi-access edge computing platforms. Recent years, the ETSI ISG has developed a series of standards or specifications for MEC, such as the framework and reference architecture [26], use cases and requirements [27], etc. The above works facilitate the conceptual understanding, software developing and industrial application of MEC. With the arrival of 5G, MEC has attracted more and more attention from both industry and academics [28]. Nowadays, MEC-related researches focus on the improvement of MEC performance from the perspectives of task offloading [29], energy optimization [30], security defense [31], etc. Nevertheless, the application of MEC in real-world industrial scenario is relatively lacking.

With the above observations, it is of great significance to integrate DT with MEC to improve the latency performance of DT systems, which could also provide a new insight into the industrial application of MEC.

2.3. Motivation

Through the above analysis, we have the following observations: (1) the construction of DTMT with high-level intellectualization is still made difficult by the current challenging issues, including the lack of a systematic and effective DT modelling method, limited data and knowledge for decision-making support, and high-latency performance; (2) the convergence of DT with MEC could not only improve the performance of DTMT systems, but also provide a new insight into the industrial application of MEC. Consequently, this paper combines DT with MEC to construct an intelligent, secure, and low-latency knowledge-sharing DTMT swarm to achieve an autonomous and flexible machining process. The DTMT swarm could take full advantage of DT for cyber-physical fusion, DTMT swarm for data/knowledge accumulation and sharing, and MEC for ensuring data security and ultra-low latency performance.

3. MEC-enabled framework

We combine DT with MEC to construct a knowledge-sharing DTMT swarm with high-level intellectualization and ultra-low latency performance to achieve an autonomous machining process (AMP) as well as smart maintenance, repair and operations (MRO). As shown in Fig. 1, the knowledge-sharing DTMT swarm is defined by a three-layer framework, including edge-computing layer, cloud-computing layer, and application layer.

Edge-computing layer is responsible for handling time-sensitive tasks during the machining process with the DTMT swarm and MEC. DTMT swarm consists of a set of DTMTs. Each DTMT performs an autonomous machining process with the in-depth fusion of the physical space (PS), data space (DS), and virtual space (VS). Here, PS acts as an automatic actuator with a CNC machine tool, where a sensor network is embedded to collect the real-time machining data, and a smart gateway is deployed to receive updated NC codes from VS for on-line control of the machine tool. DS parses and stores the real-time data collected from PS via an adapter/agent architecture and IoT (Internet of things) devices. It also provides MTConnect interfaces for other spaces to access data. VS provides a multiscale, multi-physics, and high-fidelity simulation capacity to perceive, then understand, subsequently optimize, and finally control the real-time performance of PS with MEC and cloud brain aided. MEC is characterized by the ultra-low latency and high bandwidth as well as real-time access to radio network information that can be leveraged by applications including real-time database (RTDB), DT models and knowledge models. On the one hand, MEC provides data storage, service caching, and computation offloading capacities for computation-intensive and latency-sensitive tasks in the DTMT swarm. On the other hand, each MEC host with local RAN breakout can enable massively scalable real-time duplex trusted transit delivery of data between things (sensors, control systems, etc.) and cloud that leverages secure and low-latency transactions.

Cloud-computing layer aims to learn a cloud brain from machining data/knowledge with its powerful computing resources, which acts as the brain of the DTMT swarm to handle various machining problems. To this end, a machining database, a dynamic knowledge base, and a set of knowledge models are deployed at this layer. Here, the machining database receives process-related information from RTDB for further analysis of machining process. The dynamic knowledge base is

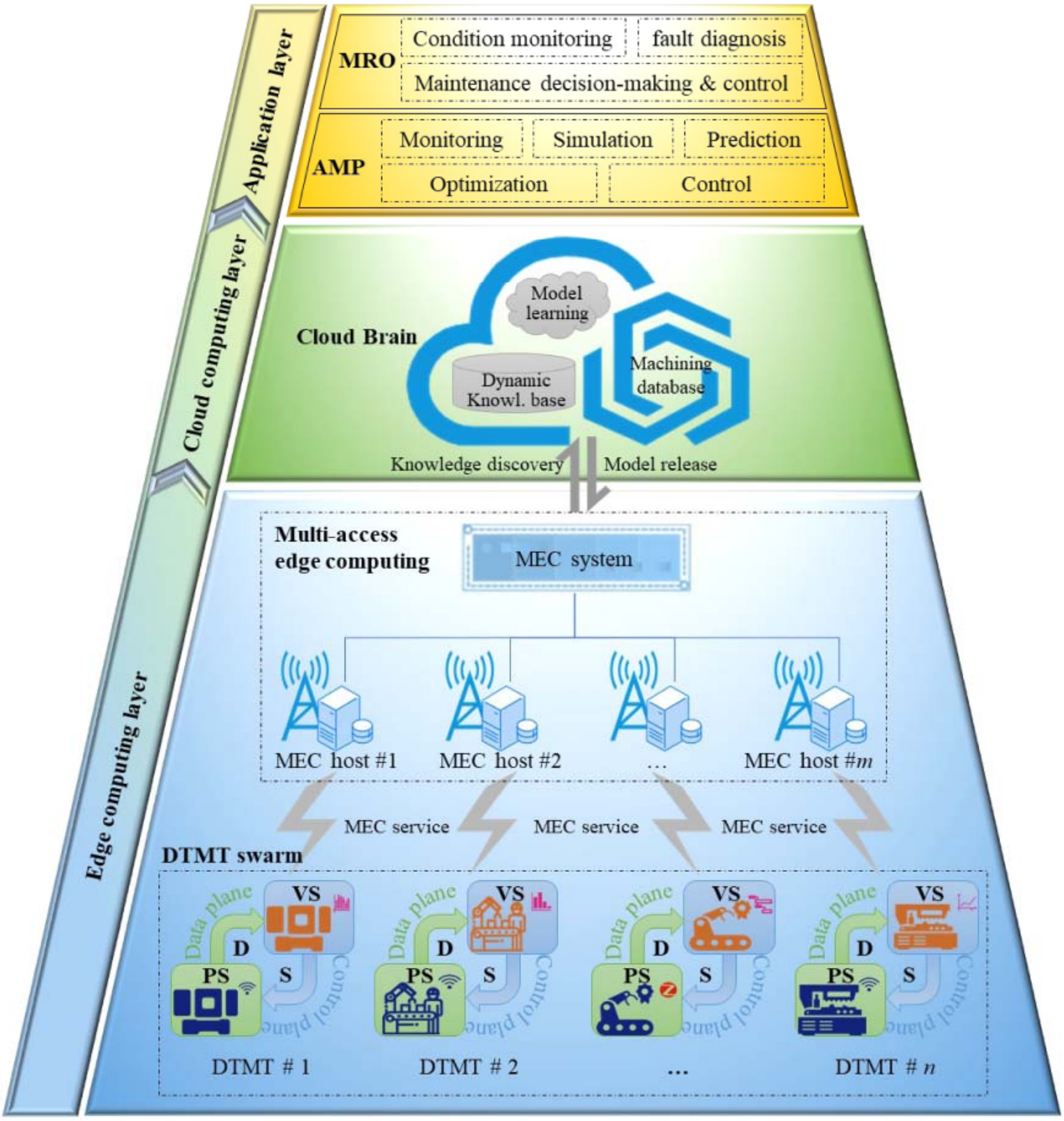


Fig. 1. MEC-enabled framework for knowledge-sharing DTMT swarm.

responsible for knowledge discovery and accumulation from machining data with data mining algorithms. Knowledge models are learned from machining data/knowledge for decision-making support during the machining process [32], which could be shared by all the authorized DTMTs in the swarm.

Application layer takes full advantage of edge computing, cloud computing, and DTMT swarm to provide AMP and MRO services during the full lifecycle of machine tools. AMP could be performed by an intelligent monitoring, simulation, prediction, optimization, and control strategy enabled by the edge-cloud collaborative DTMT swarm and its knowledge sharing mechanism, which could maximize the product quality and throughput, while reducing cost. In addition, the DTMT swarm could enhance MRO through the following three aspects: (1) The DTMT swarm could achieve the real-time machine tool condition monitoring based on the DT data; (2) Fault data/knowledge accumulated by one machine tool could be shared by another similar machine tool in the swarm, thus learning a more accurate and reliable fault diagnosis model for that machine tool; (3) The DTMT swarm could also be integrated with the augmented reality technology and predictive maintenance strategy [23] to achieve the intelligent maintenance decision-making and control. In the following sections, we mainly discuss technologies and implementation tools of the proposed framework for AMP. How to extend the technologies and implementation tools for supporting MRO will be discussed in our future works.

4. Key enabling methodologies

Based on the proposed framework, three key enabling methodologies for the construction of a knowledge-sharing DTMT swarm could be summarized, including DTMT construction and evaluation, cloud brain learning, and MEC-enhanced system deployment and operation.

