Journal of Manufacturing Systems 58 (2021) 205–222

Contents lists available at ScienceDirect

Journal of Manufacturing Systems

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

Event-driven tool condition monitoring methodology considering tool life prediction based on industrial internet

Yahui Wang, Lianyu Zheng, Yiwei Wang *

School of Mechanical Engineering and Automation, Beihang University, Beijing 100191, China

A R T I C L E I N F O

Keywords:
Event-driven
Tool condition monitoring
Industrial internet
OPC-UA
Tool life prediction

A B S T R A C T

Tool condition monitoring (TCM) and remaining useful life (RUL) prediction is of great practical significance for any machining process to ensure machining quality and reduce the machine tool downtime. At the standpoint of workshop management, the current TCM has two drawbacks. (i) Continuously acquiring data without distinguishing the working states of the machine tool and the machining tasks will inevitably bring a large volume of unwanted signals, making difficulty for tool RUL prediction. (ii) The tool condition is independent of machining task, thus cannot provide further decision-making support for workshop scheduling and machining parameters optimization. Therefore, it is an important issue to consider various random events under the right machining tasks to trigger "monitoring" and RUL prediction just in time. This paper proposes an event-driven tool condition monitoring (EDTCM) methodology. The structure of EDTCM is designed based on the architecture of the Industrial Internet. Multi-source events are collected under the architecture, including MES events, machine tool events based on the OPC-UA (OPC Unified Architecture) standard, smart mobile terminal events, etc. The event-driven mode is designed to process these events such that the "monitoring" is triggered just in time. Then the Tool RUL is predicted online with the monitored sensor data based on the Bayesian method. A prototype system of EDTCM is developed and a case study is implemented to verify the feasibility of the proposed methodology. Our work promotes that the theories of TCM and tool RUL prediction deeply integrate with the real industrial practical applications.

1. Introduction

As an important part of the machining process that directly contacts the workpiece, the state of the cutting tool greatly affects the product quality and production cost. Being subjected to continuous wear, the states of the cutting tool degrades, leading to a decrease of machining precision and even sudden downtime of machine tools if left unattended [1,2]. This is especially significant for the high-value key components such as steam turbines and aero engines, whose surface quality need to be strictly controlled since unsatisfied machining precision will cause great loss. On the other hand, appropriate planning of the tool according to the tool condition can extend the unsupervised running time of machine tool and reduce the production cost [3]. In contrast, changing cutting tools in advance without fully utilized brings unnecessary cost. According to the U.S. Cutting Tool Institute, manufacturers in the U.S. consumed over $200 million worth of cutting tools during April 2018 [4,5]. Therefore, monitoring the condition and predicting the remaining useful life (RUL) of the cutting tools such that one can change the tools in time just before failure is practical significant for reducing cost and increasing efficiency while ensuring the product quality.

The tools are typically monitored in a direct or indirect way. Generally, the direct way requires offline interrupt machining to measure the tool condition, and the indirect way monitors the tool condition through online sensing [6]. Due to the limited space accessibility and the infeasibility of frequent uninstall of tools, direct monitoring is less practical in real cases. In contrast, indirect monitoring acquires signals (such as cutting force, vibration, current, etc.) and assesses the tool condition by analyzing and modeling the signals without preventing the normal operation of the machine. Therefore, indirect monitoring is more widely used in real cases. In practice, the preparation time accounts for a majority of the running time of the machine tool and the real cutting time only accounts for a small part. Triggering the "monitoring" event at the right time such that only the "effective signals" during the cutting time are acquired is a practical issue encountered by a real manufacturing system. This issue is important for the following reasons. The running data of the workshop (such as machine tool data, sensor

* Corresponding author.
E-mail address: wangyiwei@buaa.edu.cn (Y. Wang).

https://doi.org/10.1016/j.jmsy.2020.11.019
Received 28 June 2020; Received in revised form 23 November 2020; Accepted 23 November 2020
Available online 15 December 2020
0278-6125/

data, etc.) are continuous. These data are large in volume and small in variation. Most of these data are useless for state change of the machining task and the tool RUL prediction. Continuously acquiring data without distinguishing the working states of the machine tool (e.g., whether the tool is cutting) will inevitably bring unwanted signals. Due to the high sampling rate of sensors, the data volume is normally large. A large volume of unwanted signals scattered in a small volume of useful cutting signals will bring difficulties to signal extraction and modeling, and also bring pressure for data storage. When the number of equipment is few, it may not have much impact on the system's computation tasks. However, when the number of equipment in the workshop increases greatly, it will bring a serious burden to the computation and storage of the system.

Furthermore, the machining task of the workpiece is closely related to the tool state. Therefore, only when the workpiece starts to be machined under the right machining task, the "monitoring" event is triggered to drive the monitoring of the tool condition data, and the RUL prediction. At the standpoint of workshop management, the motivation is to bind the tool conditions with machining tasks, and which can further provide decision making support such as scheduling and machining parameters optimization. This is seldom considered in the existing research. For examples, if we predict that the tool will fail within a certain number of cutting times, predictive maintenance will be scheduled in advance such that the downtime of the machine tool can be reduced. In addition, the different machining parameters of the workpiece will affect the tool life. Therefore, TCM and RUL prediction under different machining tasks (different machining parameters) has two aspects of influence. On the one hand, it can provide a tool life reference for the process planning of the next batch of workpieces. On the other hand, when severe tool wear occurs during machining, the monitoring results can provide data for changing machining parameters. The changed machining parameters enable the tool to complete the machining task within a limited life, thereby avoiding the reduction of machining quality caused by tool change during machining.

Based on the above analysis, triggering the "monitoring" event is very important for just-in-time tool condition monitoring (TCM). But triggering the "monitoring" event requires an accurate judgment of the state of machine tools and workpieces. In the era of Industry 4.0, within the architecture of Industrial Internet, the device networking enables to access the data from multiple manufacturing resources (e.g., machine tools, tools, workpieces, workers, sensors, etc.), which provide a solution for the accurate judgment of the state of the machine tools and the workpieces. Industrial Internet integrates physical and cyber components in manufacturing systems, making resources, data, and knowledge in manufacturing systems ubiquitous [7,8], based on which, Industrial Internet gains insight into industrial processes from data to improve productivity, efficiency, and reliability [9]. The interconnection and communications among multiple manufacturing resources generate multi-source events (such as "tool starts cut") in the manufacturing system [10,11]. The event-driven approach can effectively improve the responsiveness and flexibility of equipment monitoring and production scheduling in the manufacturing system [12,13]. Therefore, effectively processing these discrete events provide a solution for the issue of "just-in-time" TCM and predicting. Industrial Internet provides an effective mode to manage the multiple resources as well as to process the discrete events generated by the resources that are involved in the TCM.

Motivated by addressing the issue of just-in-time TCM and online tool RUL prediction in smart factories, this paper proposes an event-driven tool condition monitoring (EDTCM) methodology based on the Industrial Internet architecture. In the methodology, the events related to machining tasks and machine tools are monitored. The event processing technology is used to process multi-source events to generate events that trigger the TCM and then drive the RUL prediction.

The remainder of the paper is organized as follows. Related works are reviewed in Section 2. The architecture and the process of EDTCM and the data conversion of private protocol based on OPC-UA are described in Section 3. The event processing in TCM is introduced in Section 4. The event-driven tool RUL prediction flow and tool RUL prediction method are detailed in Section 5. A prototype system based on EDTCM and the discussion and analysis of the results is presented in Section 6. Finally, the conclusion and future work are given in Section 7.

2. Related works

The related works of the following two aspects involved in the EDTCM are reviewed, (1) TCM and RUL prediction, and (2) event-driven manufacturing systems.

2.1. TCM and RUL prediction

2.1.1. TCM

The main aims of TCM are to assess the tool conditions and predict the remaining useful life based on various sensor signals (acoustic emission, milling force, tool/workpiece vibration, sound, spindle torque, spindle motor current, et al.) [14,15]. Zhou et al. introduced a TCM approach that utilized the SVM model and a transition point identification method (TPIM) to reach more accurate classification results [16]. Paul et al. proposed a cyber-physical system for TCM using electrical power and a mechanistic model to improve cutting parameters [17]. Zhang et al. proposed a TCM and life estimation system based on multiple sensors such as vibration, cutting force, and power data, as well as actual machining parameters [18]. Dai et al. proposed a machine vision system for the micro-milling TCM method, which detects and captures focused image analysis and predicts tool wear states within a predetermined machining interval [19]. Ratava et al. used acceleration sensors to estimate the deflection of the tool and associated it with the main cutting force, and then used this information to detect chip marks and small cracks at the edge of the tool [20].

From the above study, firstly, it can be seen that currently TCM mainly focuses on the monitoring solutions (e.g., the type of sensor signals, the methods of assessing tool status) without considering how to integrate it into industrial practical applications. Secondly, in the current studies, the random events in the production workshop are simplified or ignored in the TCM algorithm. How to integrate TCM into workshop machining tasks, and taking into account the multi-source data (such as machining data, production planning data, sensor data, etc.) to achieve just-in-time TCM based on event-driven technology has not been studied.

2.1.2. RUL prediction

Tool remaining useful life (RUL) prediction is an important aspect of TCM. In recent years, deep learning-based tool RUL prediction methods attract lots of attentions [21–25]. Deep learning-based methods rely on a large number of run-to-fail data to train an off-line prediction model, and the trained model is sensitive to changes such as the changes in working conditions, new prediction tasks, etc. In practice, it is difficult to obtain run-to-fail data, and it is infeasibility to train prediction models covering all working conditions. In contrast, instead of using run-to-fail data to train an off-line model, the degradation model-based methods update the model parameters with the sequential arrived monitoring data and thus can predict RUL online and give increasingly accurate RUL over time with more data available [26–29]. Jaydeep et al. proposed an extended Taylor tool life equation to predict the tool life. The Bayesian inference is applied to estimate the parameters of the extended Taylor tool life equation using the Metropolis-Hastings algorithm of the Markov Chain Monte Carlo (MCMC) approach and using a discrete grid method [30,31]. For the problem of parameter updating of the degradation model in the prediction of RUL, compared with particle filtering, Kalman filtering, and other methods, the Bayesian inference has lower sensitivity to the initial value of parameters and noise of observed data, better robustness, and better prediction accuracy in most cases [32].

It should be pointed out that the tool usage process of the manufacturing system contains control strategies or control rules. Much preparation work such as jig adjustment, parameter determination of CNC program need to be done, and the time span is long. The advantage of the previous analysis algorithm model lies in its mathematical theory. But most analysis algorithm models do not consider the relevant events in the TCM process in the manufacturing system, only giving the prediction model and the results. In practice, the real cutting time of the tool only accounts for a small part of the entire machining task time. If the whole machining data is monitored, the real cutting data will be submerged in a large volume of useless signals. Therefore, it is essential to drive the TCM process through events.

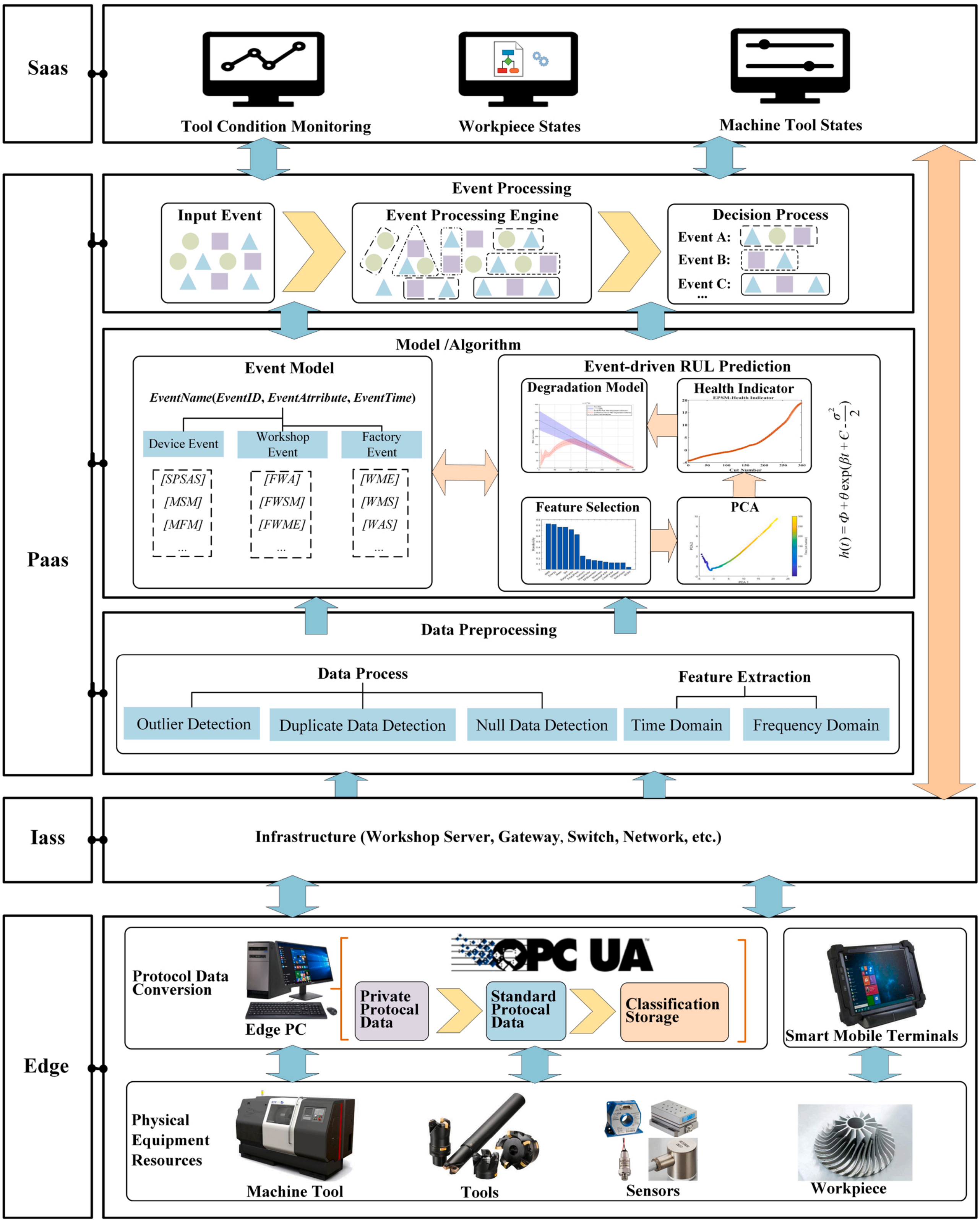

Fig. 1. Architecture of EDTCM based on Industrial Internet.

