J Intell Manuf (2019) 30:2681–2699
https://doi.org/10.1007/s10845-017-1322-6

Cloud manufacturing based service encapsulation and optimal configuration method for injection molding machine

Yingfeng Zhang[1,2] · Dong Xi[1] · Haidong Yang[3,4] · Fei Tao[5] · Zhe Wang[1]

Received: 29 September 2016 / Accepted: 27 March 2017 / Published online: 1 April 2017

Abstract This paper aims to introduce the concept of cloud manufacturing (CMfg) in the injection molding industry. The CMfg platform for injection molding enterprises is built to improve the sharing, circulation and integration of the injection molding resources. With the implementation of the Internet of Things technologies in the traditional injection molding shop, the real-time manufacturing information of resources can be accurately captured and the entire molding process becomes more visible and traceable. The virtual machining service of the injection molding machine is encapsulated as a cloud service that published into the platform for on-demand use. When task orders are published, through the presented task-driven proactive service discovery method, competent services can be quickly found. The custom-oriented evaluation method based on technique for order preference by similarity to ideal solution is designed to help the demanders to find satisfying services according to their customized criteria. Since the task orders arrive dynamically, after these orders are assigned to the specified machine, a real-time order dispatching mechanism is developed to provide an optimal scheduling plan for the cloud service. Finally, the proposed framework and methods are illustrated by a numerical simulation.

Keywords Cloud manufacturing · Injection molding machine · Proactive discovery · Service evaluation · Optimal scheduling

Introduction

In today's manufacturing industry, the increasingly fierce global market and the need to satisfy the changing demands from the customers are posing challenges for the modern enterprises to remain competitive or even just to survive. Among them, small- and medium-sized enterprises (SMEs) are of the great majority. Compared to major companies, most SMEs face the challenges of (1) limited capital, (2) weak research and development, (3) deficient managerial and technical talent, (4) insufficient market information, etc. (Farsi and Toghraee 2014). When market changes or the fluctuations of employees happen (Wilde 2010), SMEs are more vulnerable and more likely to trap into the production dilemma. Take the injection molding industry in Guangdong Province of China as an example. Most injection molding enterprises are small- and medium-sized and widely distributed. Many small enterprises do not have sufficient access to trade-related information. This may lead to their potential loss of business opportunities and suffer from inadequate task orders. They lie in the upper section of the industrial chain and are characterized to be both technology-intensive and labor-intensive. On the one hand, injection molding features strong

✉ Yingfeng Zhang
zhangyf@nwpu.edu.cn

[1] Key Laboratory of Contemporary Design and Integrated Manufacturing Technology, Ministry of Education, School of Mechanical Engineering, Northwestern Polytechnical University, Xi'an, Shaanxi, People's Republic of China

[2] Research and Development Institute in Shenzhen, Northwestern Polytechnical University, Shenzhen 518057, People's Republic of China

[3] School of Mechanical Science and Engineering, Huazhong University of Science and Technology, Wuhan, Hubei, People's Republic of China

[4] School of Mechanical Engineering, Guangdong University of Technology, Guangzhou, People's Republic of China

[5] School of Automation Science and Electrical Engineering, Beihang University, Beijing, People's Republic of China

design capability, self-innovation and high technology for the complicated molding process and product customization. On the other hand, increasing injection molding enterprises are intensifying the market competition and compressing the profit margins. SMEs that lack core technologies can only undertake relatively simple manufacturing tasks at the bottom of value chains and make modest profits (Huang et al. 2013). Additionally, they lack intellectual capital in production management. Nevertheless, they seldom purchase decision tools such as scheduling software to improve their production processes, due to the high price offered by companies who provide the enterprise information system (EIS) (Wang et al. 2014).

In recent years, the traditional manufacturing paradigm has been shifting towards the service-oriented manufacturing (SOM) (de Oliveira and Silva 2015), driven by the growing applications of information and sensor technologies like cloud computing (Puttonen et al. 2016), Internet of Things (IoT) (Welbourne et al. 2009), radio frequency identification (RFID) in the industrial field. By combining the advanced technologies above, a new computing and service-oriented manufacturing mode emerges, i.e. CMfg (Li et al. 2010). In CMfg, manufacturing resources and capabilities are virtualized and encapsulated as manufacturing cloud services (MCSs), before being registered and published into the CMfg platform. Through the network, MCSs can be easily discovered and accessed by customers. At the same time, the intelligent and efficient management and operations of MCSs have driven the full sharing of resources and the deep collaboration among enterprises. For the allied enterprises, the shared resources, knowledge and technologies can make up their technological limitations so that the barriers to take part in complex projects are lowered.

Applying the concept of CMfg into the injection molding industry is promising not only because the problems that most SMEs are faced with can be addressed, but it also brings significant contributions. Firstly, the established CMfg platform can realize the full-scale sharing, free circulation, efficient integration and optimal configuration of IMMs. Secondly, CMfg is reshaping a new business pattern for the injection molding industry. The injection molding machine (IMM) is offered as an IMM cloud service (IMMCS). Relying on the platform, the circulation of market information becomes more transparent and active, which enables enterprises to seize more business opportunities and earn more profits. Finally, via the CMfg platform, EIS providers rent their decision tools to the injection molding machine provider (IMMP) at a low price. Meanwhile, with more and more IMMPs hire their decision tools, they can make sustainable profits. Therefore, a win-win situation is reached and the problems above can be effectively tackled. However, few existing works have discussed the implementation of CMfg in the injection molding industry. Considering the specific features of the injection molding process, the following challenges need to be tackled.

1. During the service modeling process, how to describe the manufacturing capabilities of IMMCSs based on the technical parameters of the injection molding, especially to reveal the dynamic capabilities?
2. In the service selection process, if there are multiple IMMCSs for the customer to choose from, how to find one that satisfies the customized requirements best in a real-time environment?
3. In practice, task orders are continuously published into the platform. Then, they are assigned to the corresponding IMMs. For injection molding enterprises, how to generate production schedules for the scheduled task orders as well as the newly arrived ones?

In this work, we only study the injection molding operation which is a typical single-stage manufacturing process. Several key technical parameters that embody the capabilities of IMMs are provided. Through the deployed RFID devices and other sensors in the shop floor, the machine's real-time data during the molding process are captured. After the formal description of the IMM, its machining capability and related manufacturing information can be abstracted as a virtual machining service. Based on the service-oriented technologies (e.g. web service and ontology), the virtual machining services are encapsulated into IMMCSs. Then, we register and publish them into a specialized cloud service platform for the injection molding enterprises (IME-CMfgSP). Therefore, IMMCSs can be dynamically accessed, invoked and configured. The configuration mode studied here is task-driven, which is realized by a task-driven proactive service discovery method. It turns around the situation that the IMMCSs are passively found by demanders. For suppliers, they are more proactive to find tasks with the help of the free flow of market information. From another perspective, demanders play leading roles in the configuration process. Their task orders are allocated to the IMMCSs that satisfy their customized requirements best through a technique for order preference by similarity to ideal solution (TOPSIS) based evaluation method. Before tasks are performed, a production scheduling plan should be firstly generated. However, they did not consider the newly arrived orders. To deal with this scheduling problem in the CMfg environment, the real-time production data and the dynamic demand information are collected as the decision inputs to reflect the dynamic characters of the configuration process. The aim of IMMS in this work is to minimize the total weighted tardiness of orders assigned on the IMMCSs.

The remainder of this paper is organized as follows. The "Literature review" section reviews the works related to this paper. “Overall architecture of IMMCS configuration” describes an overall architecture of the configuration process for IMMCS. The servitization framework of IMMCS and an information model of IMM are presented in “Servitization of IMM” section. In “Optimal configuration method for IMMCS” section, the entire configuration process is discussed, including service discovery, service evaluation and selection, task assignment and scheduling. The “Numerical simulation” section gives a numerical simulation to illustrate the proposed methods. Finally, in “Conclusion and future work”, conclusions are summarized and the future work is introduced.

Literature review

Related works are categorized into the following three parts that aim to address the challenges above. They are CMfg and cloud service modeling, cloud service optimal selection and the scheduling strategy in a real-time scenario.

CMfg and cloud service modeling

As an emerging manufacturing paradigm, CMfg is expected to realize an effective approach to perform networked, distributed and collaborative manufacturing business (Adamson et al. 2015). So far, plenty of fundamental studies on CMfg have been done, including the architecture (Škulj et al. 2015), concept, core technologies (Kubler et al. 2016) and applications. Meanwhile, the state-of-the-art and future trends have been discussed (Wu et al. 2013; He and Da Xu 2015; Buckholtz et al. 2015).

CMfg is the manufacturing version of cloud computing and “borrows” the concept of Platform as a Service (PaaS) to build a CMfg platform, which enables ubiquitous, convenient and on-demand network access to a shared pool of manufacturing resources (Xu 2012). Thus various manufacturing resources like 3D printing (Mai et al. 2016), CAD/CAE (Zissis et al. 2016), etc. can be timely shared and efficiently utilized. Here, several cloud service platforms have been proposed to realize the effective integration and collaborative manufacturing. Huang et al. (2013) addressed the problems faced by SMEs and analyzed the existing solutions and disadvantages. For dealing with this situation, they proposed a SME-oriented CMfg service platform (SME-CMfgSP) and gave an application prototype. Valilai and Houshmand (2014) proposed a collaborative and integrated platform named STRATUS Cloud based on the CMfg paradigm for today’s global manufacturing to enable optimization systems in distributed manufacturing enterprises. Khalfallah et al. (2014) addressed the challenges of interoperability in collaborative production development environment. To ensure end-to-end interoperability between collaborative partners, they presented a cloud-based collaborative platform based on a two-phase semantic data mediation approach. Modekurthy et al. (2015) presented a cloud-based service broker system for cloud additive manufacturing (AM) service providers and designed a cloud AM service selection method enabling consumers to find satisfactory cloud AM services.