4.1. DTMT construction and evaluation

Inspired by a popular proverb in ancient China – “two heads are better than one”, the concept of DTMT swarm is proposed for the construction of intelligent machine tools. Actually, experience/knowledge of one machine tool could be applied to the operation and maintenance of another similar machine tool. Besides, the accumulation and sharing of experience/knowledge among machine tools could improve the level of intellectualization of the machine tool. To this end, a set of DTMTs is constructed to constitute the swarm. Each DTMT aims to perform an autonomous machining process with the in-depth fusion and interaction of PS, DS and VS driven by DT, as shown in Fig. 2.

PS is constituted by the hardware of a machine tool system, including the machine tool, CNC system, cutter, and workpiece. PS acts as an automatic actuator controlled by a group of NC codes. During the operating process of PS, the real-time machining data is collected by a CNC API (Application Program Interface) and IoT devices, which is further transported to DS through wired or wireless networks, such as Ethernet, Bluetooth, Zigbee, 4G/5G, etc. PS also receives control programs or advice from VS for the real-time machine tool control.

DS is responsible for the real-time data processing, storage and accessing based on the MTConnect protocol [33]. On the one hand, internal data (such as spindle speed, cutting parameters, etc.) of a machine tool collected from the CNC system is parsed and transported through a

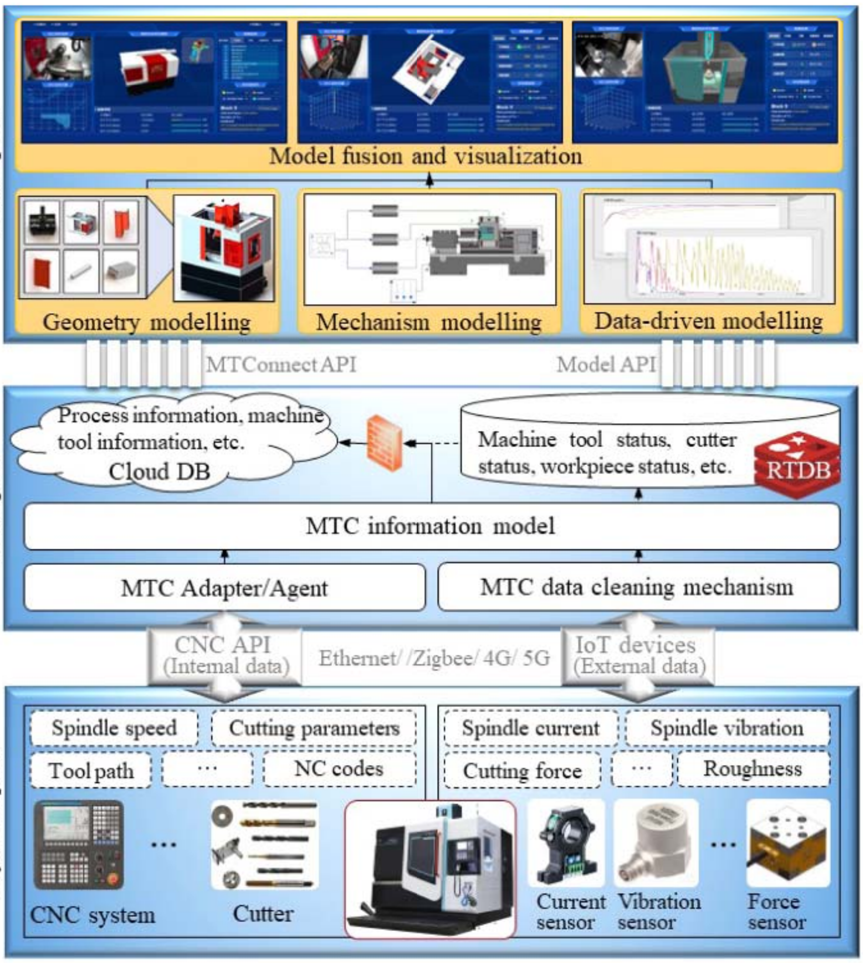


Fig. 2. Multi-level modelling of DTMT.

MTConnect adapter/agent architecture. Here, MTConnect adapter collects data from the CNC system, then filters data that does not vary over time, then turns it into SHDR (Simple Hierarchical Data Representation) data, and finally pushes SHDR data to the agent via socket. MTConnect agent receives SHDR data and turns it into MTConnect metadata, which is then verified and cached through XML interpretation and validation module and data caching module. MTConnect agent also provides access interfaces including probe, stream, asset and error for data sharing through HTTP service. On the other hand, external data (such as spindle vibration, spindle current, etc.) is collected by IoT devices. The isolated internal data and external data are further aggregated into a hierarchical structure with clear semantics and correlation through a MTConnect information model, thus becoming useful information. Among the useful information, the time-varying information is stored in RTDB implemented by Redis based on its subject, such as the machine tool status, cutter status, workpiece status, etc., which could be easily accessed by time-sensitive tasks handled at MEC end. Besides, process related data is transported to cloud database for further analysis.

VS provides a multiscale, multi-physics and high-fidelity simulation model, which is established through four steps including geometric modelling, mechanism modelling, data-driven modelling, model fusion and visualization. Geometric modelling aims to build a visual mapping of the physical object into cyber space with 3D CAD modelling tools like Solidworks and Pro/Engineer, whose complexity and size are reduced through a lightweight tool like PiXYZ Studio to improve the real-time performance of DTMT. Mechanism modelling aims to construct a multi-scale and multi-physics simulation model with the system level modelling and block level modelling. System level modelling defines the detailed requirements, structures, behavior and parametrics of the machine tool via block definition diagrams (BDD) and internal block diagrams (IBD) implemented in Systems Modelling Language (SysML). BDD represents multiple subsystems/components of a machine tool with semantic blocks in a hierarchy structure, while specifying their classification relationships through edges between blocks. IBD describes the internal behavioral logic of the machine tool by specifying input/output relationships and connectors of each block defined in the BDD model. Block layer modelling adds equation-based definitions for each block defined in the IBD model through Modelica language, which obtains mathematics and physics characteristics of multiple subsystems/components of the machine tool to support multiscale, multi-physics and high-fidelity simulation. Data-driven modelling aims to learn a model for a specific manufacturing application scene, such as the roughness prediction and surface quality control, through mining and analyzing real-world data from PS and simulation data from VS with machining learning algorithms. An example of data-driven modelling for roughness prediction could be founded in our previous work [34]. The geometric model, mechanism model, and data-driven model are integrated into a DT model operated on a webpage with a B/S architecture based on the model fusion and visualization strategy.

Considering the uncertainty of DTMT modelling and performance degradation of the physical machine tool, a fidelity evaluation method is defined with the product of three key indexes including *integrity*, *confidence*, and *synchronization*.

Integrity measures the precision of geometric model from the aspects of structural integrity, geometric accuracy, and motion constraints. *Integrity* is defined as:

$$F_i = \frac{n}{N} \times \left(\frac{1}{n}\sum_{i=1}^{n} Ga_i\right) \times \left(\frac{1}{n}\sum_{j=1}^{n} Mc_j\right) \tag{1}$$

where N is the number of components of the physical machine tool that tend to be modeled, n is the number of components contained in its corresponding geometric model; Ga_i is the geometric accuracy of the i-th component, Ga_i= 1 if the component meets its tolerance requirement, and zero otherwise; Mc_j represents the motion constraint of the i-th component, Mc_j= 1 if the motion constraint is reasonable compared with its physical machine tool, and zero otherwise.

Confidence evaluates the simulation accuracy of mechanism models and uncertainty of data-driven models by comparing the responses between the DT model and its corresponding physical twin (PT) under same excitations. *Confidence* is defined as:

$$F_c = \frac{1}{m}\sum_{i=1}^{m}\left(1 - \frac{|f_{PT}(x_i) - f_{DT}(x_i)|}{f_{PT}(x_i)}\right) \tag{2}$$

where m is the number of key outputs of the DT and PT; x_i is the excitation of DT and PT for the i-th output; $f_{PT}(x_i)$ and $f_{DT}(x_i)$ are the responses of DT and PT, respectively, under the same excitation x_i.

Synchronization evaluates the real-time performance of DTMT with the time varying between responses from the PT end and DT end. *Synchronization* is defined as:

$$F_s = \frac{1}{m}\sum_{i=1}^{m} 2^{-\mathrm{ReLU}\left(\psi\left(\tau_i^{DT} - \tau_i^{PT}\right) - \psi_i\right)} \tag{3}$$

where τ_i^{DT} and τ_i^{PT} is the response time of the i-th excitation corresponding to DT end and PT end, respectively; ReLU(x)=max(0, x) is a Rectified Linear Unit [35]; $\psi_i \in \{1, 2, 3, 4\}$ is the expected response time level for the i-th excitation, which is flexible to regulate for different manufacturing application scenes. $\psi(t)$ is a predefined algorithm that calculates the response time level for a specific excitation.