2.2. Event-driven manufacturing system

2.2.1. Concept and method of the event-driven manufacturing system

The discrete manufacturing system (DMS) is a complex, random, and dynamic system. Large numbers of discrete events exist in DMS, making the operation, control, and optimization strategies of DMS very complex. From energy saving aspects in DMS, Wang et al. proposed an event-driven estimation method of an energy-saving window based on Max-plus Algebra is presented to put the target machine to sleep, considering real-time production data of a system segment [33]. In manufacturing process control, Mourtzis and Wang proposed the use of event driven function blocks as a container of algorithms that control the manufacturing process and could be generating non-linear process plans [34,35], and Wang realized the distributed control through the use of networked and smart FB decision modules [36]. In terms of production planning and logistics control, some scholars proposed an RFID event-driven integrated production planning and control (RED-IPPC) framework and event-driven feedback control method for a better dynamic schedule [37,38]. Cao et al. proposed an event-driven data collecting units (EDCUs) for monitoring the production process of the part in job-shop floor [39]. Wang et al. used useful production events to realize the efficient implementation of production control systems, and proposed a real-time production control system for intelligent job shop resource logistics based on graphic reasoning of production events [40]. Berger et al. proposed an event-based approach to meet logistics requirements of a changing market environment [41]. In summary, in the manufacturing industry with increasingly higher informatization, event-driven manufacturing systems are more accordant with the discrete manufacturing industry, which can bring more effective execution methods to the system and improve the system's responsiveness.

2.2.2. Event-driven practical applications

Theorems et al. proposed an event-driven manufacturing information system architecture-Line Information System Architecture (LISA), which aimed to achieve flexible factory integration and data utilization [42]. In order to make better use of the real-time data at the shop floor and to facilitate the decision-making process, Umer et al. implemented an event-driven assembly service process in the assembly workshop [43]. Bousdekis et al. proposed an event-driven decision-making method based on production and maintenance, developed an associated information system contribution decision-making method and technical specifications [10]. It can eliminate maintenance-related losses and optimize business performance. The application of event-driven manufacturing systems make better use of the underlying data in manufacturing enterprises and provide effective solutions for enterprise system integration.

To sum up, on the one hand, the real-time processing of data from multiple sources in the manufacturing industry is an inevitable requirement for future smart workshops and smart factories. The event-driven manufacturing system architecture can more realistically implement discrete event processing for manufacturing systems and has been proven to be an effective control driving method that can improve system response-ability and save resources. On the other hand, event-driven monitoring technology is an important technical approach for the fusion processing of multi-source data in the manufacturing workshop, and it is a key problem that yet to be adequately resolved in the current discrete manufacturing workshop. In the above review, event-driven research focuses on manufacturing process control, parameter control, etc., in the production process, and event-driven research on tool RUL monitoring and prediction is relatively few. Therefore, event-driven tool RUL monitoring and prediction are for worth further research.

2.3. Summary and analysis

From the literature reviews, the current research of TCM and RUL prediction mainly focuses on improving the accuracy of TCM through different sensor data and prediction models. However, in the actual industrial situation, the cutting time of the tool only accounts for a small part of the whole machining task. Triggering the "monitoring" event at the right time (when workpiece starts to be machined under the right machining task) such that only the "effective signals" during the cutting time are acquired is a practical issue encountered by a real manufacturing system. Therefore, the scientific problem is how to combine the various events to support decision-making such that one can accurately judge the machining states (such as the workpiece starts being machined) and trigger the "monitoring" event at the right time. The challenge is that at the standpoint of workshop management, triggering the tool condition monitoring just at the right time is not a simple thing and thus only taking into account the data from machine tools controllers is not enough. Therefore, it is necessary to combine machining tasks with tool condition monitoring through the events generated by multiple manufacturing resources in the manufacturing system. However, the events generated by multiple manufacturing resources have multi-source and heterogeneous problems. Therefore, it is essential to utilize a unified platform to achieve multi-source event collection and use event processing technology to complete multi-source event processing to trigger tool status monitoring at the right time. Motivated by deploying the TCM and tool RUL prediction in the real manufacturing system, in this paper, an EDTCM methodology based on Industrial Internet architecture is proposed. The contributions are summarized as follows. Various events in the workshop are utilized to drive the TCM just at the right time and predict the remaining useful life of tools to further provide decision-making support for workshop management. This issue is seldom considered in the existing research work. We propose a machining task states recognition solution that considers multi-source events generated in the workshop production process and drives the TCM based on "monitoring" events. Then, the tool RUL is predicted based on the Bayesian update driven by the "monitoring" event. Furthermore, the architecture of EDTCM based on the concept of Industrial Internet is designed, which integrates the functions of data acquisition, data modeling, event processing, and RUL prediction. Finally, a prototype system is developed and deployed in the workshop production site.

3. Architecture and process of EDTCM

This section first presents the architecture of the EDTCM based on Industrial Internet, then introduces the data conversion of private protocol based on OPC-UA followed by presenting the process model that describes the data flow.

3.1. Architecture of EDTCM based on industrial internet

Fig. 1 shows the architecture of EDTCM based on Industrial Internet, which consists of four layers, i.e., the Edge layer, the Iass layer, the Pass layer, and the Saas layer.

The Edge layer includes physical equipment resources and protocol data conversion based on OPC-UA and smart mobile terminals. Physical equipment resources include machine tools, tools, sensors, workpiece, etc. OPC-UA protocol is used for data standardization of machine tools and tools in the workshop. The OPC-UA server is installed on a computer near each machine tool in the workshop to collect the machine tool private protocol data and convert it to OPC-UA protocol. Then each computer is connected to the workshop server through the switch. The smart mobile terminals are used to assist workers to scan the two-dimensional code of the workpiece and report the machining quality states of the workpiece, etc. The reason for protocol conversion at the Edge layer is to improve the efficiency of data processing. In case that

Table 1
Multi-source data.

Data type	Data Item
Machine Tool Data (OPC-UA)	Running program number, spindle (speed, load, etc.), machine switch signals, program running states, running time, tool number, etc.
Workpiece Data	Workpiece machining states, workpiece machining quality, workpiece number, workpiece position, etc.
Production Planning Data	The number of the workpiece at the station, the program number of the workpiece, the process number of the workpiece, tool number, etc.
Sensor data	Cutting force, acceleration, acoustic emission, etc.

the workshop equipment increases greatly and all the data of the machine tool are collected and converted in the Pass layer, the data processing efficiency will be reduced. For sensor data and workpiece data, the private protocol can be used to monitor the running states of key components (such as tools, screws, spindles, etc.) and the machining task states of workpieces in real time.

The Iass layer includes the infrastructure needed for physical equipment networking in the workshop. These infrastructures ensure efficient data transmission and storage. For example, the workshop server, gateway, switch, etc. that are used for collecting data from the machine tool, MES system, and the sensor data of the tool.

Pass layer consists of three modules, i.e., data preprocessing module, model/algorithm module, and event processing module. The data preprocessing module collects multi-source data for basic preprocessing such as outlier detection, duplicate value detection, null value detection, time-frequency domain feature extraction of sensor data, etc. Table 1 reports the multi-source data studied in this paper, including machine tool data, workpiece data, production planning data, and sensor data. The model/algorithm module builds models and algorithms using the data for the aims of processing events and predicting tool RUL. The function of the event processing module is to process events obtained through the layers and generate corresponding decision events to provide decision support for upper-layer applications.

The Saas layer forms applications based on the low-level event and the data analysis results. This paper focuses on the application of TCM and RUL prediction. Besides, this layer also includes the applications for monitoring the real-time status of machine tools and workpiece in the manufacturing workshop.

Four types of multi-source data involved in this paper are reported in Table 1. The machine tool data and the workpiece data are processed by edge devices such as computers, wearable devices, smart terminal devices, high-performance routers, etc. [50–53]. Specifically, for the machine tool data, the OPC-UA server installed in the edge computer near the machine tool discard a large number of repeated data and remove repetitive events related to the machine tool. For the workpiece data, the attribute information of the workpiece and the information of the final machining quality are processed by the smart mobile terminal on the edge side. The position and machining states of the workpiece with high confidence are feedback to the TCM system. In contrast, the production plan data and the sensor data are not directly processed by edge devices since the production plan data can be obtained directly from the MES system without involving edge devices. For sensor data, the idea of triggering TCM based on "monitoring" events only needs to collect effective data under the cutting state to facilitate the establishment of prediction model.

Generally, direct modeling and prediction using raw data on the edge side can save computing resources in the cloud and improve the system's responsibility. In this paper, we consider having a prediction model with the TCM system for the following reasons. (1) As the tools to be monitored in the workshop are various (e.g., different in types such as ball nose cutter, dick-type cutter, etc.), thus different prediction models are needed to bind with different types of tools. The prediction model is typically complex, and human-computer interaction is often needed to adjust, select, designate parameters, or modify the model. In addition, the raw data collected from the device side may imperfect, e.g., missing or outlier values. Therefore, before building the predictions model, it requires algorithm developers to perform customized processing (such as interpolation, outlier elimination, call the third-party software for further time-frequency analysis, multi-language mixed programming, etc.) on the basis of the raw data [54]. Due to the complicated process of customized processing and

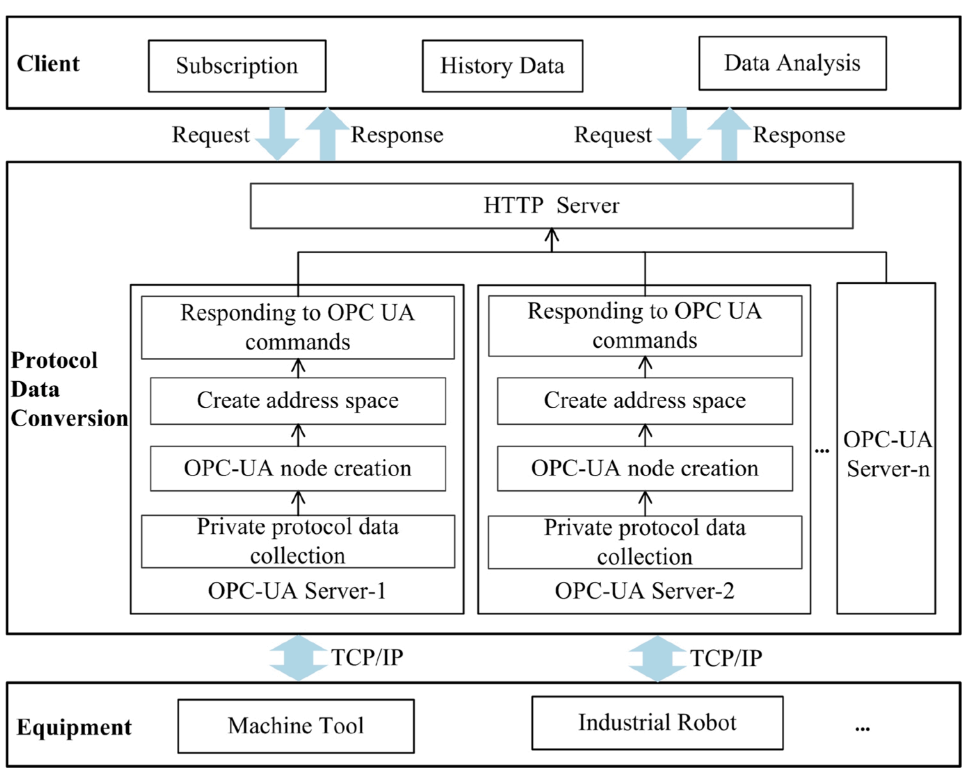

Fig. 2. Data conversion of equipment private protocol based on OPC-UA.