Manufacturing resources and capabilities are treasures to suppliers and to reveal their core competitiveness. In the service modeling process, the real-time manufacturing information that embodies the dynamic capabilities can be captured by IoT technologies, such as RFID (Barenji et al. 2014; Zhang et al. 2015b; Oner et al. 2016). Based on the multi-source manufacturing information, Zhang et al. (2015a) presented a capability model of the manufacturing machine. And web ontology language for service (OWL-S) is used to describe the ontology of the manufacturing service. Then services are published through Manufacturing Service Universal Description, Discovery, and Integration (MS-UDDI). Xu et al. (2015) proposed a knowledge description model of manufacturing equipment, including the static and dynamic information. And to reflect the dynamic characteristics, they used dynamic description logistic (DDL) to describe the dynamic model of manufacturing equipment capability. Ameri et al. (2012) presented a methodology to develop manufacturing capability ontologies and decomposed the capability model into five levels. Simple Knowledge Organization System (SKOS) is employed as the thesaurus to identify the concepts of the ontologies. Additionally, manufacturing service description language (MSDL) is used to describe the manufacturing service at a semantic level in the digital manufacturing market (Ameri and Patil 2012). Luo et al. (2013) established a multidimensional information model of manufacturing capability and adopted OWL 2 as the description carrier of the manufacturing capability description framework (MCDL). To promote reuse of the product design and manufacturing process knowledge in design and process failure modes, Chhim et al. (2017) employed ontologies and Semantic Web technologies to encode knowledge into ontologies. Then these ontologies are formalized via Protégé and uploaded onto the web. Quintanilla et al. (2016) proposed a manufacturing service modeling methodology from four perspectives, including service type, specification, profile and implementation, to support the design of customizable production-process specifications.

Cloud service optimal selection

In the CMfg platform, there are abundant cloud services that provide similar functionalities but differ in quality of service (QoS), how to choose services to satisfy specific requirements of manufacturing tasks has become a key issue to implement CMfg. In the practical production process, QoS criteria such as the execution time, cost, reliability, and so on are widely concerned and are usually considered simultaneously. Thus, the service optimal selection process can be viewed as a multi-criteria decision making (MCDM) (Tapoglou and Mehnen 2016) problem and the overall QoS level is comprehensively evaluated. Lartigau et al. (2015) proposed a method based on QoS evaluation and analyzed the geo-perspective correlation among cloud services. To optimize this method, they developed an adapted Artificial Bee Colony (ABC) algorithm with an enhanced initialization. Tao et al. (2013) investigated the service composition optimal selection (SCOS) problem with multiple objectives and constraints, and developed a parallel intelligent algorithm. Seghir and Khababa (2016) addressed the QoS-aware cloud service composition problem and proposed a hybrid genetic algorithm (HGA) combining genetic algorithm and fruit fly optimization algorithm to solve it. Haddad et al. (2010) presented a web service composition selection algorithm considering the transactional requirements and QoS characteristics to satisfy customer's preferences. Cao et al. (2016) established a service selection and scheduling model with criteria TQCS (time, quality, cost and service) considered, including the logistics time and cost. Then, an ant colony optimization (ACO) with a selection mechanism (ACOS) is proposed for this model. Xu et al. (2016) considered three kinds of correlation between services and presented a correlation-aware QoS model of aggregation service. Then, they proposed an improved discrete bees algorithm based on Pareto (IDBA-Pareto) to solve the service aggregation optimal selection problem. Chen et al. (2016) proposed a novel QoS-aware Web service composition (QWSC) method and developed an efficient ε-dominance multi-objective evolutionary algorithm (EDMOEA) to find the Pareto optimal solutions, which supports users to select services with the tradeoff between the QoS performance and the QoS risk.

Scheduling strategy in a real-time scenario

After assigning tasks to IMMCSs, scheduling is firstly performed in the service execution process and is subject to some temporal constraints derived from the real-time production status of IMMs (e.g. availability and workload) and the changing task queues. In the CMfg system, one of the principal scheduling challenges is to schedule and assign tasks with unpredictable order flow (Lartigau et al. 2012), which makes the traditional and static scheduling approach insufficient. Therefore, dynamical optimization for multiple tasks from a global perspective should have more practical significance. Considering the dynamic order change and internal disturbances in real-life manufacturing flow lines, Barenji et al. (2016) proposed a multi-agent-based dynamic scheduling system using the Prometheus methodology and a decision-making mechanism among multiple agents was illustrated. To efficiently exploit the robots in distributed enterprises, Li et al. (2015) proposed a CMfg scheduling model in which the deployment methods for four robots are considered, and further three optimization objectives that include load, cost and processing time are formulated based on the corresponding subtask scheduling strategies. With IoT and RFID technologies, the acquisition and integration of real-time production information keep the decision makers being timely informed of any dynamics, thus corresponding adjustment actions (scheduling or rescheduling) can be quickly taken to maintain the global stability (Sokolov et al. 2016). Luo et al. (2015) developed a multi-period hierarchical scheduling (MPHS) mechanism for the hybrid flow shop with the implementation of RFID technologies. In upper level shop floor scheduling, the workload for all stages is balanced and the due dates of jobs are considered. In the lower level, the stage scheduling is modeled as a parallel machine scheduling problem to minimize the makespan. Chongwatpol and Sharda (2013) presented a RFID-based traceability system to track work in processes (WIPs), parts, components and raw materials in shop-floor operations. Then the proposed information visibility-based scheduling rule was validated to improve production scheduling such as reduction in cycle times or improved capacity utilization. Based on the agent and cyber-physical system, Zhang et al. (2016b) proposed a framework of self-organizing and self-adaptive intelligent shop floor. Under this framework, the designed self-adaptive model can timely identify the exceptions during manufacturing progress, and regenerate a new schedule to minimize the total weighted delay time.

Overall architecture of IMMCS configuration

The overall architecture of IMMCS configuration process is presented in Fig. 1, where three types of cloud users are involved in the CMfg system, i.e. the CMfg platform, service providers and service demanders. Service providers play fundamental supporting roles in CMfg. They make great contributions to full-scale sharing, circulation and on-demand use of manufacturing resources. Here, we consider two kinds of service providers including IMMP and logistics provider (LP), as the production and logistics are two critical segments in the real-life supply chain networks (Wang et al. 2016a). Distributed resources (IMMs and vehicles) are virtualized and encapsulated as cloud services, i.e. IMMCSs and logistics cloud services (LCSs), then they are published into the cloud service pool and aggregate into the manufacturing cloud and the logistics cloud respectively. The advanced information and computing technologies help to realize the precise archiving and efficient management of cloud services. The injection molding task order (IMTO) is formalized and represented by the standard template and is published into the demand cloud. The application service bases are a