4.2. Cloud brain learning

Cloud brain takes full advantage of cloud-computing resources and machine learning algorithms for knowledge discovery and knowledge models/data-driven models learning, thus equipping DTMT swarm with powerful cognitive learning and decision-making capacities.

Process knowledge is the key to formalizing the cloud brain, which helps each DTMT understand problems, in order to develop good judgements to make decisions about these problems. Consequently, machining learning-enabled cloud brain learning is proposed to discover and accumulate process knowledge from machining data with the following three steps including data pre-processing, knowledge mining and knowledge graph construction.

Data pre-processing extracts valid information from unstructured machining data, such as process cards and process manuals that are usually presented in the form of pictures, with an image text recognition method - optical character recognition (OCR) [36]. Here, the valid information is formally defined as:

$$D = \{d_1, d_2 \ldots, d_i, \ldots, d_n\}, d_i = \{P_i, S_i, T_i\} \tag{4}$$

where the valid information D consists of n subsets of process information, and d_i is the *i-th* subset represented by the category data P_i, sequential data S_i, and non-sequential data T_i. Here, P_i characterizes the processing features, parts or application scopes associated with d_i; S_i is used for sequential process knowledge mining, such as process route; T_i is used for non-sequential process knowledge mining, such as process decision-making support knowledge.

Knowledge mining aims to extract process knowledge from sequential data S_i and non-sequential data T_i. Given a set of sequential data S_i= {N_1, N_2, …, N_k}, sequential process knowledge is discovered through mining the frequent subsequence S_{pi} of S_i with the extended PrefixSpan algorithm [37], which is defined as:

$$S_{pi} = \{N_m, N_{m+1}, ., N_n\}, 1 \le m \le n \le k \tag{5}$$

$$f(S_{pi}) \ge F_S \tag{6}$$

where S_{pi} is the contiguous subsequence of S_i; $f(S_{pi})$ is the frequency of S_{pi} in the input data d_i, and F_S is the frequency threshold.

Based on Eq. (5) and Eq. (6), the extended PrefixSpan algorithm for sequential process knowledge mining is defined as in Table 1.

Given a set of non-sequential data $T = \{T_1, T_2, \ldots, T_l\}$ that belongs to the same or similar category data P, non-sequential process knowledge is discovered through mining the frequent tuple T_{pi} of T_i with the extended FP-growth algorithm [38], which is defined as:

$$T_{pi} \subseteq T_i \tag{7}$$

$$f(T_{pi}) \geq F_T \tag{8}$$

where T_{pi} is a subset of T_i, $f(T_{pi})$ is the frequency of T_{pi} in T, and F_T is the frequency threshold.

Based on Eq. (7) and Eq. (8), the extended FP-growth algorithm for non-sequential process knowledge mining is defined as in Table 2.

The mined S_{pi} and T_{pi} correspond to process route decision-support knowledge and process parameter decision-support knowledge, respectively. The above knowledge is associated with its corresponding process features or parts and formalizes the knowledge graph with a knowledge graph construction process based on Neo4j graph database, which could support the intelligent decision-making of the DTMT swarm. In addition, knowledge models and data-driven models are learned from machining data/knowledge for machining problem handling, where the detailed implementation steps could be founded in Section 5.

4.3. MEC-enhanced system deployment and operation

The constructed DTMT swarm and cloud brain are deployed based on a MEC-enhanced latency-aware deployment architecture with three levels including device level, MEC level and cloud level, as depicted in Fig. 3.

The device level conducts an autonomous machining process enabled by the lightweight DTMT swarm. Each lightweight DTMT in the swarm corresponds to partial functions of DTMT that could operate on low-computing performance user equipment (UE) including industrial computers, Kanban system, smart phones, etc., through the webpage. Here, a physical machine tool and its corresponding geometric model operated on UEs are connected via real-time status data cached in the MTConnect information model for machining process visualization. Besides, each geometric model operated on UEs could connect to a local MEC host at MEC level through IP address or FQDN (fully qualified domain name), which subscribes simulation or analysis results from mechanism models and data-driven models for device's status monitoring and prediction, while receiving control orders or programs from knowledge models for device control.

MEC level provides data storage, service caching and computation offloading capacities for computation-intensive and latency-sensitive tasks at the device level with a group of MEC hosts, where RTDB, DT models (DTMs) including mechanism models and data-driven models, and knowledge models (KMs) are deployed as MEC applications. A MEC host, deployed near the machine tools, contains a MEC platform, a virtualization infrastructure, and the compute, storage, and network resources for MEC applications in virtual machines or containers. The MEC platform offers a secure environment where the above MEC applications may, via RESTful APIs, consume MEC services, while offering real-time perception, simulation, optimization, and control services that could be consumed by other MEC applications, lightweight DTMTs and cloud resources. Virtualization infrastructure in the MEC host includes a data plane that provides the execution environment for the traffic rules received by the MEC platform, and routes the traffic among applications. Besides, each MEC host with local RAN breakout can enable massively scalable real-time duplex trusted transit delivery of data from MEC to cloud for further analysis.

Cloud level provides powerful computing resources where cloud machining database, knowledge discovery model, and enterprise information systems such as MES (manufacturing execution system), ERP (enterprise resource planning), CRM (customer relationship management), etc., are deployed, and data-driven models or KMs learning process is conducted. Here, the machining database collects real-world data and simulation data from RTDB and DTMs, respectively, on which a knowledge discover model is conducted to learn machining process-related knowledge that dynamically updates the dynamic knowledge base. On that basis, a set of data-driven models and KMs are learned from knowledge, where the learned computing intensive KMs are deployed at the cloud server for the decision-making support of time-insensitive tasks, while time-sensitive KMs and DDMs are released to MEC level for the decision-making support of time-sensitive tasks in PS and VS. In addition, enterprise information systems are deployed at cloud server for upper-level production preparation.

Table 1
The extended PrefixSpan algorithm for sequential process knowledge mining.

Input: Sequential data S_i, frequency threshold F_S.
Output: Frequent subsequence S_{pi}.
1: Find all prefixes of length 1 in the input sequence data S_i with the corresponding projection data set C_{pre}
2: Count the frequency $f(pre)$ of pre
3: Remove pre whose frequency is less than F_S
4: **for each** pre **do**
5:   **for each** S_{pre} in C_{pre} **do**
6:     **for each** I in S_{pre} **do**
7:       $f(I)=f(I)+1$
8:       **if** $f(I) \geq F_S$ **then**
9:         $pre \leftarrow pre+I$
10:       **end if**
11:     **end for**
12:   **end for**
13: **end for**
14: **return** pre as S_{pi}

Table 2
The extended FP-growth algorithm for sequential process knowledge mining.

Input: Non-sequential data T_i, frequency threshold F_T.
Output: Frequent subset T_{pi}

for each T in T_i **do**
    for each D in T **do**
        $f(D) = f(D) + 1$
    end for
end for
for each D **do**
    if $f(D) \geq F_T$ **then**
        Add D to the FP-table in descending order of $f(D)$
    end if
end for
for each T in T_i **do**
    for each D in T **do**
        Add D to the FP-tree according to its order in the FP-table
    end for
end for
return T_{pi} according to the FP-tree

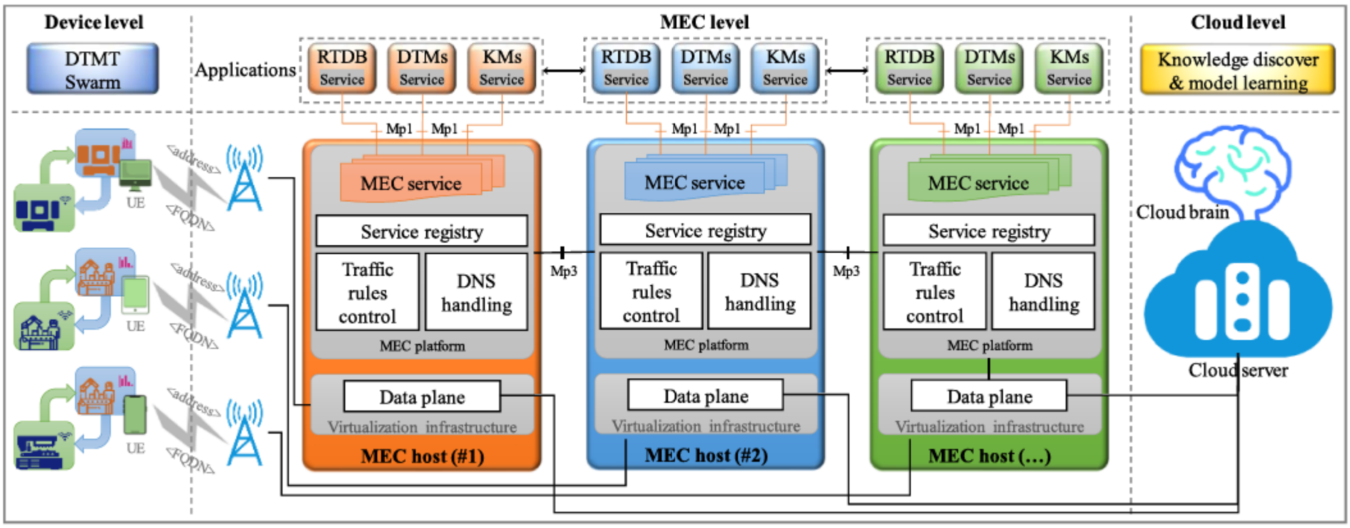


Fig. 3. MEC-enhanced system deployment and operation for DTMT swarm.