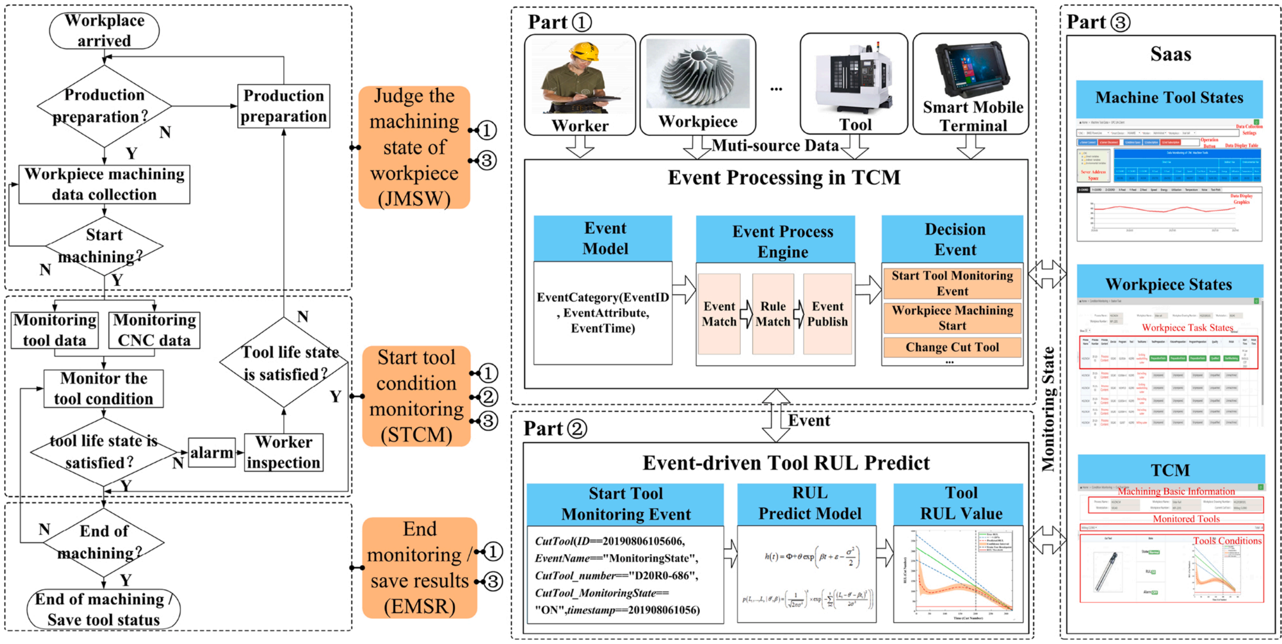

Fig. 3. Process of EDTCM.

modeling of the raw data, it is necessary to upload raw data to the Paas layer. It is more convenient for algorithm developers to build, modify, and update prediction models through the Paas layer. (2) Furthermore, in the Paas layer, high-performance server computing resources can be conveniently used, which is conducive for the algorithm developers to further analyze the data to build and update prediction models. (3) Modeling and predicting at the Pass layer has good scalability. In our future work, the monitoring of different equipment such as industrial robots, AGVs, components of machine tools, etc. will be integrated into the workshop. The prediction models of these heterogeneous equipment can be uniformly managed and updated. For example, when the monitoring task is extended to the machine tool spindle, lead screw and other equipment components in the workshop, the unified management of the model can be carried out according to different monitoring objects.

3.2. Data conversion of equipment private protocol based on OPC-UA

The motivation of this paper is that from the standpoint of workshop management, combining the various events in the workshop to drive the tool condition monitoring just at the right time, and predicting the remaining useful life of tools to further provide decision-making support for workshop management, such as scheduling and machining parameter optimization. The management of various equipment in the workshop such as machine tools, industrial robots, AGVs, etc., and the control of machining parameters need to be considered. The OPC-UA [55], MTConnect [56], and UMATI (Universal Machine Tool Interface) [57] can be considered to unify private protocol data of the workshop machine tool. Both MTConnect and UMATI have cooperated with OPC Foundation to develop joint companion specifications [58,59]. However, MTConnect and UMATI are only for machine tools. In addition, MTConnect does not support write and has no data security solutions. As mentioned above this paper is part of the research scope of the intelligent management and control of the workshop, and the equipment to be considered includes CNCs, industrial robots, AGVs, fixtures, and other equipment. Therefore, we consider not only the condition monitoring of various heterogeneous equipment, but also the control demand of industrial robots, AGVs, fixtures, etc. Therefore, we choose the OPC-UA for the following reasons (1) OPC-UA supports both reading and writing and has its own security policy. (2) OPC-UA facilitates the expansion of heterogeneous equipment such as industrial robots, AGVs, and fixtures, whose data will be extracted to support the workshop management in our future work.

Fig. 2 shows the OPC-UA-based private protocol data conversion method. Firstly, the OPC-UA Server connects the corresponding numerical control equipment of the workshop equipment layer through the TCP/IP protocol and collects equipment data through the private protocol. Secondly, multiple OPC-UA servers can be included in the protocol data conversion layer, where each OPC-UA server corresponds to a unique equipment. The protocol conversion process includes the following steps. (1) OPC-UA server creates the corresponding numerical control equipment's data model node. (2) OPC-UA server creates an address space for each node based on the OPC-UA data model. (3) OPC-UA server caches the collected private protocol data to the corresponding address space. (4) The client obtains the data in the address space through the response of OPC-UA server to the corresponding function commands and the HTTP server.

3.3. Process of EDTCM

Fig. 3 shows the process of EDTCM. The left subplot is the flowchart of the mechanism of EDTCM from the workpiece arrival to the end of the machining. The mechanism contains three stages, i.e., judging the machining states of the workpiece (JMSW), start tool condition monitoring (STCM), and end monitoring and save results (EMSR). On the right subplot, part one and two are the functions involved in the different stages of the process of EDTCM and part three shows the corresponding applications. The parts in the right subplot are attached to their corresponding stages in the left subplot.

When the workpiece arrives at the station, it enters the stage of JMSW. After the production preparation is completed, the workpiece machining data are analyzed to determine whether the workpiece starts being machined. The processing results in the JMSW stage are implemented by the data collected from multiple manufacturing resources and the event processing in TCM in part one. When the workpiece machining starts, it enters the STCM stage. RUL prediction and alarm in this stage are implemented through parts one and two. Finally, in the EMSR stage, part one is utilized to analyze the current machining states of the workpiece. When the workpiece machining finishes, the states of the tool is saved to provide data for the next machining step.

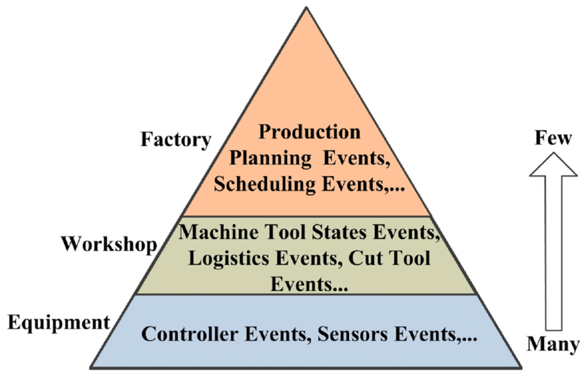

Fig. 4. Abstract levels of manufacturing enterprise events.

4. Event processing in TCM

In the process of EDTCM, events are generated by multiple manufacturing resources and event processing. Therefore, the definition of events, the event processing mechanism, and the TCM-related event processing are detailed as following.

4.1. Definition of events

An event is the object of an activity record in the system, and the event marks the activity [44]. In the workshop, events are generated by multiple manufacturing resources such as CNC machine tools, smart mobile terminals, etc. Some examples of the events include the start and stop of machine tools, and feedback events from workers in the smart mobile terminal, etc. The information model of an event is defined as:

$$EventCategory\ (EventID,\ EventAttribute,\ EventTime) \qquad (1)$$

EventCategory represents the active object that generates the event. For example, the events generated by machine tools such as the start and stop of machine tools are defined as "*MachineTool*". The events generated by the cut tool such as start and stop monitoring of cut tools, are defined as "*CutTool*". *EventID* is the unique identification number of the event. *EventAttribute* includes the attribute name (*AttributeName*) and an attribute value (*AttributeValue*) of an event. *EventTime* represents the time when the event occurred. Expression (2) gives an example of an event generated by the machine tool based on the OPC-UA node data. Based on the survey of workers' practical experience, we summarized the event of the machine tool starting machining. When the state of the machine tool, coolant, spindle load, NC-program, and the NC-program running time in the event attributes of the machine tool meet certain conditions, the event of machine tool starting machining is triggered.

$$MachineTool(ID==20190806105006, EventName=="\text{Machine tool start machining}", Cnc\_num=="G3", Cnc\_swtich=="on", Cnc\_coolant=="on", Cnc\_spdl!="0", ProgStatus=="\text{in progress}", AcTime>=180, timestamp==201908061050) \qquad (2)$$

Simple events can form a complex event, which is in a higher-level and normally provide decision support. Fig. 4 shows the event level of a manufacturing enterprise, which is divided into the equipment layer, workshop layer, and factory layer. It can be observed that the higher the level of events, the fewer the number of events.

4.2. Event processing mechanism based on rule engine

As shown in Fig. 5, the rule engine includes three parts, i.e., rule base, multi-source data, and event processing engine. The event processing engine accepts the events that are provided by the multi-source data and then processes these events by executing the corresponding rules in the rule base. The execution results provide support for the upper applications. The rule base is textual representations of expert knowledge, which is typically the fieldwork experiences of experts and the production rules accumulated during the long term manufacturing process. The rule engine combines the on-site multi-source data and expert domain knowledge to generate new events through rule matching

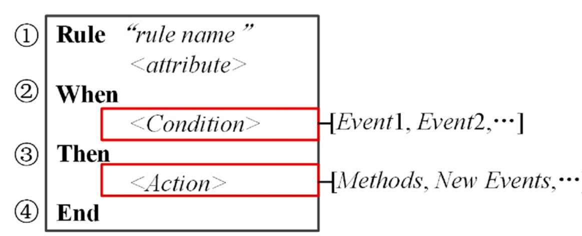

Fig. 6. Rule structure template.

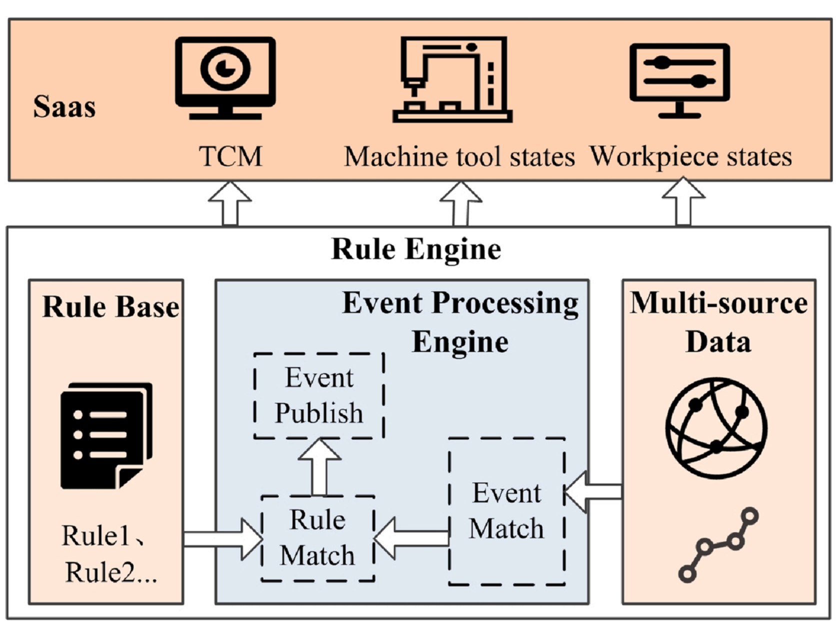

Fig. 5. Event processing mechanism based on Rule engine.

Table 2
Non-time series and time series operators between events.

	Operator	Expression	Explanation
Non-time series operators	AND(^)	$e=e_1\hat{}e_2$	When both events e_1 and e_2 occur, complex event e occurs.
	OR(ˇ)	$e = e_1\check{}e_2$	Only one of e_1 and e_2 occurred, and the complex event e occurred.
	NOT(¬)	$\neg e$	Event e did not happen.
Time series operators	SEQ(;)	$[e_1;e_2]$	e_1 has already happened when e_2 happened.
	TSEQ(;)	$[e_1;e_2]_{[t_1,t_2]}$	e_1 has already occurred when e_2 occurs, and the time distance between the two events is limited to $[t_1,t_2]$.
	SEQ(;$^+$)	$[e^+]$	One or more events of type e occur.
	TSEQ (;$^+$)	$[e^+]_{[t_1,t_2]}$	One or more events of type e occur and the time distance of the event is within $[t_1,t_2]$.

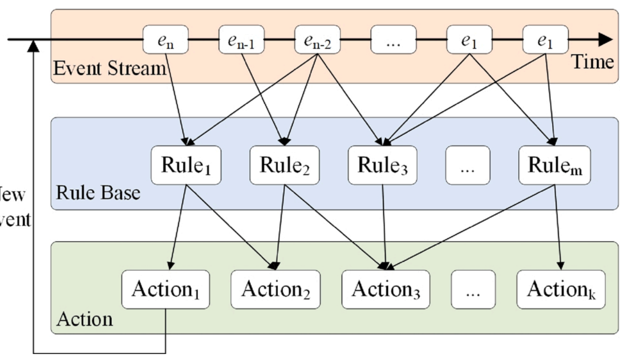

Fig. 7. Event processing based on stream.

and provides input for the applications in the Saas layer such as TCM, workpiece states, machine tool states.

Fig. 6 is a rule structure template. The *rulename* represents the meaning of the activity expressed by the rule. <*attribute*> is used for the rule matching and execution strategy (such as, how to set the priority of the rule execution). When the events in the <*Condition*> part satisfy the association rules (which will be introduced later), the methods, new events, and other actions customized by experts in the <*Action*> part will be triggered.

Events are combined through association rules to generate complex events. The association rules include time, causality, and set relationships [44].

(1) Time association events are correlated based on time. For example, when the occurrence of events *EventA* and *EventB* is strongly dependent on the time scale, *EventA* and *EventB* are time association events.
(2) Causal association events exist causality between events. If event *EventC* occurs, event *EventS* will occur. There is a strong causal relationship between the two events. Therefore, *EventC* and *EventS* are causal association events.
(3) Set associated events, the event set {*EventA1, EventA2, EventB1, EventB2,* …} leads to the occurrence of the event *EventD*, and the event set {*EventA1, EventA2, EventB1, EventB2,* …} and *EventD* are set associated events.

In real cases, the events that occur at different times are recorded to generate the event stream (referred to as "stream" hereinafter). For the stream processing, we consider three problems: (1) the same event occurs repeatedly at the adjacent positions on the timeline, and (2) different events occurs sequentially at the adjacent positions on the timeline, (3) different events are thrown into the stream in a disorderly sequence. The first problem is due to the continuous satisfaction of the trigger condition.