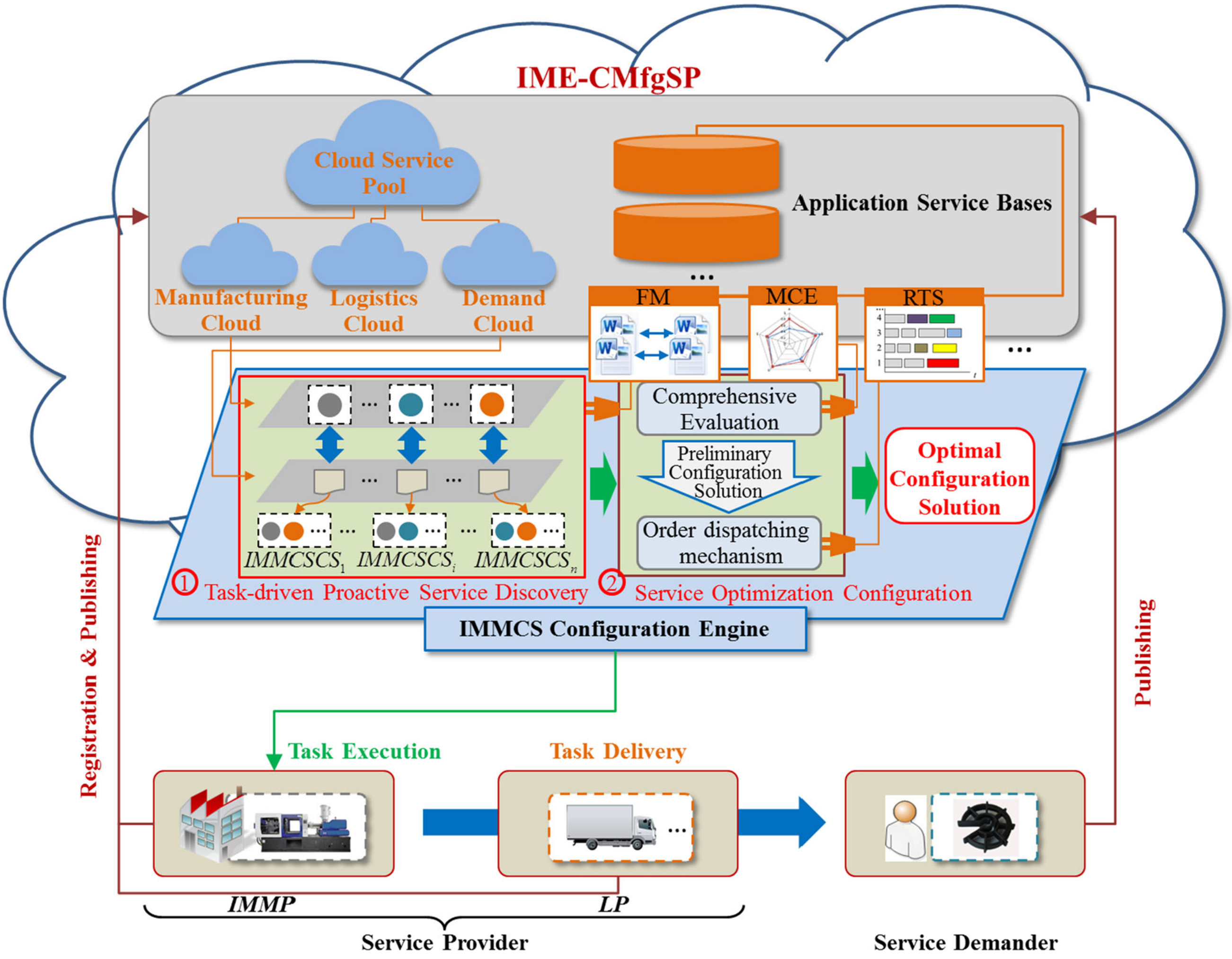

IME-CMfgSP
Cloud Service Pool
Application Service Bases
Manufacturing Cloud
Logistics Cloud
Demand Cloud
FM
MCE
RTS
Comprehensive Evaluation
Preliminary Configuration Solution
Order dispatching mechanism
Optimal Configuration Solution
IMMCSCS1
IMMCSCSi
IMMCSCSn
① Task-driven Proactive Service Discovery
② Service Optimization Configuration
IMMCS Configuration Engine
Registration & Publishing
Publishing
Task Execution
Task Delivery
IMMP
LP
Service Provider
Service Demander

Fig. 1 Overall architecture of IMMCS configuration

kind of service systems (Wang et al. 2016b). In the base, the value-added manufacturing information is wrapped into application services according to the template offered by EIS providers. Application services can be easily invoked and rented by cloud users in a plug-and-play manner. With the integration of multi-source information of resources and the application services for technical support, IME-CMfgSP can promote the operation, coordination, optimization of IMMCSs, enhancing resource utilization and dealing with variant manufacturing tasks.

This work focuses on the IMMCS configuration in the production process, the transportation part is not considered. An IMMCS configuration engine is built to optimize IMM configuration and to satisfy cloud users' needs better, which is of great importance to the implementation of CMfg. This engine comprises two core parts, they are the task-driven proactive service discovery and service optimization configuration.

The task-driven proactive service discovery is performed in the first place with functional match (FM) invoked. By FM, the competent services are pooled into IMMCS candidate set (IMMCSCS) for each task order. In the service optimization configuration, the multi-criteria evaluation (MCE) based on the classical TOPSIS method is invoked in the comprehensive evaluation process. With some customized criteria, the presented evaluation method helps the task orders to find their satisfactory solutions. Surely, each task order can only be executed by one IMMCS. For the selected IMMCSs, each of them may accept one or more orders. And the set of new orders assigned to the IMMCS is defined as the preliminary configuration solution. Real-time scheduling (RTS) can be used by the cloud platform or any individual IMMP. With RTS, we designed an order dispatching mechanism so that the platform can dynamically dispatch newly arrived orders to IMMCSs according to their real-time status, and then the processing sequence of orders on each IMM is determined. In this case, the RTS is viewed as a parallel-machine version. Similar for any IMMP, the provider utilizes RTS to find an optimal schedule for new orders or to insert new

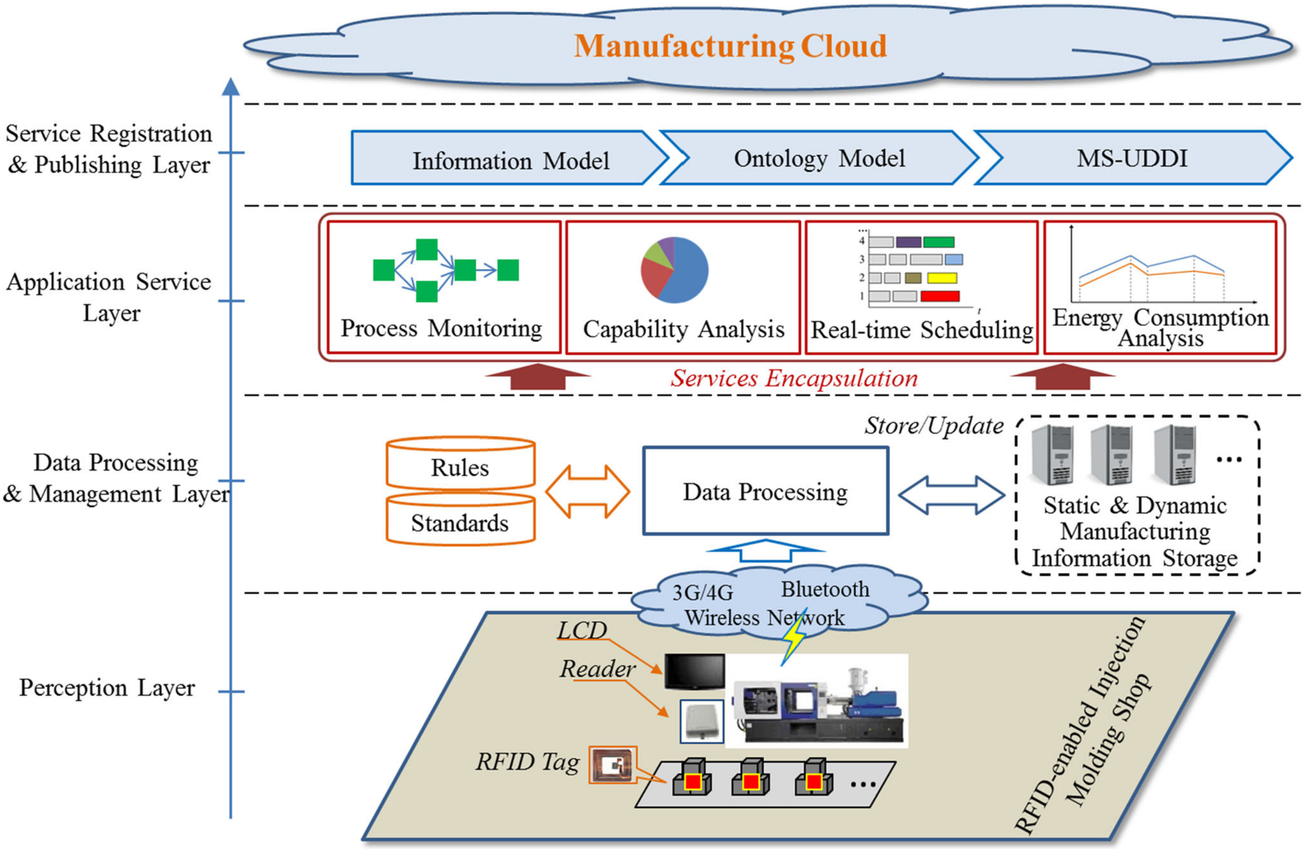


Fig. 2 Servitization framework of IMMCS

orders into the originally scheduled queue, which means to reschedule the new orders as well as the remaining unexecuted orders in the queue if necessary. So, this RTS is defined as the single-machine version and is further studied in this work. With the intelligent algorithm employed, an optimal schedule is achieved, i.e. the optimal configuration solution. Then, the CMfg platform transmits the ultimate configuration results to the corresponding IMMPs. After the task orders are completed, they are delivered to service demanders by LCSs (Zhang et al. 2016a) and the transactions are finished.

Servitization of IMM

Servitization framework of IMMCS

The servitization of IMMs plays a significant part in CMfg. As IMMs are registered and published into the CMfg platform by providers, it facilitates the open, flexible, on-demand use of resources. Furthermore, it makes much easier for providers to seek for business opportunities and complete more transactions. In addition, from the service demander's angle, the resource query and invocation of IMMCSs become readily available and easy-to-use. The whole servitization process is shown in Fig. 2, and it can be divided into four layers, namely the perception layer, the data processing & management layer, the application service layer, and the service registration & publishing layer.

Perception layer

At the perception layer, by applying IoT and RFID technologies to the injection molding shop, and with some RFID devices and communication modules deployed, a RFID-enable injection molding shop (RIMS) is created, which provides support for the real-time and seamless connectivity and interaction between the production frontline and the CMfg platform. In manufacturing activities, the multi-source production information can be dynamically sensed and captured. For IMMs, different kinds of sensors are embedded to collect the parameters of the working condition, such as the pressure sensor, the temperature sensor and the linear displacement sensor, which helps to reveal the real-time molding process. Meanwhile, materials, WIP items and products are attached with RFID tags, and the RFID readers are installed at machines and buffers, so that the dynamic manufacturing information of task orders can be identified. With the support of networks like 3G/4G networks, Bluetooth, wireless networks, etc., the lossless and high-speed transmission of real-time data is realized.

Data processing and management layer

At this layer, all the raw data acquired from the lower layer are processed in the data processing component. The whole processing cycle of data proceeds from the definition, optional filtering, aggregation and fusion, and finally based on rules and standards, the processed data are converted into value-added manufacturing information. Two kinds of manufacturing information are stored and updated in the information storage, including the static description information derived from the inherent attributes of IMM and production history, and the dynamically sensed information based on the captured real-time production data.