Based on the deployment architecture, we develop a MEC-based algorithm to support the adaptive collaboration of device level, MEC level and cloud level. Specifically, let $R_t = \{r_1, r_2, \ldots, r_m\}$ be a set of service requests received from the device level at time t, each r_i labeled with latency requirement information τ_i, minimum computing resource requirement information ζ_i, and arrival rate λ_{ij} to the j-th MEC server. Let $S = \{s_1, s_2, \ldots, s_n\}$ be a set of MEC servers. Let τ be the maximum latency threshold for each service request handling at MEC level. Let x_{ij}, f_{ij}, d_{ij} be the decision variables, where x_{ij} indicates whether request r_i is cached on MEC server s_j, f_{ij} indicates how many computing resources of server s_j is distributed to request r_i, and d_{ij} indicates whether MEC server s_j needs to migrate the request to another MEC server or cloud when processing it. As shown in Table 3, the algorithm takes a request list R_t as input, and outputs x_{ij}, f_{ij}, d_{ij} for each request in the list to decide where to handle it, namely edge or cloud. The alogrithm also optimizes the latency performance of MEC level when handling at edge based on the service caching and computation offloading algorithms embedded in MEC servers.

Through the in-depth integration of device level, MEC level, and cloud level, each machine tool could borrow the wisdom of other machine tools, namely knowledge discovered in historical data of the DTMT swarm, to seek optimal machining parameters for each workpiece, while on-line monitoring, optimizing, and controlling machining process for that workpiece. Besides, MEC also provides the mobility service to handle the situation that UE is a mobile device where the current MEC host of the user session may not be the best choice due to a

Table 3
A MEC-based algorithm for the adaptive collaboration of the system.

Input: $R_t=\{r_1, r_2, \ldots, r_m\}$, $S=\{s_1, s_2, \ldots, s_n\}$, τ_i, ζ_i, λ_{ij}.
Output: Decision variables x_{ij}, f_{ij}, d_{ij}.
Update the service request list R_t at time t
Initial MEC servers S
for each r_i in R_t **do**
  if $r_i \geq \tau$ **do**
    Send r_i to the cloud for further analysis
  else do
    for each s_j in S **do**
      Calculate the service cache priority metrics $m_{ij}= \tau_i \times \lambda_{ij}$
    end for
    Sort servers from the largest to the smallest according to m_{ij}, and formalizes a new server list S'
    for each s' in S' **do**
      Obtain its rest computing resources ζ'
      if $\zeta' \geq \zeta_i$ **do**
        $x_{ij} \leftarrow 1$
      else do
        $f_{ij}=\Phi(f_{ij})$, optimized by the service caching algorithm embedded in MEC host
        $d_{ij}=\Psi(d_{ij})$, optimized by the computation offloading algorithm embedded in MEC host
      end if
    end for
    return x_{ij}, f_{ij}, d_{ij}
  end if
end for

change in the device's location. The MEC system allows the relocation of the user context for a session from one application instance to another running in a MEC host closer to the user, to ensure the optimum lowest latency service to this device. That is, an authorized engineer could connect to the system with UE in different locations of the enterprise for the lowest latency service.

5. Prototype implementation

As shown in Fig. 4, two manufacturing cells located at iHarbor campus and Xingqing campus of Xi'an Jiaotong University (XJTU) are taken as the test bed of the prototype system. Based on the test bed, a knowledge-sharing DTMT swarm prototype is developed with the following three steps, including DTMT swarm construction, cloud brain implementation, and MEC-enhanced prototype deployment.

Fig. 4. Test bed of the prototype system.

5.1. DTMT swarm construction

As shown in Fig. 5, DT models for each of five machine tools are constructed, which are further connected to their corresponding physical machine tools, thus formalizing the DTMT swarm. Here, each DT model is constructed through geometric modelling, mechanism modelling, data-driven modelling, and model fusion and visualization. Geometric modelling builds a visual mapping of the physical machine tool into cyberspace with Solidworks, 3D laser scanner, etc. Mechanism modelling defines detailed requirements and structures of the machine tool with a semantic model constructed by SysML, where equation-based definitions are added to each block of the semantic model based on Modelica language, thus equipping the DT model with multi-scale and multi-physics simulation capacity. Data-driven modelling aims to construct a set of models for specific manufacturing application scenes (the construction of a specific data-driven model could be founded in the following cloud brain implementation subsection), which are released to and operated at MEC level for real-time analysis and control of machine tools. The above constructed models are integrated into a DT model that operates on the webpage for kinematic synchronization and data visualization of the machining process with a B/S architecture.

5.2. Cloud brain implementation

Cloud brain is responsible for knowledge discovery and knowledge models/data-driven models learning with its powerful computing resources. For knowledge discovery, based on our previous work [32], more than 800 pieces of process knowledge with their domain relationships are mined through the proposed algorithm I and algorithm II. The process knowledge is stored in dynamic knowledge bases, which could be continuously updated and accumulated with the increasing of machining data. On that basis, a process knowledge graph is constructed based on the Neo4j graph database, which builds connections between the knowledge and corresponding features or parts, to better support the decision-making of process routes, process parameters, etc. The constructed dynamic knowledge bases and knowledge graphs are shown in Fig. 6.

Cloud brain also trains or updates data-driven models and knowledge models with its powerful storage and compute resources. Here, an intelligent process planning model (iPPM) is learned as the knowledge model that provides process decision-making support for all machine tools in the swarm. The details of learning and application of iPPM could be found in our previous works [39]. A surface quality prediction model (SQPM) is learned as the data-driven model for two lathes, which could also provide reference for learning data-driven model for other machine tools. According to our previous work [34], the roughness of the workpiece during turning process has non-linear relationship with the current value of the main shaft: $R_a = F(f, d, v, I)$, where f, d and v represent the feed rate, depth of cut and spindle speed, respectively; I is the real-time current value of the main shaft. The non-linear relationship could be learned by a machine learning algorithm, namely support vector regression implemented in LibSVM embedded in Matlab 2016b. The architecture of the support vector regression is shown in Fig. 7(a). For learning, the sample is obtained by an orthogonal experiment with 3 factors and 5 levels, by turning No. 6061 aluminum alloy bars in CL20A lathe. Model training process is shown in Fig. 7(b).

5.3. MEC-enhanced prototype deployment

The constructed DTMT swarm and cloud brain are integrated into a knowledge-sharing DTMT swarm with an edge-cloud collaborative architecture enhanced by MEC, thus formalizing the prototype system, as shown in Fig. 8.

Device level consists of five machine tools in two manufacturing cells

Fig. 5. The construction of DTMT swarm.