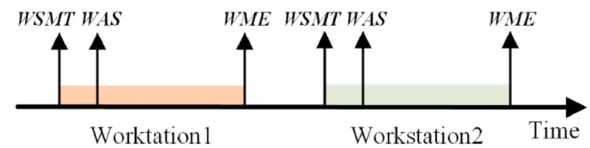

Fig. 8. Events involved in the process of machining ending and arriving at a new workstation.

For example, when the state of the machine tool is fixed, the same machine tool state event will be triggered multiple times. For this kind of problem, the repeated event data is filtered out through the customized data cleaning algorithm to ensure that the same event can only be triggered once in the same machining task. For the second and the third problems, the solution in our TCM system is to add a timestamp when the event occurs. The processing of different events in the stream is implemented using the open source component JBoss Drools Fusion [60]. JBoss Drools Fusion is used to detect events in real-time in the stream through the directed graph of rules. The association rules between events are established through non-time series and time series operators, as shown in Table 2. Fig. 7 shows the mechanism of event processing based on stream. The generated events are thrown into the stream, and the rules are matched in the rule base environment designed by experts in advance. Event detection is performed through the directed graph of rules. If there are events satisfying the association rules, the methods, new events, and other actions customized by experts in the "Action" part as shown in Fig. 7 will be triggered. When a new event is generated in the "Action" part, the new event will be thrown into the stream to continue rule matching. Through the above method, the processing of events that are very close in timeline can be realized.

Take the events that happen during the time when a workpiece is machined at two workstations as an example to illustrate the mechanism of event processing based on stream. As shown in Fig. 8, assume that a certain workpiece is to be machined in workstation1 and workstation2 in sequence. From leaving workstation1 to arriving workstation2, three events will be triggered in sequence i.e., workpiece machining end (*WME*), the workpiece being scanned by the smart mobile terminal (*WSMT*), and the workpiece arriving at the station (*WAS*). The above three events are repeated and scroll forward along the timeline in the fixed sequence at different workstations.

The relationship between the above three events is defined, as shown in Fig. 9. The ˆ symbol in the event indicates that the values of all attributes connected by the ˆ need to be satisfied.

Combining the time relationship of the three events of *WME*, *WSMT*, and *WAS* in Fig. 8, Fig. 9 is explained as follows. When the workpiece is scanned by the smart mobile terminal, a *WSMT* event is generated. Given that the *WME* event generated in workstation1 has existed in the stream, the *WAS* event will be then triggered in the presence of the events *WSMT* and *WME*. Among them, the *WME* event indicates that for the workpiece numbered with WP-2201, its 35−05 process has been machined at the SR240 station. Its next process is 35−10, and the next workstation is G3. The *WSMT* event indicates that the workpiece WP-2201 was scanned at the G3 workstation, and the process that needs to be machined is 35−10. *WSMT* occurs prior to *WME*, and the attribute values of "*workpiece_unm*", "*workpiece_nextProcessNum*", "*workpiece_nextStation*" in the *WME* event well match the corresponding attributes of "*workpiece_unm*", "*workpiece_currentProcessNum*" and "*workpiece_currentLocation*" in the *WSMT* event. Then it can be confirmed that the workpiece arrives at the new machining workstation correctly and then the *WAS* event is triggered subsequently.

4.3. TCM based on event processing

There are many real-time events in the manufacturing workshop. Table 3 summarizes some typical events relate to TCM in the manufacturing workshop. The simple events in the table are the original

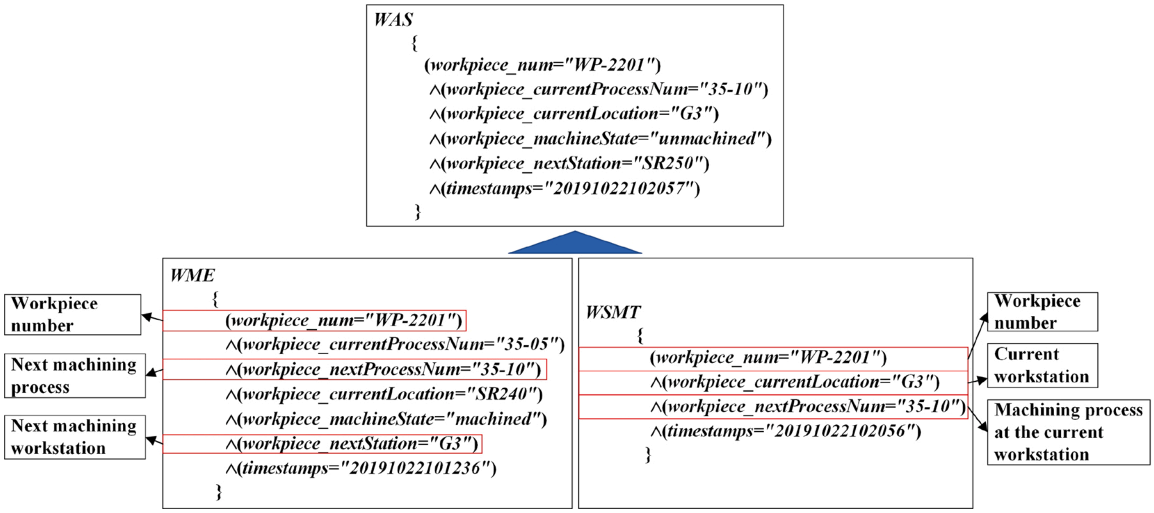

Fig. 9. The relationship between the three events of *WME*, *WSMT*, and *WAS*.

events generated by the manufacturing resources (CNC machine tools, smart mobile terminals, etc.) or the MES system. These simple events form complex events through the above-mentioned association rules.

According to the above sections, the monitoring of the tool is triggered by the complex event *WMS*. The processing of *WMS* triggering the *STLM* is shown in Fig.10. First of all, The *WMS* event is collected from the workpiece states application. Then, the *WMS* event is handed over to the engine distributor to load the available rules. Next, the event and corresponding rules are fed into the event processing engine to trigger the complex events by performing rule matching. Finally, the event processing engine feeds back the *STLM* event to the TCM application. The above steps process simple events into high-level information and provide more intuitive and meaningful data for TCM applications.

5. Event-driven tool life prediction

This section introduces the event-driven tool RUL prediction method, so we studied the event-driven tool RUL prediction flow and the key methods of RUL prediction.

5.1. Event-driven tool RUL prediction flow

In event-driven tool RUL prediction, the first step is to determine the start of the workpiece machining event, i.e., the *WMS* event, which is generated by multi-source events in the manufacturing workshop. As shown in expression (3), *WMS* is a complex event that is formed by the simple events *WAS*, *PMT*, *PPC*, and *MTSM* through the association rules described in Section 4.2. when a workpiece appears in a machine tool through the feedback of a smart mobile terminal, then it is determined that the position and machining task of the workpiece are consistent with the MES production planning, and all the production preparation tasks under the machine tool are have been completed, and the machine tool starts machining. Only when the above four conditions are met can the workpiece be judged to start machining at this station and *WMS* event can be generated.

$$[WAS; PMT; PPC; MTSM] \Rightarrow WMS \\ \Rightarrow : Thesymboloftriggerevent \tag{3}$$

Fig. 11 is an event-driven tool RUL prediction flow, which is a sub-flow of Fig. 3. *STLM* event triggers sensor data monitoring and predicting RUL method. Then, the Pass layer uses the sensor data for feature extraction and selects monotonic features (which will be introduced in Section 5.2). The principal component analysis (PCA) was used to reduce the feature dimension to construct the health indicator. The degradation model and Bayesian parameters update model are used to predict the RUL. Finally, it is judged whether the threshold is reached. If the threshold is reached, a *CCT* event is triggered. If the workpiece machining is completed, the *ETLM* event is triggered to end the tool RUL prediction method and save the RUL related data. Otherwise, the *STLM* event is triggered to continue to predict the RUL.

5.2. Tool RUL prediction based on Bayesian update

This section introduces the key methods in predicting RUL. First, features are extracted from the monitored signals and then fused by PCA method to form a health indicator. Second, the time series of the health indicator is used to update the parameters in the degradation models with the Bayesian method.

5.2.1. Build the health indicator

Features from time domain (Mean, Standard Deviation(Std), Root Mean Square (RMS), etc.), frequency domain (spectral kurtosis(SK)-related features (SKmean, SKSKewness, SKKurtosis, etc.) and time-frequency domain (the energy of wavelet coefficients) are extracted from the monitoring signal of the tool, At any time step i, the features form a feature vector x_i^j, where i={1,2,3,…, T}, T is the current time step, j={1,2,3…, N}, N is the number of the type of features. The feature matrix given T time steps is $X^{N\times T}$.

5.2.1.1. Select features. In order to select features with a high correlation with tool degradation, and optimize the analysis and prediction of tool RUL, we take the monotonicity score as the metric to select the features, i.e., the higher the score is, the better the feature is [45]. Eq. 4 shows the formula for computing the monotonicity score of the j-th feature, in which T is the total number of time steps. $diff\left(x_i^j\right)$ calculates the difference between the adjacent time steps of the j-th feature, i.e., $x_i^j - x_{i-1}^j$, and $positive\ \left(diff\left(x_i^j\right)\right)$ represents the number of positive differences between two adjacent time steps. Similar, $negative\ \left(diff\left(x_i^j\right)\right)$ indicates the number of negative differences between two adjacent time steps. Finally, N_m out of N features are selected and form the feature matrix $X^{N_m\times T}$, which will be fused to build the health indicator.

$$Monotonicity\left(x^j\right) = \frac{\left|positive\ \left(diff\left(x_i^j\right)\right) - negative\ \left(diff\left(x_i^j\right)\right)\right|}{T-1} \tag{4}$$

5.2.1.2. Build health indicator based on PCA and exponential smoothing.

Table 3
Typical events relate to TCM.

Manufacturing resources	Event type	Event	Relevance to TCM
Tool	Simple event	Reach the life threshold (*RLT*) Tool collapse edge (*TCE*)	Tool conditions related events for monitoring systems.
	Complex event	Start tool life monitoring (*STLM*) End tool life monitoring (*ETLM*) Tool breakage alarm (*TBA*) Tool life alarm (*TLA*)	Tool related events after event processing according to the conditions of the tool and the current machining task.
Machine Tool	Simple event	Spindle starts/stops (*SPSA/SPSP*) Coolant on/off (*CON/OFF*)	Provide events for judging machine tool running and machining task states.
	Complex event	Machine tool start machining (*MTSM*) Machine tool finish machining (*MTFM*) Change cut tool (*CCT*)	Provides machine tool running states related events for TCM.
Smart Mobile Terminal	Simple event	Feedback workpiece arrives (*FWA*) Feedback workpiece machining starts (*FWMS*) Feedback workpiece machining end (*FWME*) The workpiece is scanned by the smart mobile terminal (*WSMT*)	Provide feedback events for workpiece machining task states.
	Complex event	Production preparation completed (*PPC*)	Provide feedback events for TCM.
MES	Simple event	Planned workpieces arrive at the station (*PWAS*) Judge the machining states (*JMS*) Planned machining tasks (*PMT*)	Provides events for determining the states of the workpiece machining task.
	Complex event	Workpiece machining end (*WME*) Workpiece machining starts (*WMS*) Workpiece arrives at the station (*WAS*)	Provides workpiece machining task related events for TCM.

To reduce the dimension of the feature matrix $X^{N_m \times T}$, we adopt the PCA dimension reduction method to fusion the N_m features.

Before feature fusion, the z-score normalization method is used to normalize the feature matrix. The normalized feature matrix is denoted as $X_{norm}^{N_m \times T}$. The mathematical processing of PCA dimensionality reduction is given in Eq. 5- Eq.7, where C is the covariance matrix of $X_{norm}^{N_m \times T}$, Λ is a diagonal matrix, E is an eigenvector matrix after unitization. The matrix Y after dimensionality reduction can be obtained by solving the eigenvector of the covariance matrix.

$$C = X\frac{N_m \times T}{norm} \times X\frac{N_m \times T^T}{norm} \tag{5}$$

$$\Lambda = E^T CE \tag{6}$$

$$Y = E^T X\frac{N_m \times T}{norm} \tag{7}$$

We selected the first principal component data after PCA as the health indicator (*HI*). Besides, we have adopted an exponential smoothing strategy to optimize the *HI*.