Application service layer

At the application service layer, based on the application template rented from EIS providers, IMMPs encapsulate both static and dynamic manufacturing information into application services, to provide diversified, customizable, real-time applications throughout the product lifecycle (Zhang et al. 2016c). Besides aforementioned FM and RTS, typical application services available for cloud users include process monitoring (PM), capability analysis (CA), energy consumption analysis (ECA) and multi-criteria evaluation (MCE).

- FM: FM is responsible for performing a function-perspective match between task orders and cloud services by adopting the semantic match method (Paolucci et al. 2002). The semantic matching degree is rated and defined as four levels, i.e. *Exact*, *Plug in*, *Subsume* and *Fail*. During the matching procedure, the matching degrees between capability attributes and functional requirements are measured.
- RTS: RTS is the core of IMMCS configuration engine. After assigning dynamically arrived task orders to IMMs, then RTS is invoked to determine the process sequences. Different versions of RTS are available for different cloud users under different circumstances. The details of RTS are described in “RTS” section.
- PM: PM reveals the real-time molding progress by tracing the evolution of key technical parameters. When a warning deviation happens, amendments should be timely carried out or even interrupts the molding operation if necessary to avoid deteriorating the products.
- CA: CA is often invoked along with RTS. It quantifies the dynamic capability of the machine by the index load rate. And with the load rate grows, the capability weakens.
- ECA: ECA measures all kinds of energy consumption in related production activities and generates analysis reports. Energy consumptions in molding process are like water, heat, and electricity etc., and electricity is of primary concern in this work.
- MCE: MCE assesses the configuration solutions through the multiple defined criteria by employing the TOPSIS method. A decision-making system is built based on MCE, which enables the cloud users to find satisfactory solutions.

Through specific application interfaces, cloud users can inquire and invoke application services in a plug-and-play manner. Furthermore, the collaboration among application services forms an integrated function block that promotes the flexibility and adaptability to meet the varying task requirements.

Service registration and publishing layer

The service registration and publishing layer enables resources to seamlessly access the CMfg platform. The entire process consists of three steps as listed below.

- Information model of IMM: As a core element of this layer, it formally describes both static and dynamic sensed manufacturing information of machines, including basic attributes, capability attributes and so on. It can transform the physical resources into virtual ones, revealing the machining capabilities and real-time machining status and so on.
- Ontology model of IMM: In order to achieve the high-quality, efficient service discovery and service match, ontology and semantic web technologies are employed to construct the ontology model of IMM. Meanwhile, the widely used ontology description language i.e. OWL-S is chosen to semantically describe the ontology model of IMMCSs.
- Service registration and publishing mechanism based on the Manufacturing Service Universal Description, Discovery, and Integration (MS-UDDI): Based on the UDDI technology, the proposed framework of MS-UDDI consists of three sub-modules, i.e. the registration module, the publishing module and the search module. In registration module, through the OWL-S/UDDI matchmaker (Srinivasan et al. 2004), the data structure of ontology model described by OWL-S is transformed to that supported in UDDI, so that services can be registered in MS-UDDI. In publishing module, various cloud services are published into CMfg platform, and then corresponding cloud service pools are formed. In search module, service demanders can inquire cloud services according to their needs.

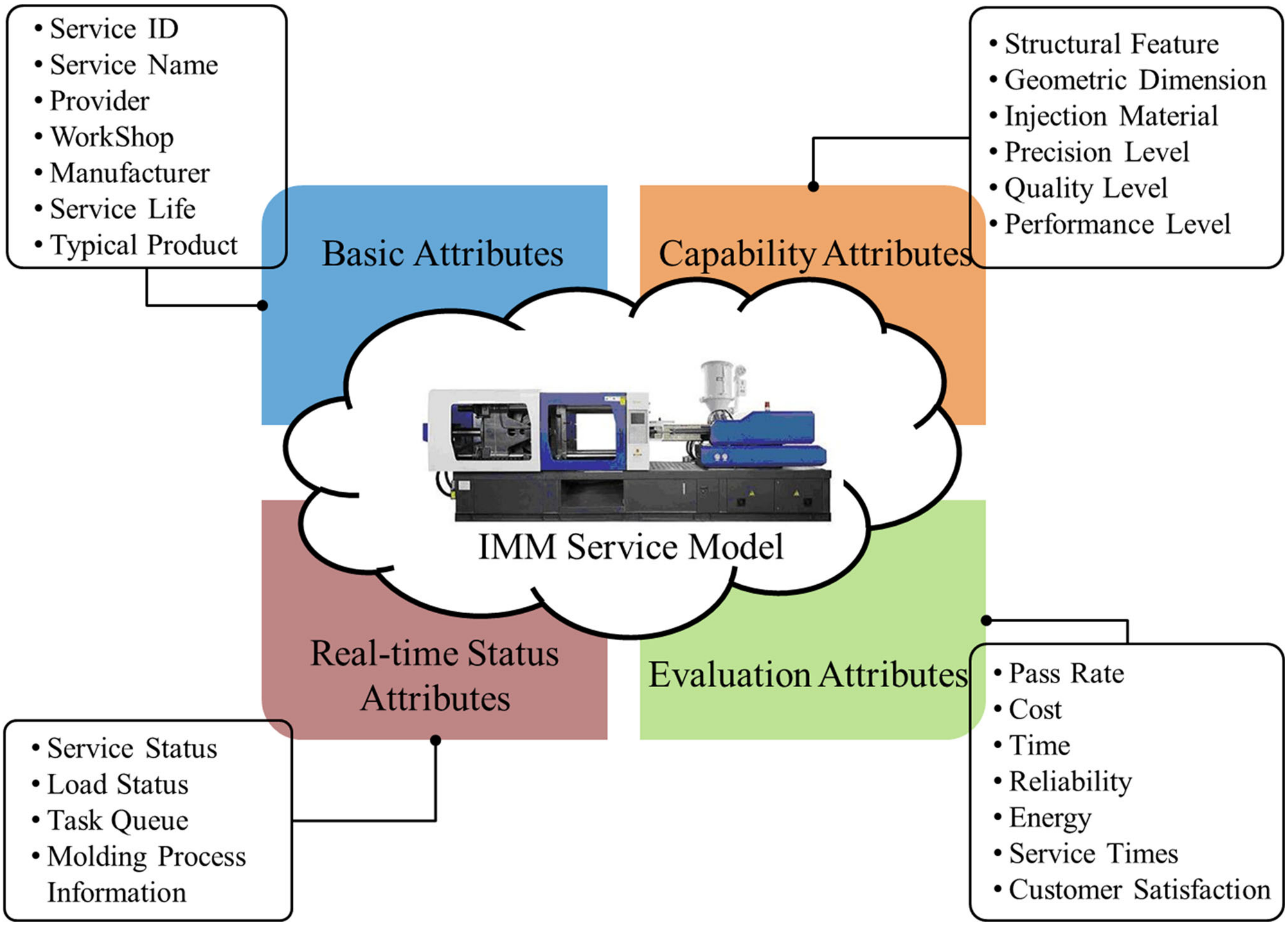


Fig. 3 IMM service model

Information model of IMM

Constructing the information model of IMM lays a solid foundation for the service registration and publishing processes. It is of great significance to the subsequent service discovery and intelligent match. In this section, comprehensive manufacturing information of the IMM is investigated, including the static manufacturing attributes and the real-time production information. In practice, heterogeneous and massive data collected from the machine's inherent attributes and related manufacturing activities increases the redundancy and inoperability of service modeling. Therefore, some representative attributes are extracted and integrated to build the IMM service model. As shown in Fig. 3, the IMM service model consists of four attributes. They are basic attributes, capability attributes, real-time status attributes and evaluation attributes respectively. The IMM service model is defined as:

$$\begin{aligned} IMMCS = (&IMMBasicAttr, IMMCapabilityAttr, \\ &IMMStatusAttr, IMMEvaluationAttr). \end{aligned}$$

Basic attribute

The basic attribute of IMM gives an overall description of the machine's inherent attributes, including service ID, service name, provider, workshop, manufacturer, service life and typical product. It is defined as:

$$\begin{aligned} IMMBasicAttr = (&IMMID, IMMName, IMMProvider, \\ &IMMWorkshop, IMMManufacturer, \\ &IMMLife, IMMProduct). \end{aligned}$$

Here, some definitions require further explanation. *IMMWorkshop* indicates the location information of IMM. *IMMProduct* describes the product output from the IMM, which can facilitate a quick match between tasks and IMMCSs.

Capability attribute

As a core component of the service model, capability attributes are pivotal criteria to measure whether IMMCS can meet the functional requirements of injection molding tasks, and to provide technical support for FM in the proactive service discovery. In the molding process, the capability attributes are embodied by the functional information which has decisive effects on the quality of products, including equipment inherent parameters, mold design information and molding technical parameters. In practical operations, to shield the heterogeneous parameters and massive data and for the convenience of executing the intelligent match, six crucial capability attributes are defined. They are the structural feature, the geometric dimension, the injection material, the precision level, the quality level and the performance level. It is defined as:

$$\begin{aligned} \mathit{IMMCapabilityAttr} = (&\mathit{IMMFeature}, \mathit{IMMGeoDimension}, \\ &\mathit{IMMMaterial}, \mathit{IMMPrecision}, \\ &\mathit{IMMQuality}, \mathit{IMMPreformance}). \end{aligned}$$

IMMFeature describes the structural characteristics, including shape and wall thickness of the product. It depends on the cavity feature of the injection mold. *IMMGeoDimension* refers to the overall size of the product. Meanwhile, it is the pivotal factor to choose injection volume, clamp force and mold size. *IMMMaterial* denotes the type of injected plastic, commonly including PP, PE, PVC, PS, ABS, etc. It is of great concern to choose the appropriate material, as different materials differ a lot in properties, affecting the molding quality and performance of products. *IMMPrecision* indicates the attainable precision level of product and is dependent on the mold structure, injection pressure, the precision of equipment and so on. *IMMQuality* reflects the injection molding quality. And the common quality flaws are like the bubble, shrinkage, warping, deformation, etc. Molding quality is usually influenced by injection rate, holding pressure, mold temperature, dwell time, etc. *IMMPerformance* mainly refers to the mechanical property of the product, which depends on the application context of the product. In the molding process, it is mainly affected by the material temperature and injection rate.