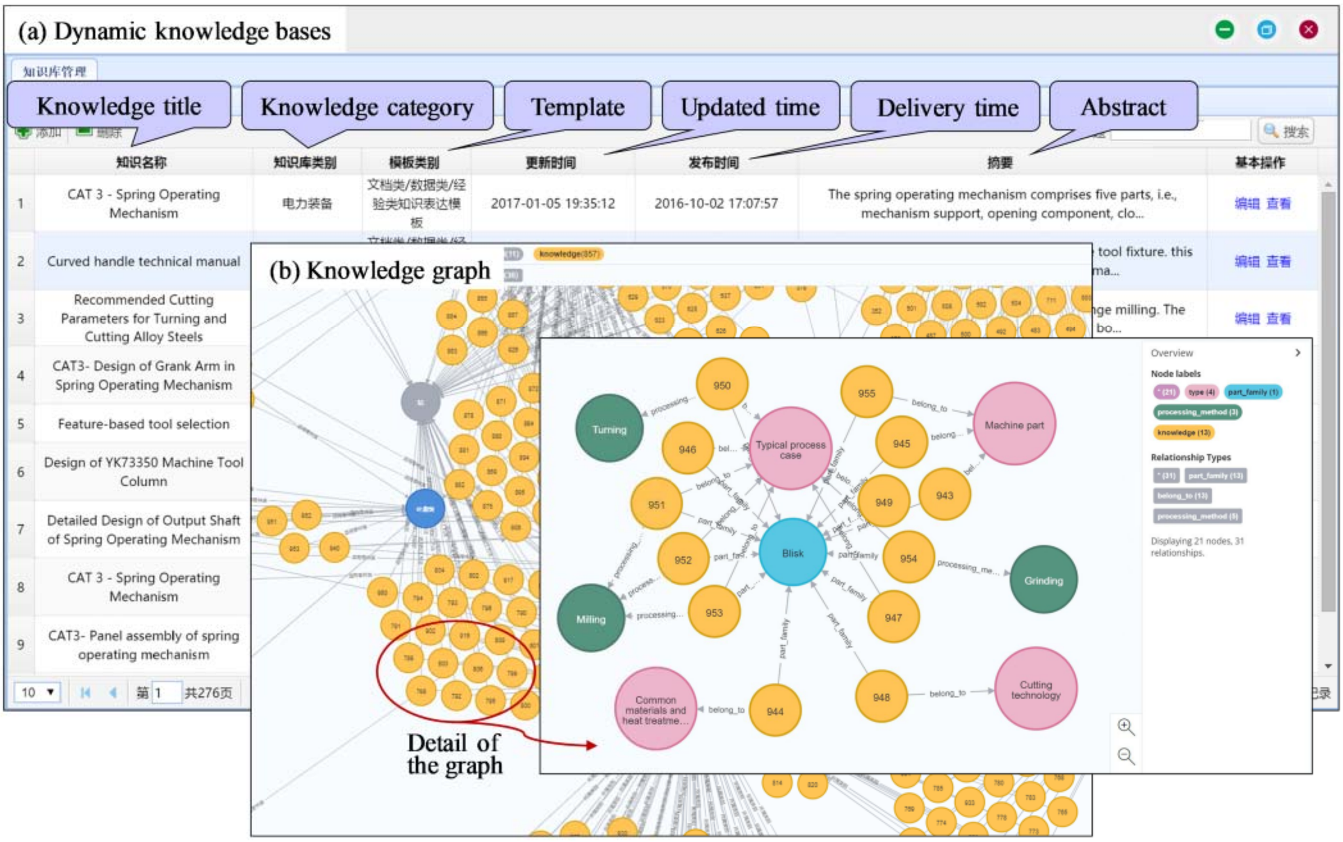


Fig. 6. The constructed dynamic knowledge bases and knowledge graph [32].

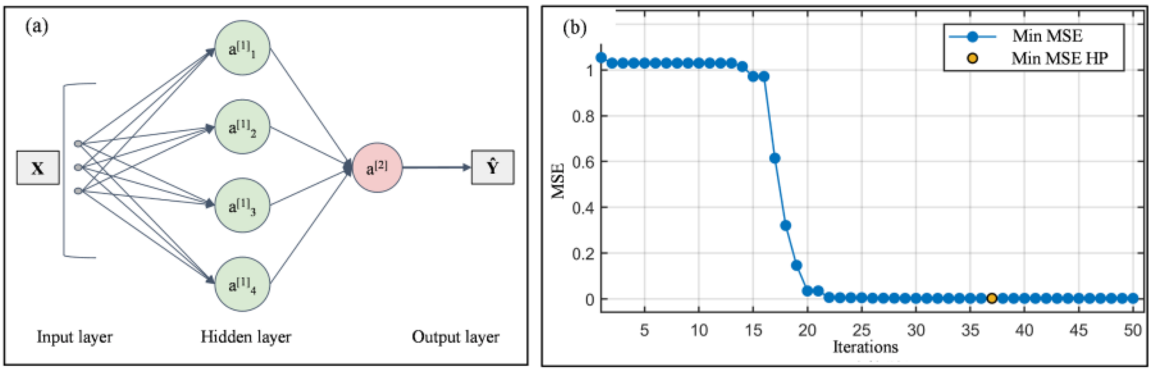


Fig. 7. SQPM learning: (a) the architecture of support vector regression and; (b) training process.

located at iHarbor campus and Xingqing campus of XJTU, which are linked to their corresponding lightweight DT models deployed in UEs through an adapter-agent architecture and IoT devices. The lightweight DT models, namely DT #1 and #2 with their physical machine tools at iHarbor, and DT #3–5 with their physical machine tools at Xingqing, are connected to MEC host #1 and #2, respectively, through IP address. Each lightweight DT model visualizes the machining process based on the real-time data, while subscribing analysis and optimization results from MEC level or cloud level enhance the performance of its corresponding machine tool.

MEC level deploys two MEC hosts that provide data storage, service caching and computation offloading services for MEC applications including RTDB, DT models, and knowledge models based on the algorithm III. For example, MEC host #1 deploys the MM (mechanism model) #1 and DDM (data-driven model) #1 of DT #1, MM #2 and DDM #2 of DT #2, knowledge models (for example, iPPM) and RTDB as MEC applications that offer real-time simulation, analysis, optimization and control service for device level via RESTful APIs. In addition, each MEC host supports the real-time delivery of data from RTDB or DTMs to cloud database for further analysis.

Cloud end provides powerful storage and compute resources and application operation environment where machining database, dynamic knowledge database, enterprise information systems such as MES, ERP, CRM, etc., are deployed. The knowledge discovery and data-driven model/knowledge model learning process are also conducted at cloud level. Here, the learned computing-intensive knowledge models are deployed at cloud server for decision-support of time-insensitive tasks, while time-sensitive knowledge models and data-driven models are released to MEC level for decision-making support of time-sensitive tasks at edge end.

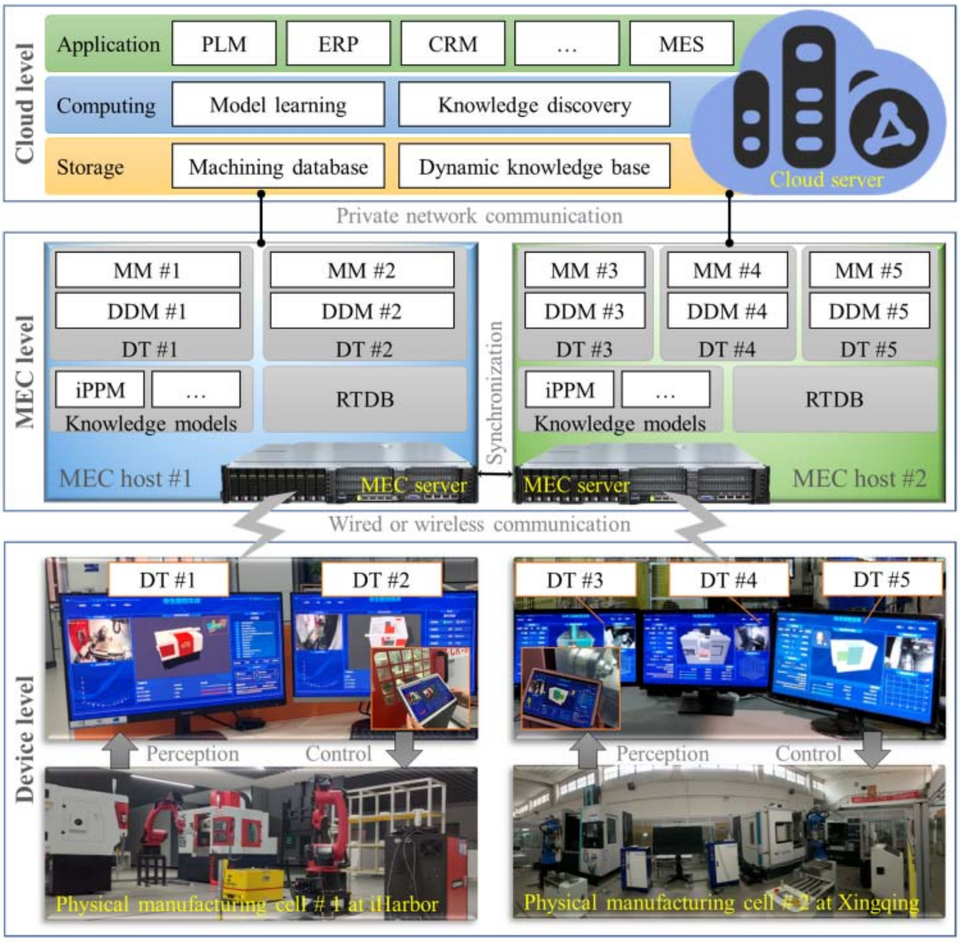


Fig. 8. MEC enhanced prototype system.

6. Case study

This section explores the application scenarios of the prototype system. Then, the effectiveness of the prototype system is demonstrated through the performance evaluation experiments. Finally, the benefits and challenges ahead of the approach are discussed.

6.1. Application scenario of the prototype system

The connection and interaction of device level, MEC level and cloud level contribute to an intelligent machining process with the in-depth integration of cyber-physical spaces and the sharing of knowledge across the authorized DTMTs in the swarm. As shown in Fig. 9, autonomous machining module and auxiliary module are employed to illustrate the secure and autonomous operation mechanism of the prototype system, which are further demonstrated through an application example.