Exponential smoothing as shown in Eq. 8, S_t is the smoothing value, α is the smoothing coefficient, y_t is the time series of the first principal component data, $S_t^{(1)}$, $S_t^{(2)}$, $S_t^{(3)}$ are the exponential smoothing values of the first, second and third times respectively. In this paper, we use cubic exponential smoothing, $i = 3$, $\alpha = 0.1$.

$$\begin{cases} S_t^{(1)} = \alpha y_t + (1-\alpha) S_{t-1}^{(1)} \\ S_t^{(2)} = \alpha S_t^{(1)} + (1-\alpha) S_{t-1}^{(2)} \\ S_t^{(3)} = \alpha S_t^{(2)} + (1-\alpha) S_{t-1}^{(3)} \end{cases}, HI = S_t^{(i)}, i = 1,2,3 \tag{8}$$

5.2.2. Prediction model based on Bayesian update

5.2.2.1. Exponential degradation model. The tool life prediction model proposed in this paper is the exponential degradation model [29], which is shown in Eq. 9. Φ is a constant, θ and β are random variable parameters subject to normal distribution. The time series of health indicator *HI* are used to update the joint distribution of θ and β with the sequential arrival of monitored data at each time step, ε is random error subject to normal distribution $N \sim (0, \sigma^2)$, σ^2 is a constant.

$$h(t) = \Phi + \theta \exp\left(\beta t + \varepsilon - \frac{\sigma^2}{2}\right) \tag{9}$$

For easier updating the parameters of the exponential model, we subtract Φ from both sides of Eq. 9 and then take the natural logarithm to obtained Eq. 10, in which $\theta' = \ln\theta - \frac{\sigma^2}{2}$, $\ln\theta \sim N(\mu_0, \sigma_0^2)$, $\beta \sim N(\mu_1, \sigma_1^2)$, $\theta' \sim N(\mu'_0, \sigma_0^2)$, $\varepsilon \sim N(0, \sigma^2)$. Now the goal is to estimate the joint

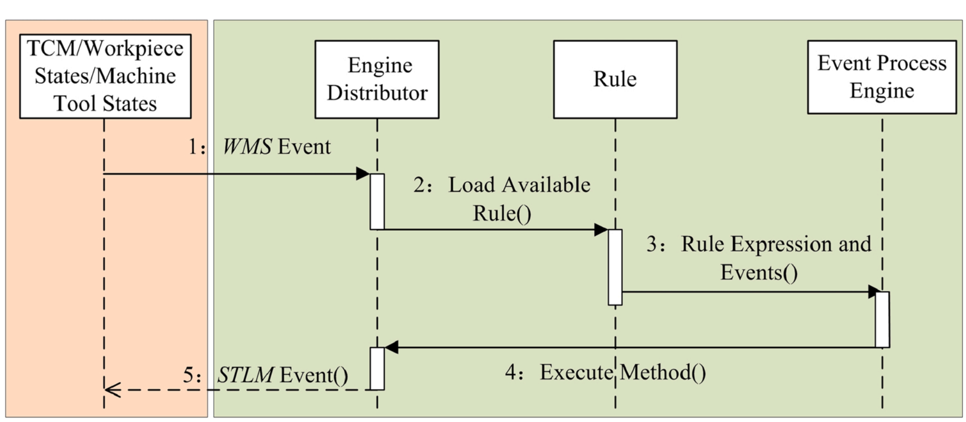

Fig. 10. Sequence diagram of event processing in the EDTCM.

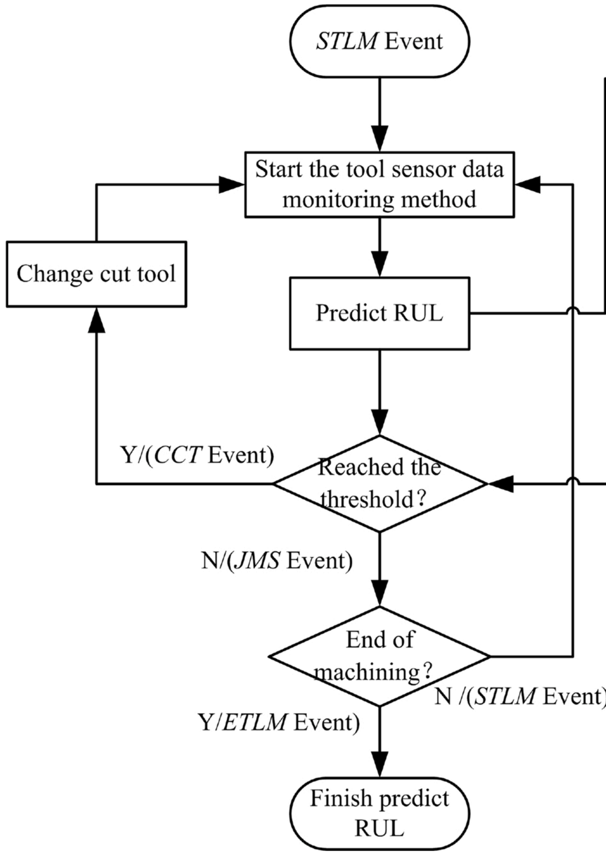

Fig. 11. Event-driven tool RUL prediction flow.

distribution of the unknown parameters θ' and β at each time step when the new sensor data arrive.

$$\begin{aligned} L(t) &= \ln\theta + \beta t + \varepsilon - \frac{\sigma^2}{2} \\ &= \theta' + \beta t + \varepsilon \end{aligned} \tag{10}$$

5.2.2.2. Parameters estimation based on Bayesian updating. The parameters of exponential degradation model are updated by Bayesian method [29]. When new sensor data are available, the posterior estimates of θ' and β are obtained according to Bayesian rules and new data. The likelihood function, i.e., the conditional joint density function of $L_1, \ldots, L_k$ is,

$$p(L_1, \ldots, L_k|\theta', \beta) = \left(\frac{1}{\sqrt{2\pi\sigma^2}}\right)^k \times \exp\left(-\sum_{i=1}^{k}\left(\frac{(L_i - \theta' - \beta t_i)^2}{2\sigma^2}\right)\right) \tag{11}$$

Secondly, because of the prior joint distribution of θ' and β belongs to one of the conjugate families of the sample distribution $p(L_1, \ldots, L_k|\theta', \beta)$, the posterior joint distribution of θ' and β still belong to a normal distribution. Therefore, $(\theta', \beta)\Big|L_1, \ldots, L_k \sim N\left(\mu_{\theta',k}, \sigma^2_{\theta',k}, \mu_{\beta,k}, \sigma^2_{\beta,k}, \rho_k\right)$. According to Bayesian theory that the posterior distribution is proportional to the product of prior distribution and likelihood function,

$$\begin{gathered} p(\theta', \beta|L_1, \ldots, L_k) \propto p(L_1, \ldots, L_k|\theta', \beta)p(\theta', \beta) \propto \\ \frac{1}{2\pi\sigma_{\theta',k}\sigma_{\beta,k}\sqrt{1-\rho_k^2}} \exp\left\{-\left[\frac{\sigma^2_{\beta,k}\left(\theta' - \mu_{\theta',k}\right)^2 - 2\sigma_{\theta',k}\sigma_{\beta,k}\rho_k\left(\theta' - \mu_{\theta',k}\right)\left(\beta - \mu_{\beta,k}\right) + \sigma^2_{\theta',k}\left(\beta - \mu_{\beta,k}\right)^2}{2\sigma^2_{\theta',k}\sigma^2_{\beta,k}\left(1-\rho_k^2\right)}\right]\right\} \end{gathered} \tag{12}$$

Given the observed data $L_1, \ldots, L_k$, the joint posterior distribution of (θ', β) at time step k is a bivariate normal distribution with mean $(\mu_{\theta',k}, \mu_{\beta,k})$, variance $\left(\sigma^2_{\theta',k}, \sigma^2_{\beta,k}\right)$, and correlation coefficient ρ_k. The parameter $\left\{\mu_{\theta',k}, \sigma^2_{\theta',k}, \mu_{\beta,k}, \sigma^2_{\beta,k}, \rho_k\right\}$ are obtained by Eq. 13-Eq. 17. Whenever a new observation data arrives, it will be utilized to update the posterior distribution $p(\theta', \beta|L_1, \ldots, L_k)$ and generate an updated degradation model.

$$\mu_{\theta',k} = \frac{\left(\sum_{i=1}^{k} L_i\sigma_0^2 + \mu'_0\right)\left(\sum_{i=1}^{k} t_i^2\sigma_1^2 + \sigma^2\right) - \left(\sum_{i=1}^{k} t_i\sigma_0^2\right)\left(\sum_{i=1}^{k} L_i t_i^2\sigma_1^2 + \mu_1\sigma^2\right)}{(k\sigma_0^2 + \sigma^2)\left(\sum_{i=1}^{k} t_i^2\sigma_1^2 + \sigma^2\right) - \left(\sum_{i=1}^{k} t_i\sigma_1^2\right)\left(\sum_{i=1}^{k} t_i\sigma_0^2\right)} \tag{13}$$

$$\mu_{\beta,k} = \frac{(k\sigma_0^2 + \sigma^2)\left(\sum_{i=1}^{k} L_i t_i^2\sigma_1^2 + \mu_1\sigma^2\right) - \left(\sum_{i=1}^{k} t_i\sigma_1^2\right)\left(\sum_{i=1}^{k} L_i\sigma_0^2 + \mu'_0\sigma^2\right)}{(k\sigma_0^2 + \sigma^2)\left(\sum_{i=1}^{k} t_i^2\sigma_1^2 + \sigma^2\right) - \left(\sum_{i=1}^{k} t_i\sigma_1^2\right)\left(\sum_{i=1}^{k} t_i\sigma_0^2\right)} \tag{14}$$

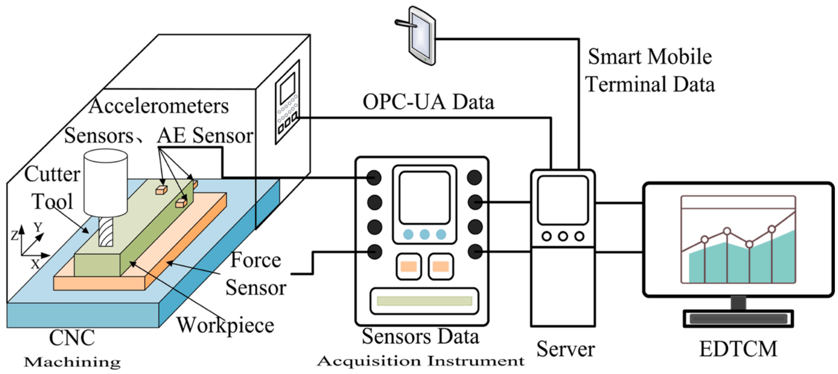

Fig. 12. Experimental scheme of TCM.

$$\sigma^2_{\theta',k} = \frac{\sigma^2\sigma_0^2\sigma_1^2}{\sigma_1^2}\frac{\sum_{i=1}^{k} t_i^2\sigma_1^2+\sigma^2}{(k\sigma_0^2+\sigma^2)\left(\sum_{i=1}^{k} t_i^2\sigma_1^2+\sigma^2\right)-\left(\sum_{i=1}^{k} t_i\right)^2\sigma_0^2\sigma_1^2} \tag{15}$$

$$\sigma^2_{\beta,k} = \frac{\sigma^2\sigma_0^2\sigma_1^2}{\sigma_0^2}\frac{k\sigma_0^2+\sigma^2}{(k\sigma_0^2+\sigma^2)\left(\sum_{i=1}^{k} t_i^2\sigma_1^2+\sigma^2\right)-\left(\sum_{i=1}^{k} t_i\right)^2\sigma_0^2\sigma_1^2} \tag{16}$$

$$\rho_k = \frac{-\sigma_0\sigma_1\sum_{i=1}^{k} t_i}{\sqrt{k\sigma_0^2+\sigma^2}\sqrt{\sigma_1^2\sum_{i=1}^{k} t_i^2+\sigma^2}} \tag{17}$$

6. Case study

This section first introduces the scene and scheme of the experiment, and then analyzes and verifies the experimental data step by step according to the data processing method mentioned in Sections 4–5. Finally, the experimental results are discussed and analyzed through a prototype system based on EDTCM.

6.1. Experimental scheme

The data used in the case study are from two sources, i.e., the public run-to-fail data of cutting tool from PHM2010 data challenge, and the private in-site data acquired by the authors. The in-site data include the CNC machine tool data acquired through the OPC-UA protocol, smart mobile terminals data, production planning data, etc. We use public datasets for the following reasons. The run-to-fail data is normally difficult to obtain in practice, and the datasets published by PHM2010 data challenge have been recognized as a benchmark and used by many researchers to validate their prediction models of tool RUL [46–48]. On the other hand, we employed the public data to verify the tool RUL prediction model detailed in section 5.2. The data and the prediction model are further integrated into the developed prototype system together with our in-site data to verify the EDTCM methodology.

Fig. 12 shows the experimental scheme of TCM. The machining workpiece is stainless steel (HRC52). The tool is a three-slot ball-end carbide tool, and the machining parameters are: (1) the spindle speed is 10400RPM, (2) the feed rate is 1555 mm/min, (3) the Y-cut Deep (radial) 0.125 mm, (4) Z-cut depth (axial) 0.2 mm, (5) Data collection frequency is 12 KHz/channel. Seven channels of sensor data are collected during the machining, including the cutting force and the vibration signals in three directions of XYZ, and one channel of acoustic emission signal. The data are collected by DAQ NI PCI1200 data acquisition instrument. The cutter's flank wear was measured after each complete surface cutting using a LEICA MZ12 microscopy system [49].

The OPC-UA data are obtained from the Siemens 840D CNC, including tool number, machining G code, spindle state, coolant state, etc. The smart mobile terminal data collected from workers include workstation data, workpiece scanning data, and machining state data of the workpiece. Production planning data include workshop scheduling data, basic workpiece data, and other data related to the workshop plan. Multi-source data are analyzed and fed back through the server and

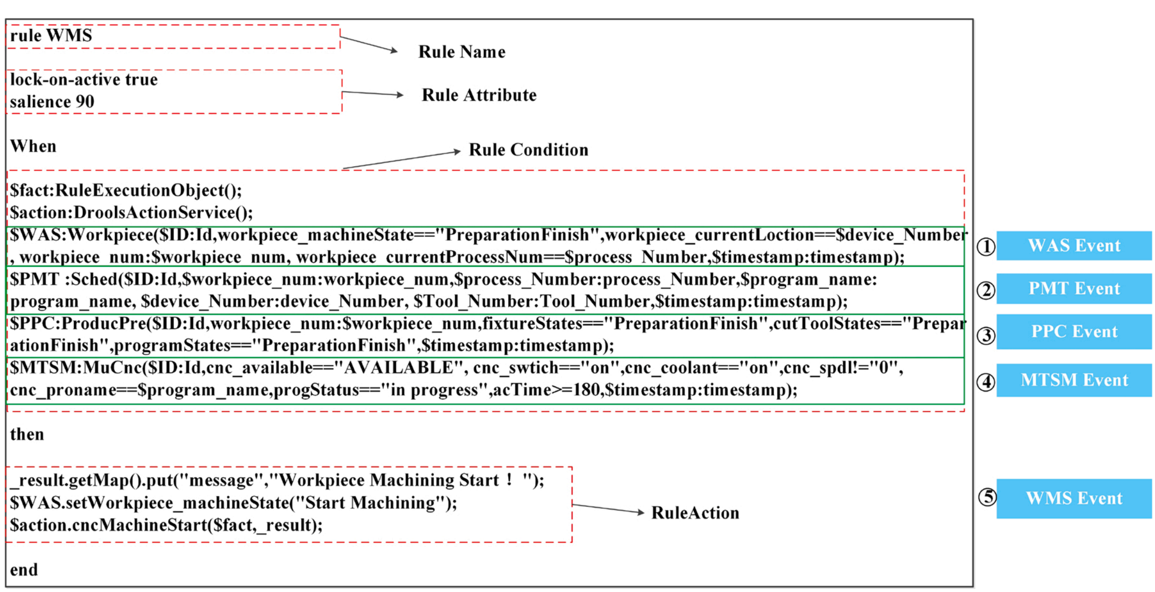

Fig. 13. Example of *WMS* event rule.