Real-time status attribute

Accurate perception of the real-time production data makes the entire molding process more intelligent, transparent and controllable. Furthermore, it makes significant contributions to service monitoring, service scheduling. Real-time status attributes contain service status, load status, task queue and molding process information. It is defined as:

$$\begin{aligned} \mathit{IMMStatusAttr} = (&\mathit{IMMStatus}, \mathit{IMMLoadStatus}, \\ &\mathit{TaskQueue}, \mathit{MoldingProcessInfo}). \end{aligned}$$

IMMStatus refers to the working condition of IMM, including running, idle and maintenance. *IMMLoadStatus* indicates the load condition of the equipment. *TaskQueue* is the sequence information of task orders assigned on the IMM, including orders under process and others waiting for processing. When new task orders arrive, *TaskQueue* should be timely updated. *MoldingProcessInfo* describes the real-time status values of vital technical parameters like pressure, temperature and so on, to monitor, control and record the current molding process.

Evaluation attribute

The evaluation attribute is of great importance in assessing machining capability of IMM and provides support for constructing evaluation system in service configuration process as well. To realize a comprehensive assessment of IMMCS, evaluation indicators are defined and selected from the production records and the feedbacks from customers. Typical criteria like pass rate, cost, time, reliability, energy, service times, and customer satisfaction (CS) are included. It is defined as:

$$\begin{aligned} \mathit{IMMEvaluationAttr} = (&\mathit{PassRate}, \mathit{Cost}, \mathit{Time}, \\ &\mathit{Reliability}, \mathit{Energy}, \mathit{STimes}, \mathit{CS}). \end{aligned}$$

Reliability refers to the execution reliability of IMMCS. *Energy* means the energy consumption (e.g. electricity) in the related molding process. *STimes* indicates the invocation times of IMMCS. The more times service is invoked, the more convincing evaluation attributes are.

Optimal configuration method for IMMCS

This section presents an IMMCS configuration engine that focuses on how to assign orders to IMMCSs in a dynamic scenario. As shown in Fig. 4, the configuration process contains two parts, including task-driven proactive service discovery and service optimization configuration. In the entire configuration process, the IMMP invokes application services in a plug-and-play manner according to the dynamically updated manufacturing information. With the feedback of the configuration results and service execution information, the closed-loop of service discovery, service evaluation and selection, task scheduling and execution is built in a real-time environment.

Let $\mathit{IMTOS} = \{\mathit{IMTO}_i;\ i = 1, 2, 3, \ldots, n\}$ denote the set of newly arrived orders of the lth batch, where IMTO_i represents the ith order; and define the variable At^l as the arrival time. At time At^l, there are m service providers being operative in the CMfg platform, and each provider owns one IMM. Let $\mathit{IMMPS} = \{\mathit{IMMP}_j;\ j = 1, 2, 3, \ldots, m\}$ denote the set of providers, where IMMP_j is the jth provider, and its IMM is denoted as IMMCS_j accordingly.

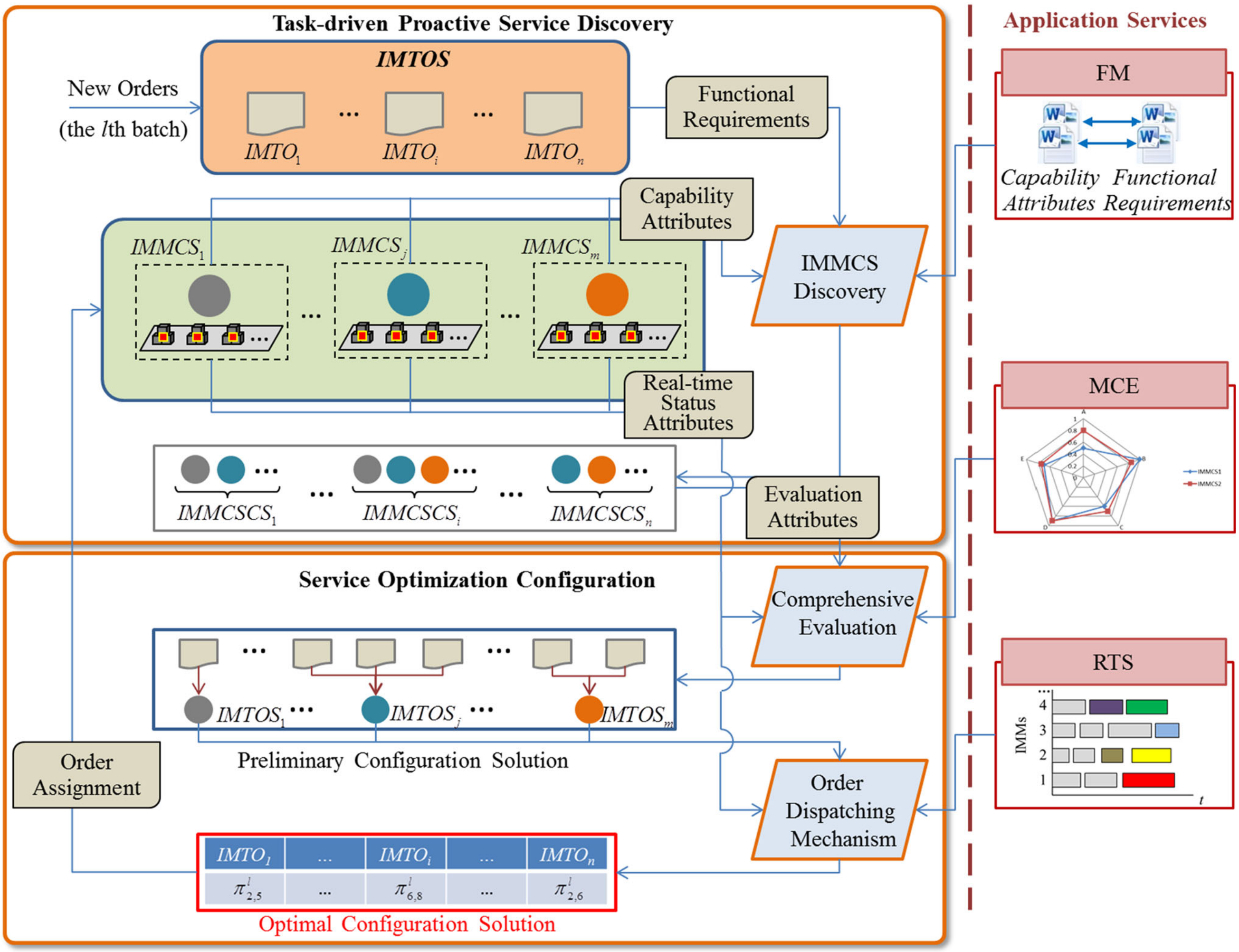


Fig. 4 Working logic of the IMMCS configuration engine

Task-driven proactive service discovery

Orders may arrive in the cloud platform at any moment, which requires rapid responses from the IMMPs. For IMMPs, they take the initiative to request for processing orders. With the support of FM, IMMCSs are filtered according to the task requirements, and then the competent ones are quickly found. This procedure is called IMMCS discovery.

Description model of the manufacturing task

Above all, to realize the fast and accurate match between orders and IMMCSs, a detailed description of the order is imperative to give. In Fig. 5, the description model of IMTO contains the basic information and the functional requirement information. It is defined as:

$$IMTO = (TaskBasicInfo, TaskFunRequInfo).$$

TaskBasicInfo refers to the basic information of order. It includes order ID, customer information, task name, task type, due date and batch. Here, we assume that the order only contains one task, which means the batch of the order is one. *TaskFunRequInfo* describes the functional requirement information of the task, including the structural feature, the geometric dimension, the material, the precision, the molding quality and the performance.