Autonomous machining module illustrates the overall operation logic of the system through three closed operation loops, including the perception-control loop, edge analysis loop and remote analysis loop. The perception-control loop at device level takes advantage of in-depth integration of cyber-physical spaces of DTMT to improve the performance of the physical machine tool through an intelligent perception, simulation, prediction, optimization, and control strategy enhanced by edge analysis and remote analysis loops. In edge analysis loop, each DT operated on authorized UEs could publish login requests to and subscribe MEC services, including data storage, mechanism simulation, status analysis, parameters optimization, and control program mapping from the corresponding MEC host, to handle time-sensitive tasks at the perception-control loop. Remote analysis loop receives login requests from authorized MEC hosts and provides cloud services to handle time-insensitive but computing-intensive tasks at perception-control loop or edge analysis loop. The adaptive collaboration of three loops is enabled by algorithm III.

Auxiliary module illustrates the synchronization, knowledge discovery, and model learning services of the system. The synchronization service allows DT data to synchronize between two authorized MEC hosts, namely MEC host #1 and #2, for which UEs could receive very low-latency DT data no matter where they are located, iHarbor or Xingqing campus of XJTU. This could facilitate the scenario that engineers work across the above two regions. Knowledge discovery and model learning are two off-line processes, where knowledge discovery learns machining process related knowledge from history machining data that dynamically updates the dynamic knowledge base. In addition, data-driven models and knowledge models are learned from machining data/knowledge.

The above operation mechanism is demonstrated through an application example of process planning and online control of a lathe at Xingqing campus of XJTU. The operation flow of the example is shown in the bottom of Fig. 9. The real-world scenario of the example is shown in Fig. 10. For process planning, iPPM at cloud level receives production orders from ERP as input and generates a theoretical process plan that is transformed to MEC host #2, where DT #5 subscribes turning tasks arranged and validates the capacity and availability of the lathe, while optimizing the cutting parameters through a simulation process. The verified and optimized cutting parameters are subscribed by the lightweight DT #5 at device level, which is further mapped into NC codes and distributed to the physical lathe for machining process control.

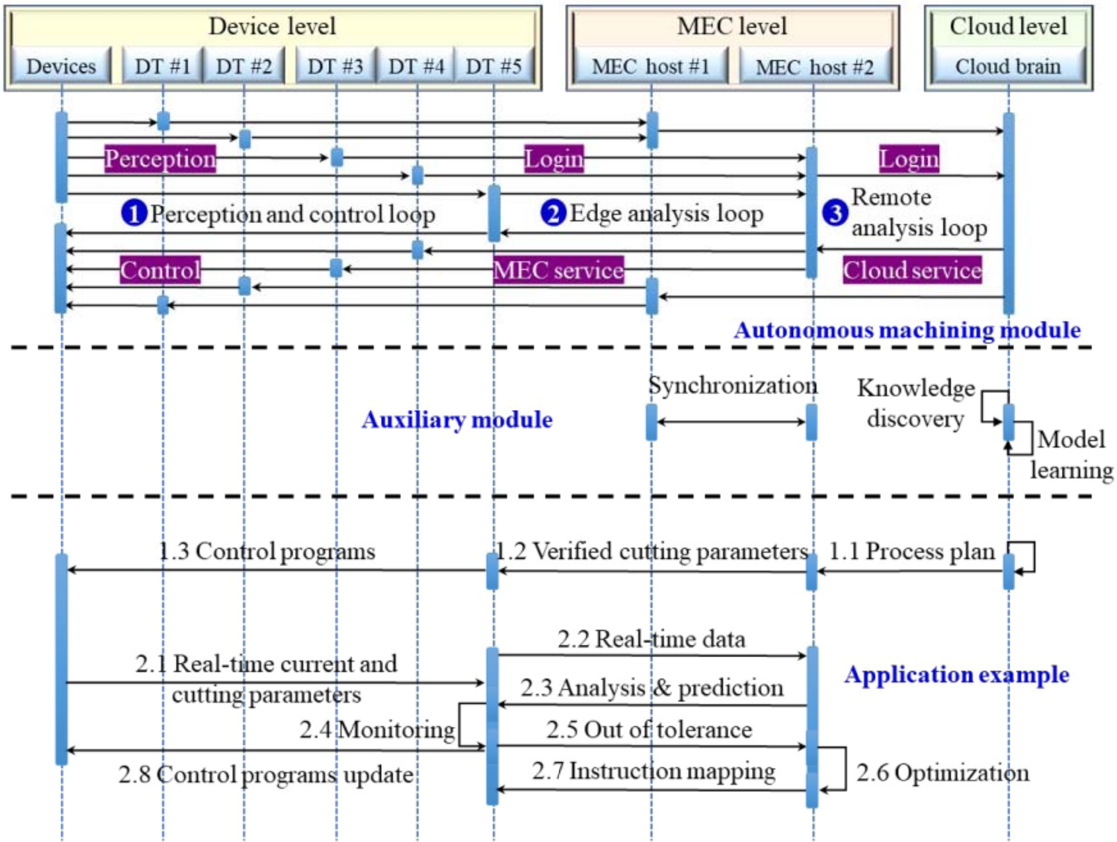


Fig. 9. System operation and application mechanism.

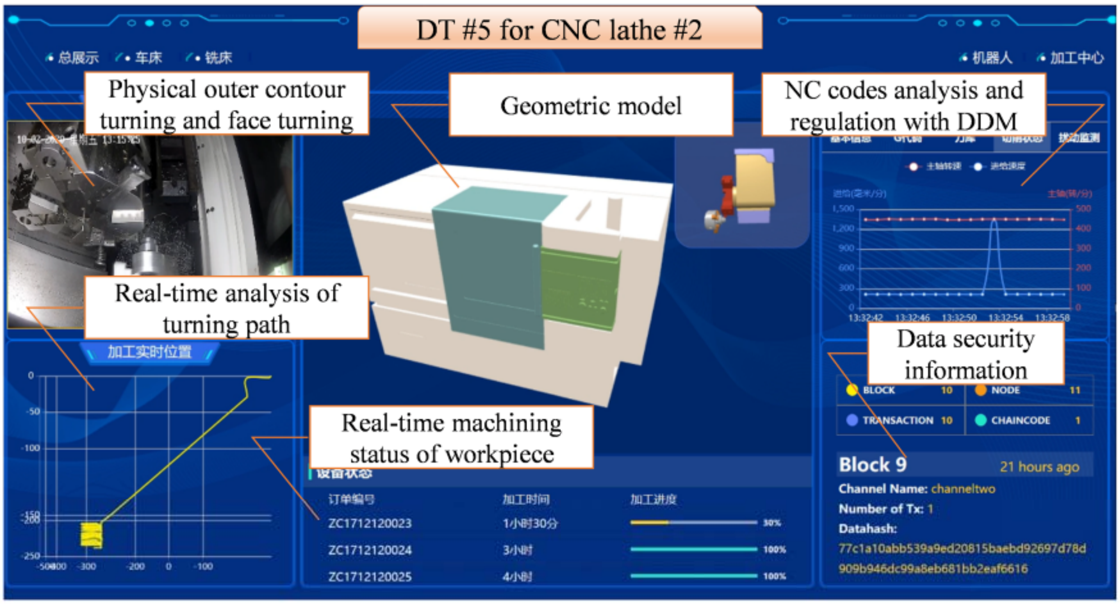


Fig. 10. Example of outer contour and face turning.

During machining process, the lightweight DT #5 perceives, simulates, and monitors the real-time status, such as the cutting path, workpiece status, and surface quality of the lathe with MM #5 and DDM #5 at MEC level. If the surface quality prediction model in DDM #5 evaluates that the roughness is out of tolerance, the optimization process will be carried out, which optimizes the spindle speed, feed rate and depth of cut of the lathe, and maps them into NC codes as updated NC files to control the turning process.

6.2. *Performance evaluation and discussion*

This section first evaluates the key performance of the prototype system with the fidelity of DTMTs. Then, the latency performance of the system is compared with a general system without MEC deployed, where

MEC servers are replaced by a set of normal edge servers.

The fidelity of each of five DTMTs is evaluated from the perspective of *integrity*, *confidence* and *synchronization* with the proposed evaluation approach. Experimental results indicate that the DTMT swarm obtains a very high average comprehensive accuracy of 95.24%. Specifically, taking DT #5 as an example, its fidelity is evaluated through the following three steps. Firstly, the integrity of DT #5, namely $F_i = 1$, is obtained as the visual twin fit with the physical lathe in terms of the number of components, geometric accuracy and the motion constraints. Secondly, the confidence of DT #5 is measured based on the simulation accuracy of the mechanism model and uncertainty of the data-driven model (namely surface quality prediction model). We compare the simulation results of the spindle speed and current of the mechanism model with that of the physical lathe when turning a 6061-aluminum alloy bar with size $\varphi 80$ mm $\times$ 200 mm, under the conditions of f= 100 mm/min, d= 1.5 mm, v= 1000 r/min. The results are shown in Fig. 11(a), where the average confidence of 97.18% is obtained for DT #5. Finally, *synchronization* of DT #5 is evaluated by measuring the difference between responses from the physical end and DT end. The result shows that the average delay between the physical end and DT end is less than 20 ms, which, therefore, indicates that $F_s = 1$ as the average delay of DT #5 accords with the expected delay level. Based on the above evaluation experiments, the comprehensive accuracy of DT #5 is 97.18%. Considering the uncertainty of DT modelling and performance degradation of the physical lathe, DT #5 is maintained and optimized in terms of *integrity*, *confidence* and *synchronization*, which guarantees the performance of DT #5 during its lifecycle.