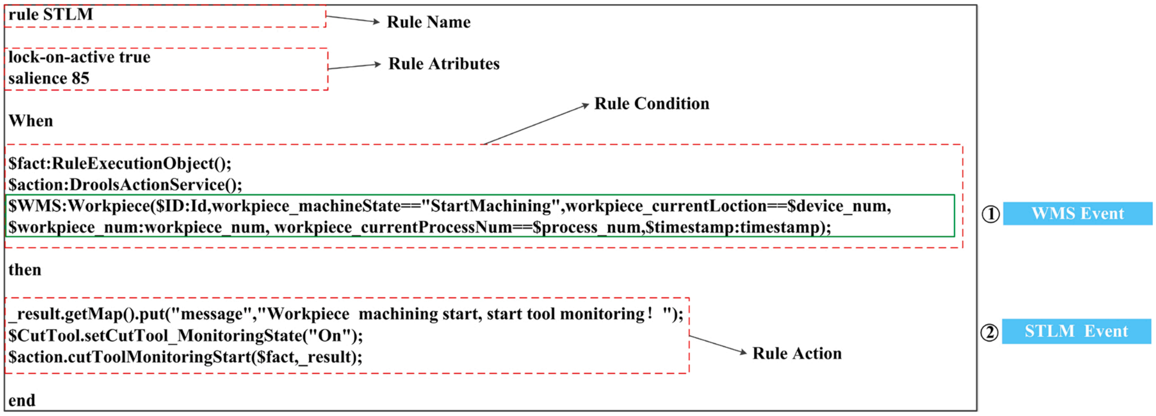

Fig. 14. Example of *STLM* event rule.

provides support for EDTCM in the workshop.

6.2. Data processing in the TCM process

During the process of EDTCM, the sensor data of the tool are driven by the *STLM* event to predict the tool RUL. Therefore, the following two sections introduce the *STLM* event and the results of analyzing the RUL with the sensor data.

6.2.1. Event processing for STLM

The *WMS* event is a key event in the EDTCM, which triggers the tool's *STLM* event. Fig. 13 shows a rule that triggers the *WMS* event.

The *WMS* event is driven by multi-source data such as production planning and production preparation data, workpiece data, and OPC-UA data. The detailed analysis is as follows. The rule expression shows in Fig. 13 is a set of associated events in Section 4.2. The rule condition part composed of four parts of data, the production planning data from the MES system machining task, the production preparation data about on-site production preparation task, the workpiece data collected by a smart mobile terminal, the OPC-UA data from the running of machine tools. These four parts of data represent the *WAS*, *PMT*, *PPC*, and *MTSM* four events. When the four events meet the set association rules, it will trigger the *WMS* event.

During the process of EDTCM, when the state of the workpiece changes to the start machining state, the *STLM* event will be triggered. Therefore, Fig. 14 shows the causal association rule for the *STLM* event. When the *WMS* event occurs, the state of the workpiece changes from the non-machining state to the start-machining state, and then the *STLM* event is triggered.

It should be pointed out that the rules involved in this paper mainly include two categories, one is the rules related to workpiece machining tasks, and the other is the rules related to tool monitoring tasks. According to the actual production needs of the target workshop that we investigated, we have fully considered the physical attributes of the relevant events involved in the target workshop in order to build the information model of the event that covers wide and complete attributes. Therefore, the information model generally does not change but the attribute values can be changed according to the real-time data in

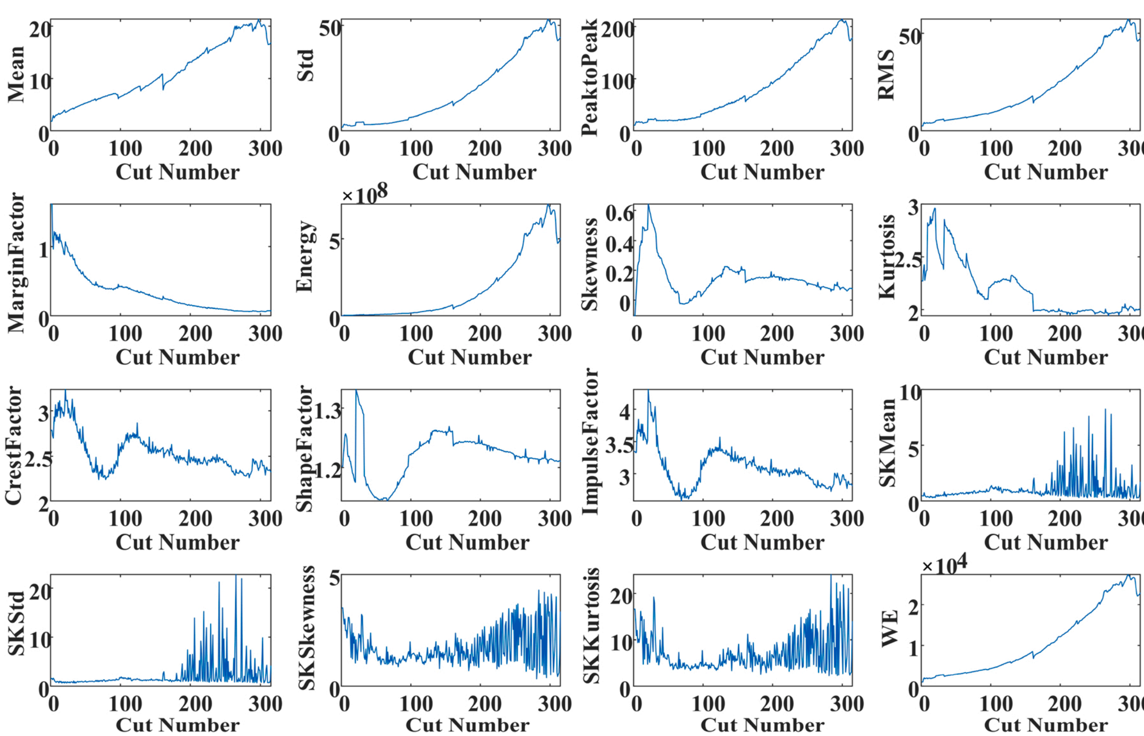

Fig. 15. Extracted features of force signal in the X-direction.

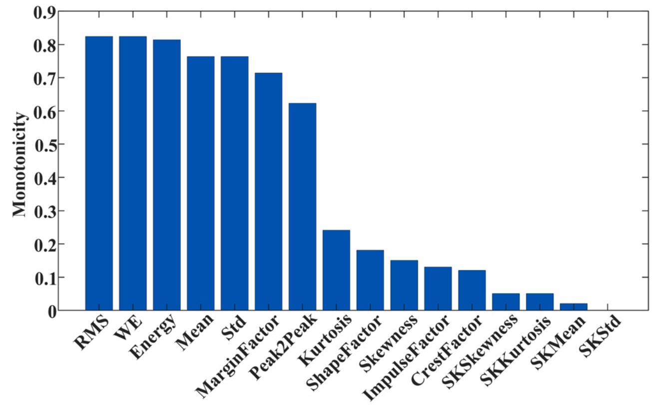

Fig. 16. Feature monotonicity ranking.

the workshop to adapt the process change. In other words, for different process steps, the judgment conditions of the rules are based on attribute variables, rather than customized instantiation rules for each process step. Therefore, the rules of this paper can adapt to the changes in process steps.

However, if new event attributes or new manufacturing resources are added in the process, the original rules will not be able to identify the events in the current process, and the system rules need to be expanded accordingly.

6.2.2. *Sensor data processing for tool RUL prediction*

6.2.2.1. Feature extraction and selection. The cutting force signal in X-direction is used to build the RUL prediction model. 16 features of time domain and time frequency domain are extracted from the raw signals, as shown in Fig. 15 The time-domain features include Mean, Standard Deviation(Std), PeaktoPeak, Root Mean Square (RMS), MarginFactor, Energy, Skewness, Kurtosis, CrestFactor, ShapeFactor, ImpulseFactor of the signals. The frequency domain contains spectral kurtosis(SK) related features, such as SKmean, SKStd, SKSKewness, SKKurtosis. The feature of time-frequency domain is the energy of the wavelet coefficient WE.

The features are selected according to their monotonicity score since the degradation process of tool wear is monotonic. The monotonicity score of each feature is calculated based on Eq.4 and scores of the 16 features are normalized to achieve the ranking of feature monotonicity, as shown in Fig. 16. We see that RMS has the highest rank of monotonicity, followed by WE and Energy. We choose the features whose monotonicity score is greater than 0.7 to better reflect the characteristics of the tool wear process and provide a better data basis for subsequent feature fusion. Finally, six features are selected including RMS, WE, Energy, Mean, Std, MarginFactor.

6.2.2.2. Build health indicator based on PCA. The six selected features are fused using PCA technique detailed in section 5.2.1 to build the health indicator. Fig. 17(a) shows the PCA1-PCA6 of the fused data, and the trend of each principal component over time can be obtained. PCA1 was selected as the health indicator due to the best monotonicity. The final health indicator curve (Fig.17(b)) is obtained after exponential

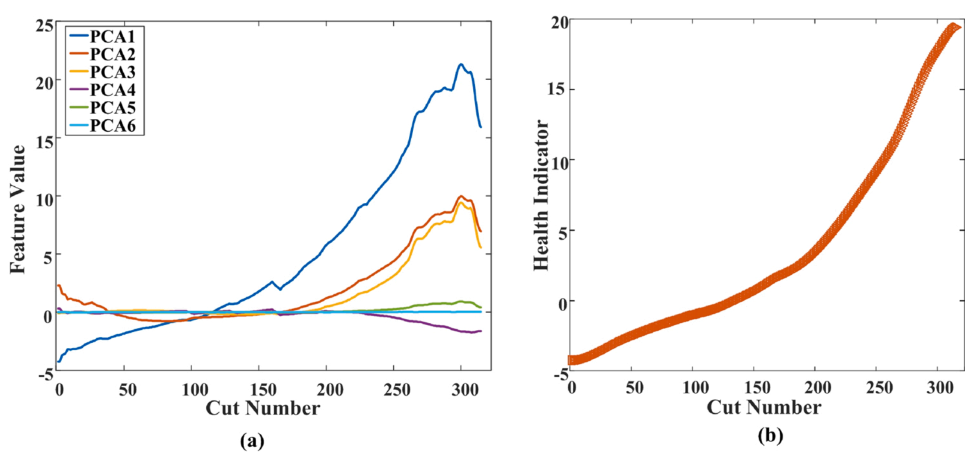

Fig. 17. (a) Principal components, (b)Health indicator.

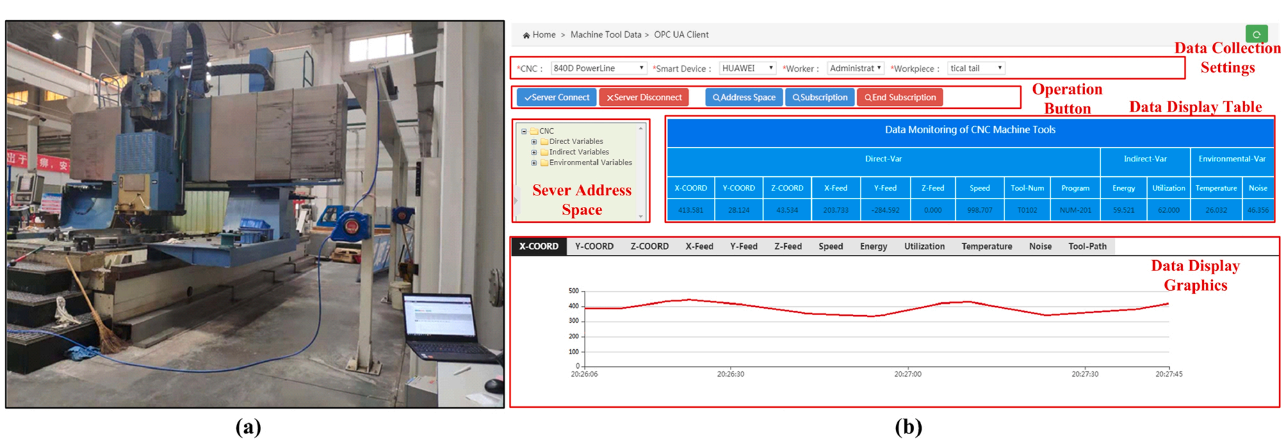

Fig. 18. Machine tool data collection based on OPC-UA: (a) CNC machine tool, (b) OPC-UA client.