IMMCS discovery

By FM, the semantic match between capability attributes of IMMs and functional requirements of task orders is executed. And the matching degree for each pair of attributes is measured as shown in Eq. (1).

$$\begin{aligned}&Match(IMTFeature, IMMFeature)\\&\quad\wedge Match(IMTGeoDimension, IMMGeoDimension)\\&\quad\wedge Match(IMTMaterial, IMMMaterial)\end{aligned}$$

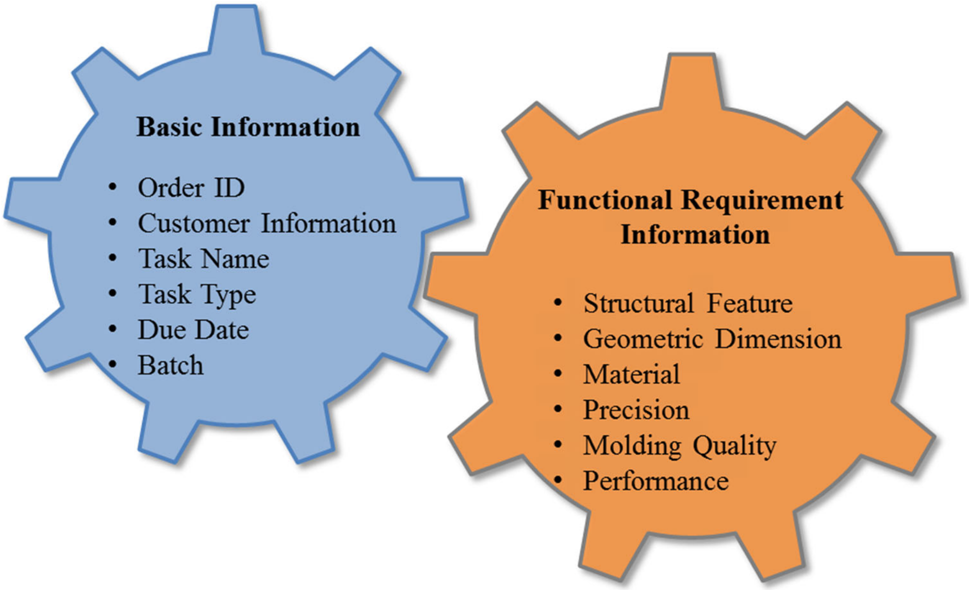


Fig. 5 Description model of IMTO

$$\wedge\, Match(IMTPrecision, IMMPrecision)$$
$$\wedge\, Match(IMTQuality, IMMQuality)$$
$$\wedge\, Match(IMTPerformance, IMMPerformance) = Exact \tag{1}$$

Only if all the matching degrees reach *Exact*, the IMMCS can fully satisfy the task requirements. The competent IMMCS is viewed as a candidate service and is pooled into corresponding IMMCSCS afterward. Let $IMMCSCS_i = \{IMMCS_i^p; p = 1, 2, \ldots, m_c^i\}$ be the candidate set for $IMTO_i$, where m_c^i is the total number of candidate services and $IMMCS_i^p$ is the pth candidate.

Comprehensive evaluation

For the service demanders, with multiple candidate IMMCSs to choose, invoking MCE based on TOPSIS method can achieve a comprehensive evaluation of services. Here, we consider the *IMMStatusAttr* and *IMMEvaluationAttr* of the IMMCS and choose five typical criteria, i.e. the minimum cost, the greatest priority, the highest reliability, the lowest energy consumption and the maximum customer satisfaction to evaluate the candidates.

- Cost (C): C is the execution cost of an IMMCS includes all the costs of the production operations and the associated materials and operators.
- Priority (Z): Z is the priority of an order on the IMM is measured by the operation slack (OSL) (Haupt 1989). Here, OSL is modified when considering the real-time queue information, as shown in Eq. (2).

$$OSL = d_i - s_{i,j} - p_{i,j} \tag{2}$$

where d_i is the due date of $IMTO_i$; $s_{i,j}$ is the starting time and $p_{i,j}$ is the processing time on $IMMCS_j$. The smaller the value of OSL is, the greater Z is. Empirically, the order with greater priority is more likely to be executed in the early stage, thus reducing the possibility of being delayed resulted from the preemption of the subsequent orders with greater priorities.

- Reliability (R): R is the execution reliability of the IMMCS, which can be quantified by the probability of working functionality in the production activities.
- Energy consumption (EC): EC reflects the energy consumption (e.g. electricity) of the IMMCS in the whole injection molding process.
- Customer satisfaction (CS): CS is the subjective rating of the IMMCS from the service demanders after completing the task orders; Besides the objective evaluation of the product quality, credit, cooperation and so on in the transactions are drawing common concern from the customers.

The TOPSIS method is one of the most widely used tools for MCDM (Dymova et al. 2013), and it is introduced in the service evaluation method. This MCDM problem for each task order is based on the aforementioned five criteria and their respective candidate IMMCSs. Let $D[x_{pq}]_{m_c^i \times e}$ be the decision matrix, where x_{pq} represents the value of qth criterion of pth candidate with$1 \leq p \leq m_c^i$ and $1 \leq q \leq e = 5$. And vector $W = (w_1, w_2, w_3, w_4, w_5)$ denotes the weights of criteria satisfying $\sum_{q=1}^{5} w_q = 1$. It also reflects the QoS preferences of service demanders (Choi and Jeong 2014). The entire evaluation procedure consists of the following steps:

1. Normalize the decision matrix:

$$r_{pq} = \frac{x_{pq}}{\sqrt{\sum_{p=1}^{m_c^i} x_{pq}^2}}, \ \ p{=}1, 2, \ldots, m_c^i, \ \ q{=}1, 2, 3, 4, 5 \tag{3}$$

Considering the weights of criteria, the weighted and normalized decision matrix is obtained according to Eq. (3).

$$v_{pq} = w_q r_{pq}, \quad p = 1, 2, \ldots, m_c^i, \quad q = 1, 2, 3, 4, 5 \tag{4}$$

2. Determine the positive and negative ideal solutions:

$$\begin{aligned} A^+ &= \{v_1^+, v_2^+, v_3^+, v_4^+, v_5^+\} \\ &= \{(\max_p v_{pq} \,|q \in I_b), (\min_p v_{pq} \,|q \in I_c)\} \end{aligned} \tag{5}$$

$$\begin{aligned} A^- &= \{v_1^-, v_2^-, v_3^-, v_4^-, v_5^-\} \\ &= \left\{(\min_p v_{pq} \,|q \in I_b), (\max_p v_{pq} \,|q \in I_c)\right\} \end{aligned} \tag{6}$$

where I_b is the set of benefit-oriented criteria (i.e. *R* and *CS*), and I_c is the set of cost-oriented criteria (i.e. *C*, *Z* and *EC*).

3. Calculate the distances of each candidate from the positive and negative ideal solutions: Two Euclidean distances for each candidate are calculated respectively as follows:

$$S_p^+ = \sqrt{\sum_{q=1}^{5} (v_{pq} - v_q^+)^2}, \quad p = 1, 2, 3, \ldots, m_c^i \tag{7}$$

$$S_p^- = \sqrt{\sum_{q=1}^{5} (v_{pq} - v_q^-)^2}, \quad p = 1, 2, 3, \ldots, m_c^i \tag{8}$$

4. Calculate the relative closeness to the ideal candidates:

$$RC_p = \frac{S_p^-}{S_p^+ + S_p^-}, \quad p = 1, 2, 3, \ldots, m_c^i \tag{9}$$

Rank the candidates according to their relative closeness to the ideal candidate in descending order, the bigger RC_p is, the better the candidate $IMMCS_i^p$ is.

For each service demander, they found satisfactory solutions after the comprehensive evaluation. Here, let $IMMCS_i^*$ be the optimal solution for $IMTO_i$. In practice, there are usually one or more orders being assigned to the same IMMCS. Therefore, let $IMTOS_j = \{IMTO_{j,k}; k = 1, 2, \ldots, K_j\}$ denote the set of orders assigned to the $IMMCS_j$, where K_j is the total number satisfying $n = \sum_{j=1}^{m} K_j$ and $IMTO_{j,k}$ is the *k*th order. With all the orders assigned, $PCS = \{IMTOS_j; j = 1, 2, \ldots, m\}$ is defined as the preliminary configuration solution.

Order dispatching mechanism

RTS

After the new orders are accepted, relying on the transparent and accessible production process in the RIMS, the real-time status data are input to the order dispatching module. In this work, we focus on the single-machine problem in a real-time environment, which means each IMMCS invokes RTS to determine the permutation of the orders to meet the specific scheduling objective with the real-time production information and the order information considered. In reality, missing an order's due date is a common production problem and usually results in a penalty for delay. Therefore, the scheduling objective is to minimize the total weighted tardiness of orders assigned to the IMMCS, which is referred as the single machine total weighted tardiness problem. Here, some notations used in the scheduling model are explained firstly.

- π: the sequence of orders on machine and $\pi_{j,k}$ denotes the *k*th order in a schedule on $IMMCS_j$;
- $IMTOS_j^{l-1} = \{IMTO_{j,1}, IMTO_{j,2}, \ldots, IMTO_{j,K_j^{l-1}}\}$: the set of unexecuted orders remained in the $(l-1)$th batch on $IMMCS_j$ at At^l, and these K_j^{l-1} orders have been scheduled optimally and π^{l-1} is the optimal schedule with no idle time between the adjacent orders, and $IMTO_{j,0}$ denotes the order currently under processing at At^l;
- $IMTOS_j = \{IMTO_{j,K_j^{l-1}+1}, IMTO_{j,K_j^{l-1}+2}, \ldots, IMTO_{j,K_j^{l-1}+K_j}\}$: the set of new orders;
- $IMTOS_j^l = IMTOS_j^{l-1} \cup IMTOS_j = \{IMTO_{j,1}, IMTO_{j,2}, \ldots, IMTO_{j,K_j^l}\}$: the set of new orders and unexecuted orders, where $K_j^l = K_j^{l-1} + K_j$;
- $s_{j,k}^l$: the starting time of *k*th order, and $s_{j,1}^1 = 0$, $s_{j,1}^l \neq 0$ according to the real-time queue information, noted that except breakdown happens, no idle time is allowed on the machine;
- $p_{j,k}^l$: the processing time of *k*th order on $IMMCS_j$, and the setup time for order is merged into the processing time;
- $c_{j,k}^l$: the completion time of *k*th order;
- $d_{j,k}^l$: the due date of *k*th order;
- $\alpha_{j,k}^l$: the tardiness penalty for *k*th order;

Fig. 6 Pseudo-code of the designed GA

```
Procedure GA
Begin
    $t$=0;
    Generate an initial population $P(t)$ of size $N_{pop}$ randomly;
    Calculate the fitness of each individual in $P(t)$;
    While (not satisfy stopping criterion) do
        $t$=$t$+1;
        Select $P(t)$ from $P(t$-1);
        Perform crossover operation based on the crossover probability $P_c$ ;
        Perform mutation operation based on the mutation probability $P_m$ ;
        Calculate the fitness of each individual in $P(t)$;
    End while
End
```