The latency performance of the system is compared with that without MEC deployed. The latency of two systems is measured by the time difference between the input timestamp of a service request from device level and DTMT response timestamp at edge for solving that request. As shown in Fig. 12, the results show that the proposed prototype system could achieve an ultra-low average latency performance of 47.04 ms, which significantly outperforms the average latency performance of 352.47 ms obtained by the prototype system without MEC deployed. This may attribute to the MEC-based algorithm that provides optimal service caching and computation offloading capacities to handle the request more effectively.

7. Conclusion and future work

This paper proposes a novel MEC-enabled knowledge-sharing DTMT swarm to construct a smart and flexible machining process towards intelligent manufacturing in Industry 4.0. Based on the experimental results obtained in this paper, the following contributions are of significance.

(1) A novel framework for the construction of a knowledge-sharing DTMT swarm is proposed by integrating DT with MEC, which enjoys significant advances in real-time perception, high-fidelity simulation, high-confidence decision-making and control. The framework also provides a MEC-based security mechanism for knowledge sharing across the authorized machine tools in the swarm, while achieving an ultra-low latency performance.

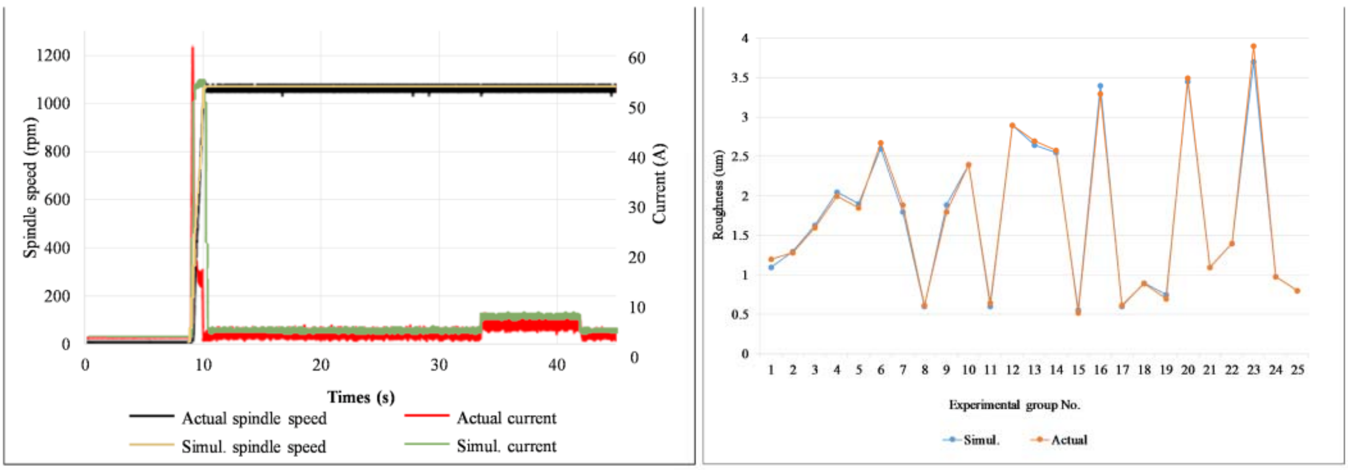
(2) Three key enabling methodologies of the framework are introduced from the perspective of DTMT swarm construction, knowledge-based cloud brain learning, and MEC-enhanced system deployment, which provide an insight into the industrial implementation of the MEC-enabled knowledge-sharing DTMT swarm.

(3) The implementation of a knowledge-sharing DTMT swarm prototype demonstrates the feasibility of the proposed approach, where its application examples show that the proposed approach could equip each of machine tools in the swarm with high-level intellectualization. In addition, evaluation experiments illustrate the superiority of the proposed approach compared with the traditional edge-cloud based system.

This paper explores the key aspects of the MEC-enabled knowledge-sharing DTMT swarm. Nevertheless, several challenges are remained to be addressed in the near future. Firstly, the security mechanism provided by MEC is difficult to cope with the increasing scale of the system, especially with cyber-threats on the rise and sophistication of attacks increasing. Therefore, we plan to integrate advanced concepts and technologies, such as zero-trust networking and blockchain, into the system to improve the overall security of MEC services. Secondly, human-centric smart manufacturing is one of the most important features in Industry 5.0 [40,41], where, however, human aspect is missed in the proposed framework. We plan to introduce the augmented reality technology into the proposed framework for the construction of a human-machine collaborative manufacturing environment to enhance the process planning and machining process. In addition, we also plan to combine the MEC-enabled knowledge-sharing mechanism with the predictive maintenance strategy [23] to provide smart guidance for human-oriented MRO services during the lifecycle of machine tools.

CRediT authorship contribution statement

Chao Zhang: Conceptualization, Methodology, Software, Investigation, Validation, Formal analysis, Data curation, Writing – original draft. **Guanghui Zhou:** Resources, Writing – review & editing, Supervision. **Jingjing Li:** Investigation, Writing – original draft. **Fengtian Chang:** Investigation, Writing – review & editing. **Kai Ding:** Writing – review & editing. **Dongxu Ma:** Writing – review & editing.



(a) Simulation accuracy of the mechanism model

(a) Prediction accuracy of the surface quality prediction model

Fig. 11. Confidence evaluation of DT #5.

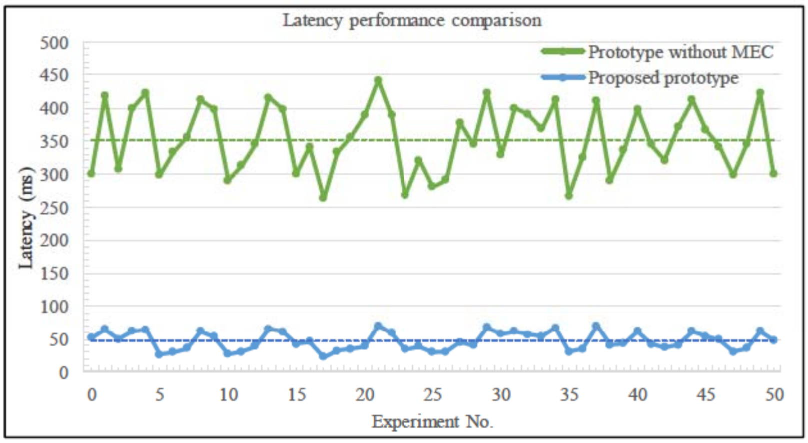


Fig. 12. Latency performance comparison experiments.

Declaration of Competing Interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgement

This work was supported by the National Key Research and Development Program of China [grant number 2021YFB3301400], the National Natural Science Foundation of China [grant number 52105530], and the China Postdoctoral Science Foundation [grant number 2021M692556].