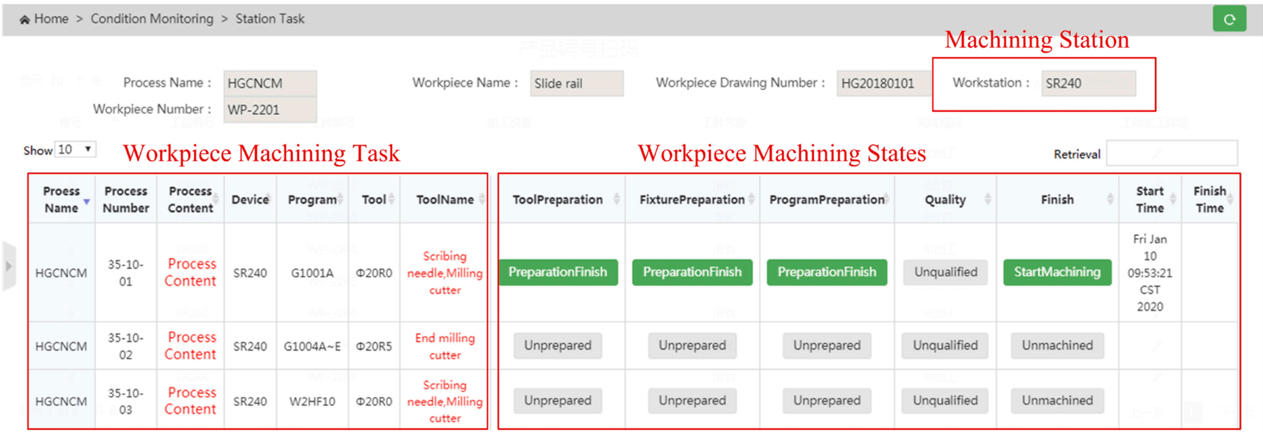

Fig. 19. Workpiece machining task and the machining states.

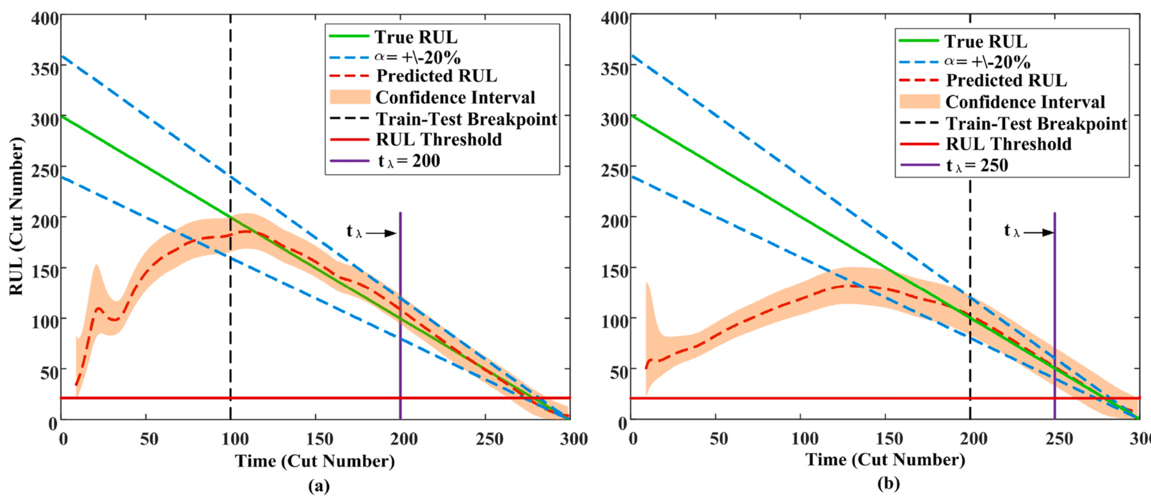

Fig. 20. Prediction effect of tool RUL under different steps ahead: (a) 200 steps ahead, (b) 100 steps ahead.

smoothing, which is used to build the model of tool RUL prediction.

6.3. Results and discussion

This section analyzes and discusses the practical workshop machining case and integrates the results of the public tool life dataset. The prototype system based on EDTCM is developed and the effectiveness of the OPC-UA data collection, machining state of the workpiece, RUL prediction of the tool, and the TCM interface are discussed.

6.3.1. Machine tool data collection

Fig. 18 shows the client that collects CNC machine tool data using OPC-UA standard. The CNC machine tool in Fig. 18 (a) is equipped with the Siemens 840D system, which outputs data in the OPC protocol. Through OPC-UA protocol conversion, the machine tool data can be better integrated with the upper-layer system. Fig. 18 (b) shows the OPC-UA client, which is used to collect and monitor machine tool running data, including tool coordinates, feed rate, spindle load, spindle speed, program name, tool number, etc. These data help judge whether the workpiece is in the machining state.

6.3.2. Workpiece machining state data

In the process of EDTCM, the current machining state of the workpiece is a prerequisite for monitoring the tool just in time, which avoids unnecessary long-term data acquisition. Fig. 19 shows one of the user interfaces of the developed prototype system, from which the real-time machining states of the workpiece can be viewed at any machining station in the workshop. The machining states of the workpiece at different stages are obtained by processing the events generated by machine tools, smart mobile terminal, MES, etc. using the event processing mechanism introduced in Section 4.2.

6.3.3. Tool life prediction effect

Fig. 20 shows the tool RUL prediction results versus time. It should be noted that true RUL is unknown in practice and here is only used to verify the effectiveness of the proposed algorithm. Fig.20 shows that in the earlier time of prediction the error between the predicted RUL and the true RUL is large. This is due to insufficient observations resulting in poor estimations for the degradation model parameters. But with more observations available, the predicted RUL getting more and more closer to the true RUL.

The metric $\alpha - \lambda$ accuracy proposed by Saxena et al. [61] is used to evaluate the prediction results. $\alpha - \lambda$ accuracy determines whether a prediction result falls within the α accuracy zone at a specific cycle t_λ. The accuracy zone varies with $\pm\alpha$ ratio to the true RUL. The blue dashed line in the figure is the $\pm\alpha$ zone corresponding to $\alpha = 0.2$. The specific cycle t_λ is expressed with a fraction of λ between starting cycle of RUL prediction t_s (corresponding to$\lambda = 0$) and the true end of life t_{EOL} (corresponding to$\lambda = 1$), which is $t_\lambda = t_s + \lambda(t_{EOL} - t_s)$. The black dashed line in the figure indicates the breakpoint t_s. The observations before breakpoint were used to estimate the degradation model parameters and the right side is the prediction interval. Here $t_{EOL} = 300$, $\lambda = 0.5$. The

Table 4
Comparison of [62] and the proposed method under different steps ahead for RUL prediction.

Prediction errors	200 steps ahead prediction	100 steps ahead prediction
[62]	9.6 %	3.5 %
The proposed method	8.65 %	2.34 %

starting cycle of RUL prediction t_s can be at any time. Normally, the earlier the prediction starts, the more uncertainty the prediction results will be due to insufficient observations. Here we take $t_s = 100$ and 200 in order to compare with other work. Therefore, $t_\lambda = 200$ and 250. It can be seen from the figure that the $\alpha - \lambda$ index value is "True" in the prediction of 100 and 200 steps ahead. The red solid line is the RUL threshold. When the predicted RUL is below this threshold, the "alarm" will be triggered and predictive maintenance will be scheduled in advance to reduce the downtime of the machine tool.

Table 4 shows a comparison experiment with the results of the reference [62]. The prediction errors of reference [62] are 9.6 % and 3.5 % respectively when 200 and 100 steps ahead. The prediction errors of the method proposed in this paper are 8.65 % and 2.34 % respectively when 200 and 100 steps ahead. The comparison results are reported in Table 4, which indicates compared with reference [62] the proposed method reduces the prediction error by 9.8 % and 33 % respectively when 200 and 100 steps ahead. Both the methods have a large prediction error under 200 steps ahead. This is due to insufficient observations resulting in poor estimations for the degradation model parameters. When the prediction is made 100 steps ahead, the method proposed in this paper has a smaller prediction error.

6.3.4. TCM interface

According to the analysis of the workpiece machining states data and machine tool running data, the developed prototype system based on the EDTCM architecture described in Section3–5 performs the necessary calculation and analysis during tool machining to avoid unnecessary waste of resources. The EDTCM makes the TCM process under machining tasks. The TCM interface is shown in Fig. 21, showing the basic information of the current machining of a station, the monitored tool information. The conditions of the tool includes the state, RUL, alarm, and other functions. Besides, TCM under machining tasks can provide strong machining task-related tool data support for scheduling and machining parameter optimization in workshop management.

7. Conclusion and future work

This paper proposed an EDTCM methodology considering tool life prediction based on the Industrial Internet and developed a prototype system that verifies the effectiveness of the EDTCM through case studies.

The detailed contributions are as follows. First of all, the architecture of EDTCM based on Industrial Internet is proposed. Under this architecture, OPC-UA is used to collect underlying data based on the standard protocol. It also provides decision support for TCM through event modeling from multiple manufacturing resources. Secondly, EDTCM aims to combine the machining tasks with TCM and predict the RUL of the tool when the workpiece machining starts. Therefore, based on the concept of event-driven, TCM is realized by using effective cutting data at the right time. Finally, the analysis of multi-source data such as machine tool running data, workpiece machining states data, production planning data, and smart mobile terminal data based on the event-driven approach, which can more accurately determine the machining states of the workpiece. Therefore, the monitoring effect of the tool is more reliable and the TCM bind with machining task can provide strong machining task-related tool data support for scheduling and machining parameter optimization in workshop management.

The method given in this paper is effective through experimental data test results, but there are still shortcomings. Firstly, this paper is only an integrated verification for the combination of practical scenarios and public datasets. The application effect in practical machining scenarios needs further research. Secondly, the RUL prediction effect of multiple types of tools under multiple cutting parameters still needs further research. Thirdly, how to extend the tool life by optimizing the machining parameters after predicting the tool RUL, or to guarantee the machining quality by optimizing the scheduling of workshop resources, etc., requires more in-depth research. Fourth, this paper unifies the communication protocol of the machine tool based on OPC-UA to solve the problem of equipment interface diversity. However, it is worth mentioning that UMATI is a machine tool interface standard based on the OPC-UA initiative by the German Machine Tool Builders' Association (VDW). The combination of UMATI and OPC-UA will make the management of heterogeneous equipment in the workshop more convenient. Therefore, the unification of the machine tools data interface based on UMATI is also a topic worth studying. Besides, in terms of system stability, in the future, it is necessary to consider the high concurrency of events that exist when lots of equipment is connected in the workshop. How to effectively solve the event processing in a high

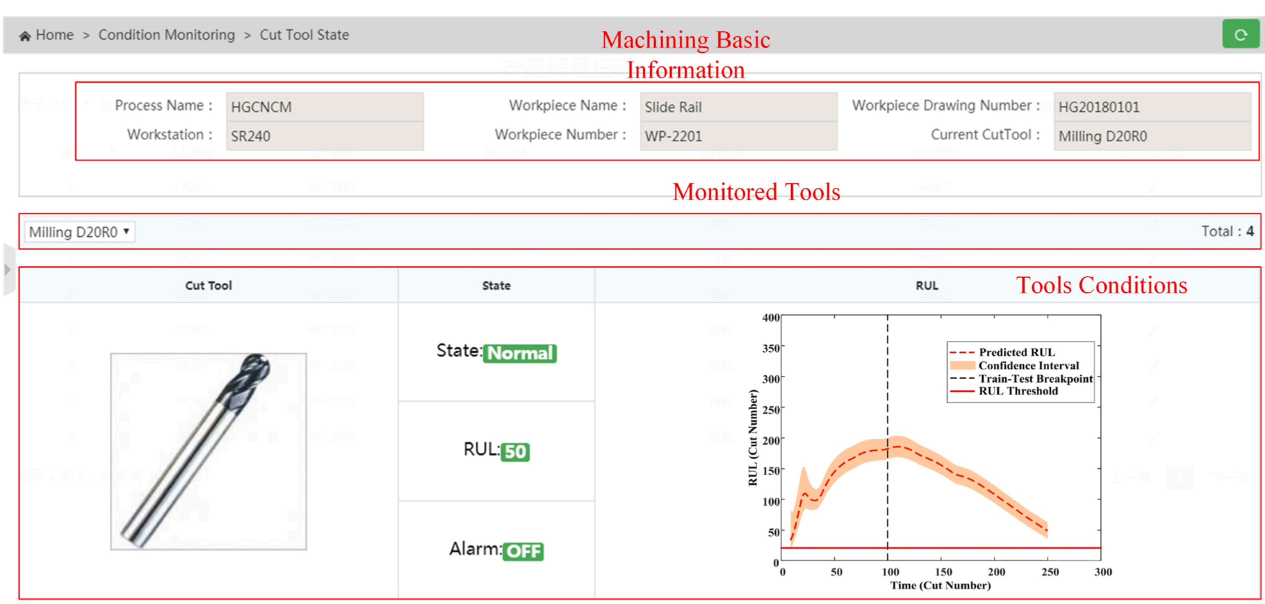

Fig. 21. TCM interface.

concurrency state is a problem worthy of in-depth study.

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.

Acknowledgments

This research is funded by the National Natural Science Foundation of China (No.51805262, No.51775024), the Civil Airplane Technology Development Program (MJ-2016-G-59), and the Beijing Key Laboratory of Digital Design and Manufacturing Project.