Table 1 Information of the arrived orders at time At^8

Order ID	Due date	Weight	Batch
T001	20	1	5
T002	19	6	10
T003	30	7	10
T004	16	6	2
T005	26	1	4
T006	22	2	10
T007	25	8	5
T008	13	4	15
T009	28	3	10
T010	21	7	5
T011	31	5	8
T012	17	7	10
T013	14	3	5
T014	23	1	10
T015	13	5	3

Table 2 Processing time of task orders on IMMCSs

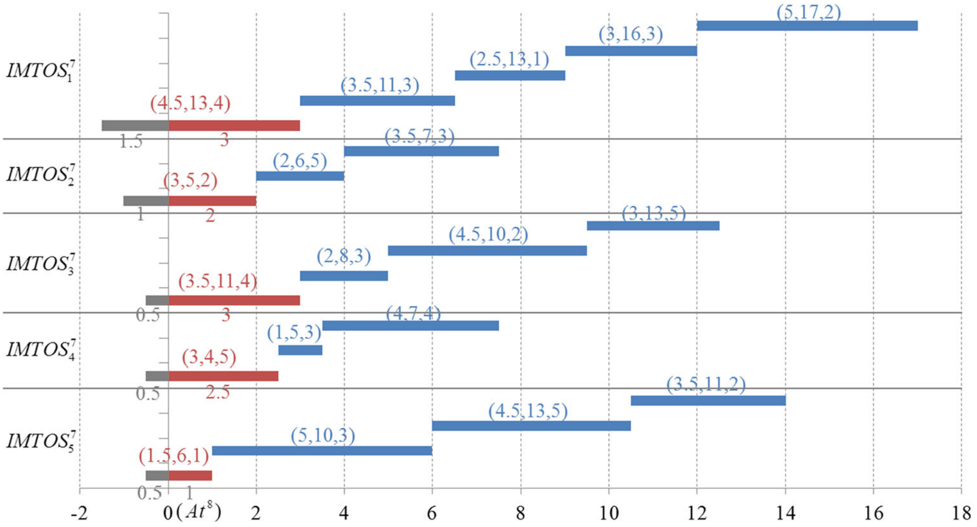
Order ID	$IMMCS_1$	$IMMCS_2$	$IMMCS_3$	$IMMCS_4$	$IMMCS_5$
T001	5	–	4.5	5	6
T002	3.5	4	–	4	4.5
T003	–	5.5	5.5	6	–
T004	3	–	3	4	3.5
T005	5	5.5	4	4.5	5
T006	–	6	6.5	5	7
T007	3.5	5	–	–	4
T008	5	5	5.5	–	6
T009	4.5	4	3.5	5	5
T010	4	–	4	4.5	–
T011	4.5	4.5	5.5	5	–
T012	6	6.5	–	5.5	7
T013	–	5	5.5	–	5.5
T014	5	–	4.5	3.5	6
T015	5.5	3.5	–	5	–

As new orders arrive randomly, at At^l, if $(At^l - c^{l-1}_{j,K^{l-1}_j}) > 0$, the IMMCS is idle until the first order in the lth batch starts to process, where c^{l-1}_{j,K^{l-1}_j} denotes the completion time of the last order in the $(l-1)$th batch. Particularly, when $(At^l - c^{l-1}_{j,K^{l-1}_j}) = 0$ the first order starts immediately after the last order finishes. In this case, only K_j orders need to be scheduled, which is defined as Problem 1(P1). If $(At^l - c^{l-1}_{j,K^{l-1}_j}) < 0$, the schedule π^{l-1} may be disrupted, and a new schedule for the remaining orders in the queue as well as the new orders, is needed (Hall and Potts 2004; Duron et al. 2009; Rahmani and Heydari 2014). Noted that At^l is considered as the initial time of the new schedule and called time 0. It is assumed that the orders that have been already processed are removed from the queue, and the partly processed order should be processed until completion. In this case, K^l_j orders of $IMTOS^l_j$ need to be scheduled, and π^l which denotes the optimal schedule should be determined. This case is defined as Problem 2 (P2). We study P1 and P2 as follows:

1. *P1* we assume that the first order in the lth batch starts processing as soon as this batch arrives, which means $s^l_{j,1} = At^l$. So the completion time of kth order is:

$$c^l_{j,k} = s^l_{j,1} + \sum_{h=1}^{k} p_{j,h},\ \ h=1, 2, \ldots, k,\ \ k=1, 2, \ldots, K_j \tag{10}$$

Fig. 7 Real-time task queues of IMMCSs



And an order's tardiness $T_{j,k}^l$ is defined as $T_{j,k}^l = \max(c_{j,k}^l - d_{j,k}^l, 0)$. The scheduling objective is:

$$\min f^l(IMMCS_j) = \min \sum_{k=1}^{K_j} \alpha_{j,k}^l \cdot T_{j,k}^l \tag{11}$$

2. *P2* when orders arrive at At^l, if an order is under processing and has not been completed yet, thus the first order in $IMTOS_j^l$ has to wait until the current order complete. We define the variable $c_{j,0}^{l-1}$ as the completion time of the current order of the $(l-1)$th batch. As no idle time between orders, we assume $s_{j,1}^l = c_{j,0}^{l-1}$, so the completion time of kth order is calculated by Eq. (12):

$$c_{j,k}^l = s_{j,1}^l + \sum_{h=1}^{k} p_{j,h}, \ h=1,2,\ldots,k, \ k=1,2,\ldots,K_j^l \tag{12}$$

The tardiness is $T_{j,k}^l = \max(c_{j,k}^l - d_{j,k}^l, 0)$ and the objective is:

$$\min f^l(IMMCS_j) = \min \sum_{k=1}^{K_j^l} \alpha_{j,k}^l \cdot T_{j,k}^l \tag{13}$$

GA-based solving method

Considering that the scheduling problem has a large solution space, we use the genetic algorithm (GA) to find an optimal or near optimal solution within a reasonable computation time. GA has been widely studied in the scheduling fields and shows promising performance. Here, pseudo-code of the designed GA is shown in Fig. 6:

We design the encoding method for constituting the chromosome to solve P1and P2 respectively. An integer-based encoding method is used and each integer represents a gene. In P1, the chromosome consists of K_j genes and the sequence of genes represents a schedule on the machine. While in P2, the chromosome has K_j^l genes. $P(t)$ represents the population at generation t, and the initial population should be denoted as $P(0)$. In this approach, Eqs. (11) and (13) are employed as the fitness function to evaluate each individual. The roulette wheel selection operator, two-point crossover operator, and swap mutation operator are adopted as genetic operations. This algorithm terminates when the stopping criterion is satisfied, which is reaching the maximum number of generations N_{gen}.

When the algorithm stops, the computational results are achieved. Therefore, for P1and P2, the permutation of orders on the machine is determined, i.e. the optimal configuration solution. We define that the parameter $\pi_{j,k}^l$ represents the kth order assigned on $IMMCS_j$ in the optimal schedule π^l. With the configuration results, the IMMCSs update their queue information and start to process orders immediately.

As in a real industrial environment, the IMMPs are faced with batches of randomly arrived orders. Therefore, their initiative and adaptability to the changing orders should be strengthened. The presented IMMCS configuration engine can provide satisfactory solutions for them, so that enhances resource utilization, production efficiency and better satisfy the service demanders at the same time.

Numerical simulation

In this section, a numerical simulation of the proposed IMMCS configuration method is given as an example by taking a batch of typical injection molding tasks that arrived in the CMfg platform. The entire IMMCS configuration process is described as follows:

Table 3 Evaluation criteria of IMMCSs

Order ID	$IMMCS_1$					$IMMCS_2$					$IMMCS_3$					$IMMCS_4$					$IMMCS_5$				
	C	*Z*	*R*	*EC*	*CS*	*C*	*Z*	*R*	*EC*	*CS*	*C*	*Z*	*R*	*EC*	*CS*	*C*	*Z*	*R*	*EC*	*CS*	*C*	*Z*	*R*	*EC*	*CS*
T001	55	12	0.91	5.3	8.5	–	–	–	–	–	72	12.5	0.90	5.8	8.9	69	12.5	0.86	6.7	8.5	70	13	0.80	6.5	8.4
T002	60	12.5	0.80	6.0	7.8	50	13	0.85	7.4	8.2	–	–	–	–	–	65	12.5	0.78	6.1	8.0	59	13.5	0.78	7	7.1
T003	–	–	–	–	–	64	22.5	0.73	5.7	8.3	70	21.5	0.85	6.4	8.3	68	21.5	0.79	7.3	8.1	–	–	–	–	–
T004	75	10	0.85	6.2	9.0	–	–	–	–	–	68	10	0.75	6.1	8.0	70	9.5	0.87	5.8	7.8	63	11.5	0.76	5.9	7.9
T005	70	18	0.88	7	8.6	65	18.5	0.78	5.5	7.4	65	19	0.89	6.5	9.2	73	19	0.85	6	7.5	60	20	0.74	5.7	8.8
T006	–	–	–	–	–	70	14	0.82	6	7.8	71	12.5	0.83	7.5	8.5	72	14.5	0.91	6.9	8.9	65	14	0.81	7.1	8.5
T007	48	18.5	0.85	5.5	8.5	60	18	0.86	7.1	8.0	–	–	–	–	–	–	–	–	–	–	71	20	0.86	6.5	8.0
T008	70	5	0.92	6.3	8.5	58	6	0.80	4.8	8.5	60	4.5	0.79	7	8.3	–	–	–	–	–	62	6	0.88	6.8	7.8
T009	64	20.5	0.78	4.8	7.9	72	22	0.74	6.5	8.0	55	21.5	0.76	7.4	7.9	58	20.5	0.75	7	7.1	55	22	0.79	7	7.5
T010	68	14	0.86	6.5	7.5	–	–	–	–	–	65	14	0.81	6.3	8.4	75	14	0.86	5.9	8.5	–	–	–	–	–
T011	59	23.5	0.90	5.8	9.1	64	24.5	0.85	5.8	9.0	64	22.5	0.85	6.7	8.4	56	23.5	0.80	6	8.0	–	–	–	–	–
T012	71	8	0.82	6.5	8.3	58	8.5	0.83	5.5	8.7	–	–	–	–	–	69	9	0.92	7	8.7	70	9	0.83	6	8.0
T013	–	–	–	–	–	73	7	0.79	6.4	8.1	68	5.5	0.74	7.3	7.2	–	–	–	–	–	71	7.5	0.85	5.9	8.3
T014	62	15	0.75	7.1	7.6	–	–	–	–	–	70	15.5	0.76	5.9	7.8	55	17	0.78	6.3	7.9	64	16	0.80	7.3	7.9
T015	70	4.5	0.80	5.9	8.4	67	7.5	0.81	5.9	7.3	–	–	–	–	–	67	5.5	0.74	5.8	7.5	–	–	–	–	–