References

[1] Deloitte, 2022 manufacturing industry outlook, ⟨https://www2.deloitte.com/us/en/pages/⟩ energy-and-resources/articles/manufacturing-industry-outlook.html; 2022 [accessed 24 October 2022].
[2] Gartner, Smart Factory Trends for Manufacturing Industries, ⟨https://www.gartner.com/en/supply-chain/trends/smart-factory-trends⟩; 2022 [accessed 24 October 2022].
[3] Mourtzis D. Simulation in the design and operation of manufacturing systems: state of the art and new trends. Int J Prod Res 2020;58:1927–49. https://doi.org/10.1080/00207543.2019.1636321.
[4] Zhang C, Zhou G, Li H, Cao Y. Manufacturing blockchain of things for the configuration of a data- and knowledge-driven digital twin manufacturing cell. IEEE Internet Things J 2020;7:11884–94. https://doi.org/10.1109/JIOT.2020.3005729.
[5] Lou P, Liu S, Hu J, Li R, Xiao Z, Yan J. Intelligent machine tool based on edge-cloud collaboration. IEEE Access 2020;8:139953–65. https://doi.org/10.1109/ACCESS.2020.3012829.
[6] Chen J, Hu P, Zhou H, Yang J, Xie J, Jiang Y, Gao Z, Zhang C. Toward intelligent machine tool. Engineering 2019;5:679–90. https://doi.org/10.1016/j.eng.2019.07.018.
[7] Leng J, Sha W, Lin Z, Jing J, Liu Q, Chen X. Blockchained smart contract pyramid-driven multi-agent autonomous process control for resilient individualised manufacturing towards Industry 5.0. Int J Prod Res 2022:1–20. https://doi.org/10.1080/00207543.2022.2089929.
[8] Zhou G, Zhang C, Li Z, Ding K, Wang C. Knowledge-driven digital twin manufacturing cell towards intelligent manufacturing. Int J Prod Res 2020;58:1034–51. https://doi.org/10.1080/00207543.2019.1607978.
[9] Glaessgen EH, Stargel DS. The digital twin paradigm for future NASA and U.S. Air force vehicles. Hawall: IAA/ASME/ASCE/AHS/ASC Structures, Structural Dynamics and Materials ConferenceHonolulu,; 2012. https://doi.org/10.2514/6.2012-1818.
[10] Leng J, Zhou M, Xiao Y, Zhang H, Liu Q, Shen W, Su Q, Li L. Digital twins-based remote semi-physical commissioning of flow-type smart manufacturing systems. J Clean Prod 2021;306:127278. https://doi.org/10.1016/j.jclepro.2021.127278.
[11] Ghosh AK, Ullah AMMS, Teti R, Kubo A. Developing sensor signal-based digital twins for intelligent machine tools. J Ind Inf Integr 2021;24:100242. https://doi.org/10.1016/j.jii.2021.100242.
[12] Pan L, Guo X, Luan Y, Wang H. Design and realization of cutting simulation function of digital twin system of CNC machine tool. Procedia Comput Sci 2021;183:261–6. https://doi.org/10.1016/j.procs.2021.02.057.
[13] Liu K, Song L, Han W, Cui YM, Wang YQ. Time-varying error prediction and compensation for movement axis of CNC machine tool based on digital twin. Ieee Trans Ind Inform 2022;18:109–18. https://doi.org/10.1109/tii.2021.3073649.
[14] Q.Z. Qiao, J.J. Wang, L.K. Ye, R.X. Gao, Digital Twin for Machining Tool Condition Prediction, 52nd CIRP Conference on Manufacturing Systems (CMS)Ljubljana, SLOVENIA, 2019, pp. 1388–1393. doi: 10.1016/j.procir.2019.04.049.
[15] Stavropoulos P, Mourtzis D. Chapter 10 - Digital twins in industry 4.0. In: Mourtzis D, editor. Design and Operation of Production Networks for Mass Personalization in the Era of Cloud Technology. Elsevier; 2022. p. 277–316.
[16] Zhang C, Zhou G, Jing Y, Wang R, Chang F. A digital twin-based automatic programming method for adaptive control of manufacturing cells. IEEE Access, 10; 2022. p. 80784–93. https://doi.org/10.1109/ACCESS.2022.3195905.
[17] Zhao G, Cao X, Xiao W, Zhu Y, Cheng K. Digital Twin for NC Machining Using Complete Process Information Expressed by STEP-NC Standard. Proceedings of the 2019 4th International Conference on Automation, Control and Robotics Engineering. Shenzhen, China: Association for Computing Machinery,; 2019. https://doi.org/10.1145/3351917.3351979.
[18] Tong X, Liu Q, Pi SW, Xiao Y. Real-time machining data application and service based on IMT digital twin. J Intell Manuf 2020;31:1113–32. https://doi.org/10.1007/s10845-019-01500-0.
[19] Z.Y. Zhao, S.B. Wang, Z.H. Wang, S.L. Wang, C. Ma, B. Yang, Surface roughness stabilization method based on digital twin-driven machining parameters self-adaption adjustment: a case study in five-axis machining, Journal of Intelligent Manufacturing. doi: 10.1007/s10845–020-01698–4.
[20] Zhou Y, Xing T, Song Y, Li YJ, Zhu XF, Li G, Ding ST. Digital-twin-driven geometric optimization of centrifugal impeller with free-form blades for five-axis flank milling. J Manuf Syst 2021;58:22–35. https://doi.org/10.1016/j.jmsy.2020.06.019.
[21] Luo WC, Hu TL, Ye YX, Zhang CR, Wei YL. A hybrid predictive maintenance approach for CNC machine tool driven by Digital Twin. Robot Comput-Integr Manuf 2020;65. https://doi.org/10.1016/j.rcim.2020.101974.
[22] Wei YL, Hu TL, Zhou TT, Ye YX, Luo WC. Consistency retention method for CNC machine tool digital twin model. J Manuf Syst 2021;58:313–22. https://doi.org/10.1016/j.jmsy.2020.06.002.
[23] Mourtzis D, Angelopoulos J, Panopoulos N. Design and development of an edge-computing platform towards 5G technology adoption for improving equipment predictive maintenance. Procedia Comput Sci 2022;200:611–9. https://doi.org/10.1016/j.procs.2022.01.259.
[24] Nikravan M, Haghi Kashani M. A review on trust management in fog/edge computing: Techniques, trends, and challenges. J Netw Comput Appl 2022;204:103402. https://doi.org/10.1016/j.jnca.2022.103402.
[25] Leng J, Chen Z, Sha W, Ye S, Liu Q, Chen X. Cloud-edge orchestration-based bi-level autonomous process control for mass individualization of rapid printed circuit boards prototyping services. J Manuf Syst 2022;63:143–61. https://doi.org/10.1016/j.jmsy.2022.03.008.
[26] ESTI, Multi-access Edge Computing (MEC) Framework and Reference Architecture, ⟨https://www.etsi.org/deliver/etsi_gs/MEC/001_099/003/02.02.01_60/gs_MEC003v020201p.pdf⟩, 2020 [accessed 31 December 2020].
[27] ESTI, Multi-access Edge Computing (MEC); Phase 2: Use Cases and Requirements, ⟨https://www.etsi.org/deliver/etsi_gs/MEC/001_099/002/02.02.01_60/gs_MEC002v020201p.pdf⟩, 2022 [accessed 31 January 2022].
[28] Liang B, Gregory MA, Li S. Multi-access edge computing fundamentals, services, enablers and challenges: a complete survey. J Netw Comput Appl 2021:103308. https://doi.org/10.1016/j.jnca.2021.103308.
[29] Song M, Lee Y, Kim K. Reward-oriented task offloading under limited edge server power for multiaccess edge computing. IEEE Internet Things J 2021;8:13425–38. https://doi.org/10.1109/JIOT.2021.3065429.
[30] Liu P, An K, Lei J, Zheng G, Sun Y, Liu W. SCMA-based multi-access edge computing in IoT systems: an energy-efficiency and latency tradeoff. IEEE Internet Things J 2021:1-1. https://doi.org/10.1109/JIOT.2021.3105658.

[31] Ali B, Gregory MA, Li S. Multi-access edge computing architecture, data security and privacy: a review. IEEE Access 2021;9:18706–21. https://doi.org/10.1109/ACCESS.2021.3053233.
[32] Zhang C, Zhou G, Li J, Qin T, Ding K, Chang F. KAiPP: An interaction recommendation approach for knowledge aided intelligent process planning with reinforcement learning. Knowl-Based Syst 2022:110009. https://doi.org/10.1016/j.knosys.2022.110009.
[33] Liu C, Xu X, Peng Q, Zhou Z. MTConnect-based Cyber-Physical Machine Tool: a case study. Procedia CIRP 2018;72:492–7. https://doi.org/10.1016/j.procir.2018.03.059.
[34] Li J, Zhou G, Zhang C. A twin data and knowledge-driven intelligent process planning framework of aviation parts. Int J Prod Res 2022;60:5217–34. https://doi.org/10.1080/00207543.2021.1951869.
[35] Wagner KH, McComb S. Optical rectifying linear units for back-propagation learning in a deep holographic convolutional neural network. IEEE J Sel Top Quantum Electron 2020;26:1–18. https://doi.org/10.1109/JSTQE.2019.2946655.
[36] Singla SK, Yadav RK. Optical character recognition based speech synthesis system using LabVIEW. J Appl Res Technol 2014;12:919–26. https://doi.org/10.1016/S1665-6423(14)70598-X.
[37] Tsai C-Y, Lai B-H. A Location-Item-Time sequential pattern mining algorithm for route recommendation. Knowl-Based Syst 2015;73:97–110. https://doi.org/10.1016/j.knosys.2014.09.012.
[38] Shabtay L, Fournier-Viger P, Yaari R, Dattner I. A guided FP-Growth algorithm for mining multitude-targeted item-sets and class association rules in imbalanced data. Inf Sci 2021;553:353–75. https://doi.org/10.1016/j.ins.2020.10.020.
[39] Zhang C, Zhou G, Hu J, Li J. Deep learning-enabled intelligent process planning for digital twin manufacturing cell. Knowl-Based Syst 2020;191:105247. https://doi.org/10.1016/j.knosys.2019.105247.
[40] Huang S, Wang B, Li X, Zheng P, Mourtzis D, Wang L. Industry 5.0 and Society 5.0—Comparison, complementation and co-evolution. J Manuf Syst 2022;64:424–8. https://doi.org/10.1016/j.jmsy.2022.07.010.
[41] Leng J, Sha W, Wang B, Zheng P, Zhuang C, Liu Q, Wuest T, Mourtzis D, Wang L. Industry 5.0: prospect and retrospect. J Manuf Syst 2022;65:279–95. https://doi.org/10.1016/j.jmsy.2022.09.017.