References

[1] Liu R, Kothuru A, Zhang S. Calibration-based tool condition monitoring for repetitive machining operations. J Manuf Syst 2020;54:285–93. https://doi.org/10.1016/j.jmsy.2020.01.005.
[2] Wong SY, Chuah JH, Yap HJ. Technical data-driven tool condition monitoring challenges for CNC milling: a review. Int J Adv Manuf Technol 2020;107:4837–57. https://doi.org/10.1007/s00170-020-05303-z.
[3] Noël M, Sodhi MS, Lamond BF. Tool planning for a lights-out machining system. J Manuf Syst 2007;26(3-4):161–6. https://doi.org/10.1016/j.jmsy.2007.10.004.
[4] Niu BY, Sun J, Yang B. Multisensory based tool wear monitoring for practical applications in milling of titanium alloy. Mater Today Proc 2020;22:1209–17. https://doi.org/10.1016/j.matpr.2019.12.126.
[5] Li ZX, Liu R, Wu DZ. Data-driven smart manufacturing: tool wear monitoring with audio signals and machine learning. J Manuf Process 2019;48:66–76. https://doi.org/10.1016/j.jmapro.2019.10.020.
[6] Wang JJ, Wang P, Gao RX. Enhanced particle filter for tool wear prediction. J Manuf Syst 2015;36:35–45. https://doi.org/10.1016/j.jmsy.2015.03.005.
[7] Martin W, Thilo S, Juergen J. The future of industrial communication: automation networks in the era of the internet of things and industry 4.0. IEEE Ind Electron M 2017;11(1):17–27. https://doi.org/10.1109/MIE.2017.2649104.
[8] Shao W, Ding H, Tang JY, Peng SD. A data-driven optimization model to collaborative manufacturing system considering geometric and physical performances for hypoid gear product. Robot Comput Integr Manuf 2018;54:1–16. https://doi.org/10.1016/j.rcim.2018.05.004.
[9] Wang JL, Xu CQ, Zhang J, Bao JS, Zhong R. A collaborative architecture of the industrial internet platform for manufacturing systems. Robot Comput Integr Manuf 2020;61:101845. https://doi.org/10.1016/j.rcim.2019.101854.
[10] Bousdekis A, Papageorgiou N, Magoutasa B, Apostoloua D, Mentzas G. Enabling condition-based maintenance decisions with proactive event-driven computing. Comput Ind 2018;100:173–83. https://doi.org/10.1016/j.compind.2018.04.019.
[11] Esmaeilian B, Behdad S, Wang B. The evolution and future of manufacturing: a review. J Manuf Syst 2016;39:79–100. https://doi.org/10.1016/j.jmsy.2016.03.001.
[12] Wang LH, Adamson G, Holm M, Moore P. A review of function blocks for process planning and control of manufacturing equipment. J Manuf Syst 2012;31(3):269–79. https://doi.org/10.1016/j.jmsy.2012.02.004.
[13] Yang C, Peng T, Lan S, Lan SL, et al. Towards IoT-enabled dynamic service optimal selection in multiple manufacturing clouds. J Manuf Syst 2020;56:213–26. https://doi.org/10.1016/j.jmsy.2020.06.004.
[14] Mohanraj T, Shankar S, Rajasekar R, Sakthivel NR, Pramanik A. Tool condition monitoring techniques in milling process-a review. J Mater Res Technol 2020;9(1):1032–42. https://doi.org/10.1016/j.jmrt.2019.10.031.
[15] Zhou CA, Guo K, Sun J, Yang B, Liu JW, Song G, et al. Tool condition monitoring in milling using a force singularity analysis approach. Int J Adv Manuf Technol 2020;107:1785–92. https://doi.org/10.1007/s00170-019-04664-4.
[16] Zhou CA, Yang B, Guo K, Liu JW, Sun J, Song G, et al. Vibration singularity analysis for milling tool condition monitoring. Int J Mech Sci 2020;166:105254. https://doi.org/10.1016/j.ijmecsci.2019.105254.
[17] Paul G, Dimitrios P, Andrew W. A cyber physical system for tool condition monitoring using electrical power and a mechanistic model. Comput Ind 2020;118:103223. https://doi.org/10.1016/j.compind.2020.103223.
[18] Zhang XY, Lu X, Wang S, Wang W, Li WD. A multi-sensor based online tool condition monitoring system for milling process. 51st CIRP Conference on Manufacturing Systems, 72; 2018. p. 1136–41. https://doi.org/10.1016/j.procir.2018.03.092.
[19] Dai Y, Zhu KP. A machine vision system for micro-milling tool condition monitoring. Precis Eng 2017;52:183–92. https://doi.org/10.1016/j.precisioneng.2017.12.006.
[20] Ratava J, Lohtander M, Varis J. Tool condition monitoring in interrupted cutting with acceleration sensors. Robot Comput Integr Manuf 2017;47:70–5. https://doi.org/10.1016/j.rcim.2016.11.008.
[21] Drouillet C, Karandikar J, Nath C, Journeaux AC, Mansori ME, Kurfess T. Tool life predictions in milling using spindle power with the neural network technique. J Manuf Process 2016;22:161–8. https://doi.org/10.1016/j.jmapro.2016.03.010.
[22] Zheng S, Ristovski K, Farahat A, Gupta C. Long short-term memory network for remaining useful life estimation. IEEE International Conference on Prognostics and Health Management (ICPHM) 2017. https://doi.org/10.1109/ICPHM.2017.7998311.
[23] Sun C, Ma M, Zhao Z, Tian S, Yan R, Chen X. Deep transfer learning based on sparse autoencoder for remaining useful life prediction of tool in manufacturing. IEEE T Ind Inform 2019;15(4):2416–25. https://doi.org/10.1109/TII.2018.2881543.
[24] Mikołajczyk T, Nowicki K, Bustillo A, Pimenov DY. Predicting tool life in turning operations using neural networksand image processing. Mech Syst Signal Pr 2018;104:503–13. https://doi.org/10.1016/j.ymssp.2017.11.022.
[25] Corne R, Nath C, El Mansori M, Kurfess T. Study of spindle power data with neural network for predicting real-time tool wear/breakage during inconel drilling. J Manuf Syst 2017;43:287–95. https://doi.org/10.1016/j.jmsy.2017.01.004.
[26] Zhu KP, Liu TS. On-line tool wear monitoring via hidden semi-markov model with dependent durations. IEEE T Ind Inform 2017;99:69–78. https://doi.org/10.1109/TII.2017.2723943.
[27] Wang JJ, Xie JY, Zhao R, Zhang LB, Duan LX. Multisensory fusion based virtual tool wear sensing for ubiquitous manufacturing. Robot Comput Integr Manuf 2017;45:47–58. https://doi.org/10.1016/j.rcim.2016.05.010.
[28] Gebraeel NZ, Lawley MA, Rong LI, Ryan JK. Residual-life distributions from component degradation signals: A bayesian approach. IIE Trans 2005;37(6):543–57. https://doi.org/10.1080/07408170590929018.
[29] Li NP, Lei YG, Lin J, Ding SX. An improved exponential model for predicting remaining useful life of rolling element bearings. IEEE T Ind Electron 2015;62(12):7762–73. https://doi.org/10.1109/TIE.2015.2455055.
[30] Karandikar JM, Abbas AE, Schmitz TL. Tool life prediction using bayesian updating. Part 2: turning tool life using a markov Chain Monte carlo approach. Precis Eng 2014;38(1):18–27. https://doi.org/10.1016/j.precisioneng.2013.06.007.
[31] Karandikar JM, Abbas AE, Schmitz TL. Tool life prediction using bayesian updating. Part 1: milling tool life model using a discrete grid method. Precis Eng 2014;38(1):9–17. https://doi.org/10.1016/j.precisioneng.2013.06.006.
[32] Wang YW, Gogu C, Binaud N, BES C, Fu J. A model-based prognostics method for fatigue crack growth in fuselage panels. Chin J Aeronaut 2019;32(2):396–408. https://doi.org/10.1016/j.cja.2018.11.010.
[33] Wang JF, Huang YQ, Chang Q, Li SQ. Event-driven online machine State decision for energy-efficient manufacturing system based on digital twin using max-plus algebra. Sustainability 2019;11:5036. https://doi.org/10.3390/su11185036.
[34] Mourtzis D, Vlachou K, Xanthopoulos N, Givehchi M, Wang LH. Cloud-based adaptive process planning considering availability and capabilities of machine tools. J Manuf Syst 2016;39:1–8. https://doi.org/10.1016/j.jmsy.2016.01.003.
[35] Wang LH, Hao Q, Shen WM. A novel function block based integration approach to process planning and scheduling with execution control. Int J Manuf Technol Manage 2007;11(2):228–50. https://doi.org/10.1504/IJMTM.2007.013193.
[36] Adamson G, Wang LH, Moore P. Feature-based control and information framework for adaptive and distributed manufacturing in cyber physical systems. J Manuf Syst 2017;43:305–15. https://doi.org/10.1016/j.jmsy.2016.12.003.
[37] Yao X, Zhang J, Li Y, Zhang C. Towards flexible RFID event-driven integrated manufacturing for make-to-order production. Int J Comput Integ M 2018;31(3):228–42. https://doi.org/10.1080/0951192X.2017.1407455.
[38] Chen WC, Liu HW, Qi ES. Discrete event-driven model predictive control for real-time work-in-process optimization in serial production systems. J Manuf Syst 2020;55:132–42. https://doi.org/10.1016/j.jmsy.2020.03.002.
[39] Cao W, Jiang PY, Lu P, Liu B, Jiang KY. Real-time data-driven monitoring in job-shop floor based on radio frequency identification. Int J Adv Manuf Technol 2017;92(5-8):2099–120. https://doi.org/10.1007/s00170-017-0248-7.
[40] Wang C, Jiang PY, Lu TT. Production events graphical deduction model enabled real-time production control system for smart job shop. P I Mech Eng C-J Mec 2018:1989–96. https://doi.org/10.1177/0954406217728531.
[41] Berger C, Zipfel A, Braunreuther S, Reinhart G. Approach for an event-driven production control for cyber-physical production systems. 12th CIRP Conference on Intelligent Computation in Manufacturing Engineering, 79; 2019. p. 349–54. https://doi.org/10.1016/j.procir.2019.02.085.
[42] Theorin A, Bengtsson K, Provost J, Lieder M, Johnsson C, Lundholm T, et al. An event-driven manufacturing information system architecture for industry 4.0. Int J Prod Res 2016;55(5–6):1297–311. https://doi.org/10.1080/00207543.2016.1201604.
[43] Umer M, Mahesh B, Hanson L, Khabbazi MR, Onori M. Smart power tools: An industrial event-driven architecture implementation. 51st CIRP Conference on Manufacturing Systems, 72; 2018. p. 1357–61. https://doi.org/10.1016/j.procir.2018.03.058.
[44] Luckham D. The power of events. Addison Wesley; 2002.
[45] Saidi L, Ben AJ, Bechhoefer E, Benbouzid M. Wind turbine high-speed shaft bearings health prognosis through a spectral kurtosis-derived indices and SVR. Appl Acoust 2017;120:1–8. https://doi.org/10.1016/j.apacoust.2017.01.005.
[46] Sun HB, Zhang JD, Mo R, Zhang XZ. In-process tool condition forecasting based on a deep learning method. Robot Comput Integr Manuf 2020;64:101924. https://doi.org/10.1016/j.rcim.2019.101924.
[47] Wang JJ, Yan JX, Li C, Gao RX, Zhao R. Deep heterogeneous GRU model for predictive analytics in smart manufacturing: application to tool wear prediction. Comput Ind 2019;111:1–14. https://doi.org/10.1016/j.compind.2019.06.001.
[48] Javed K, Gouriveau R, Li X, Zerhouni N. Tool wear monitoring and prognostics challenges: a comparison of connectionist methods toward an adaptive ensemble model. J Intell Manuf 2016;29(8):1873–90. https://doi.org/10.1007/s10845-016-1221-2.

[49] Li X, Lim BS, Zhou JH, Huang S, Phua SJ, Shaw KC, et al. Fuzzy neural network modeling for tool wear estimation in dry milling operation. Annual Conference of the Prognostics and Health Management Society 2009:1–11. http://www.phmsociety.org/node/79.
[50] Ren J, Guo H, Xu CG, Zhang YX. Serving at the Edge: A scalable IoT architecture based on transparent computing. IEEE Netw 2017;(5):12–21. https://doi.org/10.1109/MNET.2017.1700030. PP.
[51] Cao KY, Liu YF, Meng GJ, Sun QM. An overview on edge computing research. IEEE Access 2020:85714–28. https://doi.org/10.1109/ACCESS.2020.2991734.
[52] Sittón-Candanedo I, Alonso RS, Corchado JM, Rodríguez-Gonzáleza S, Casado-Varaa R. A review of edge computing reference architectures and a new global edge proposal. Future Gener Comp SY 2019;(99):278–94. https://doi.org/10.1016/j.future.2019.04.016.
[53] Khan WZ, Ahmed E, Hakak S, Yaqoob I, Ahmed A. Edge computing: a survey. Future Gener Comp SY 2019;97:219–35. https://doi.org/10.1016/j.future.2019.02.050.
[54] Kong TX, Hu TL, Zhou TT, Ye YX. Data construction method for the applications of workshop digital twin system. J Manuf Syst 2020. https://doi.org/10.1016/j.jmsy.2020.02.003.
[55] Hu Y, Zheng LY, Wang YH, Fan W. Semantic information model and mobile smart device enabled data acquisition system for manufacturing workshop. J Phys Conf Ser 2018;1074(1). https://doi.org/10.1088/1742-6596/1074/1/012148.
[56] Lei P, Zheng LY, Wang LH, et al. MTConnect compliant monitoring for finishing assembly interfaces of large-scale components: a vertical tail section application. J Manuf Syst 2017;45:121–34. https://doi.org/10.1016/j.jmsy.2017.09.001.
[57] Mohring HC, Wiederkehr P, Erkorkmaz K, Kakinuma Y. Self-optimizing machining systems. CIRP Ann-Manuf Techn 2020. https://doi.org/10.1016/j.cirp.2020.05.007.
[58] https://opcfoundation.org/markets-collaboration/mtconnect/.
[59] https://opcfoundation.org/markets-collaboration/umati/.
[60] https://www.drools.org/.
[61] Saxena A, Celaya J, Saha B, et al. On applying the prognostic performance metrics. Paper Presented at the Annual Conference of the Prognostic and Health Management Society, San Diego, California, USA, 27 September-1 October 2009.
[62] Wang P, Gao RX. Adaptive resampling-based particle filtering for tool life prediction. J Manuf Syst 2015;37:528–34. https://doi.org/10.1016/j.jmsy.2015.04.006.