Table 4 Preliminary configuration solution

$IMTOS_1$	T001,T007,T011,T015
$IMTOS_2$	T002,T003,T008,T012
$IMTOS_3$	T006,T010,T013
$IMTOS_4$	T004,T014
$IMTOS_5$	T005,T009

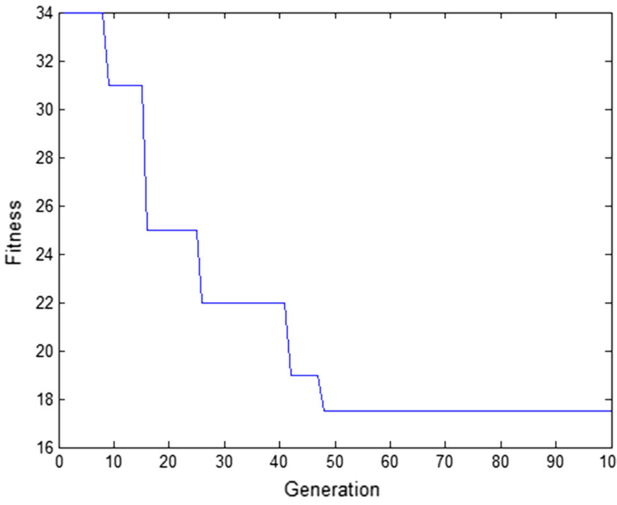


Fig. 8 The schedule result on $IMMCS_1$

At time At^8, the eighth batch that contains fifteen task orders arrives at the CMfg platform. The detailed information is given in Table 1. Meanwhile, there are five IMMCSs being operative, denoted as $\{IMMCS_1, IMMCS_2, IMMCS_3, IMMCS_4, IMMCS_5\}$. After completing the service discovery procedure, each order can find their candidate services. The processing time of orders on IMMCSs is listed in Table 2, in which the symbol "-" means this IMMCS is not included in the candidate set. According to the previous schedule, the orders being processed and those unexecuted ones in the queues are shown in Fig. 7. The data (x, y, z) denote the order information, "x" is the processing time; "y" is the due date; "z" is the weight of the order. The order being processed can be represented by: the gray rectangle denotes the part has been finished; the red rectangle denotes the part has not been completed. The rest blue rectangles are scheduled and waiting to be executed.

Based on the real-time task queues, the priorities of orders on IMMs can be calculated by Eq. (2). The evaluation criteria of IMMCSs are shown in Table 3. Through the TOPSIS based evaluation method, service demanders can customize the weights of each criterion to find their satisfactory IMMCSs. In other words, the task assignments on IMMs, i.e. the preliminary configuration solutions are shown in Table 4.

After new orders are accepted by IMMCSs, RTS is performed immediately. Take $IMMCS_1$ for example, totally eight orders are scheduled. Here, we use GA to achieve a satisfactory solution. The GA parameters are listed as follows:$N_{pop} = 50$, $N_{gen} = 100$, $P_c = 0.8$, $P_m = 0.1$. The result is shown in Fig. 8. Then the new order sequence is determined. All the schedule results on IMMCSs are shown in Fig. 9. These optimal configuration solutions are fed back

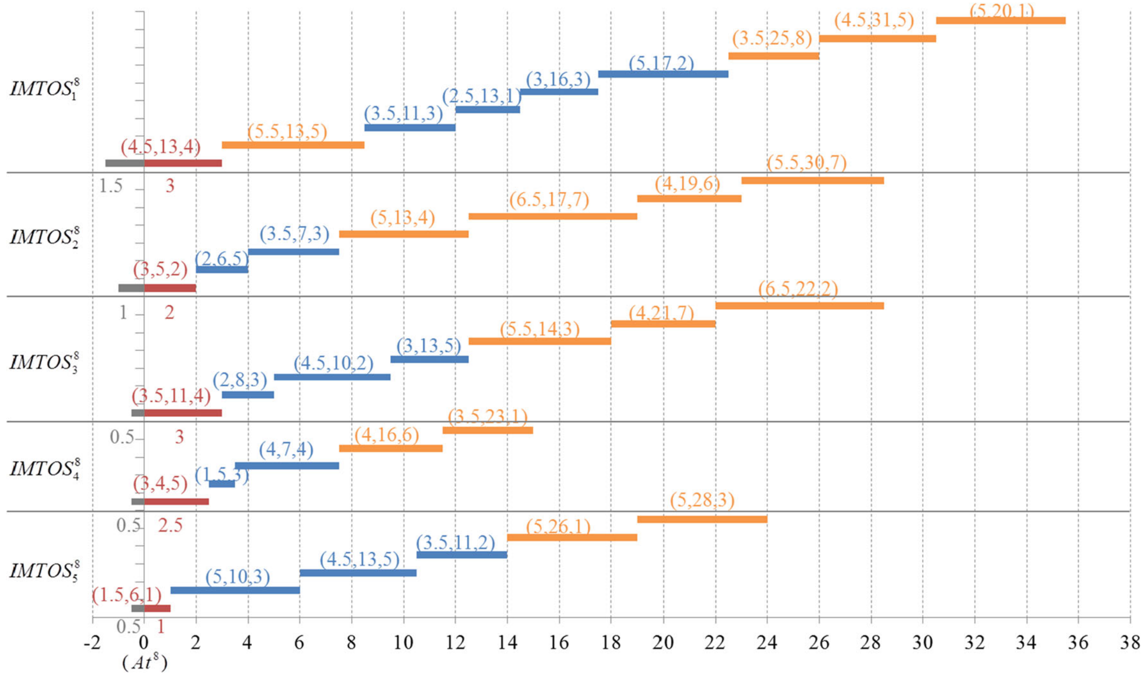


Fig. 9 Optimal configuration solutions

to the IMMCSs and waiting to be executed. When another batch of orders arrives, repeat above configuration procedure.

Conclusion and future work

To solve the problems that most injection molding enterprises are faced with, e.g. inadequate task orders, unreasonable production scheduling, etc., applying CMfg in the injection molding industry seems to be an effective solution. When comes to the specific implementation and considers the features of injection molding, three challenges need to be addressed, i.e. IMMCS modeling, IMMCS selection and scheduling. All three parts are tightly coupled with each other and key segments in the service configuration.

In this work, the overall architecture of IMMCS configuration and its components are presented. Within this architecture, IME-CMfgSP is established to realize the full-scale sharing, free circulation and efficient integration of IMMs. In the servitization of IMM, RIMS is designed to acquire real-time production data in the molding process. Based on it, a four-dimension information model of IMM is built to transform the IMM into a virtual machining service. By employing the ontology method, OWL-S, and MS-UDDI, the virtual machining services are encapsulated as IMMCSs, and then registered and published into the platform. The IMMCS configuration engine aims to assign orders to IMMCSs in a dynamic environment. The proposed task-driven proactive service discovery method facilitates the efficient and proactive discovery of potential IMMCSs for orders. A multi-objective evaluation method based on TOPSIS is designed for customers to find satisfactory services from massive candidate IMMCSs. Furthermore, RTS is presented to schedule or re-schedule all the orders on the IMM timely through a single-machine scheduling model, minimizing the total weighted tardiness. In other words, the penalty cost caused by the delay can be reduced. This mechanism helps IMMPs dynamically adapt to the changing task queue, especially the interruption resulted from the newly arrived orders. At last, the proposed configuration method is validated by a numerical simulation.

This work still has some limitations that need further research. Firstly, we mainly focus on the load status of the IMM while other dynamic characteristics like pressure and temperature are few concerned. And how PM works needs deeper research, including the parameter monitoring and analysis, exception diagnosis and prediction. So that customers can check the production progress and the quality of the product through PM at any time. Secondly, after task orders are assigned to IMMCSs, customers are not fully involved in the entire manufacturing process. A concrete interaction mechanism to promote information exchange and ensure the quality of the product through the product life cycle needs to be further studied. So that joint manufacturing for customers and enterprises is achieved and the customer satisfaction can be greatly increased. Finally, in the future research, the IMMCS and the LCS are combined as a cloud service composition that undertakes task orders throughout the production part and the transportation part. How to orchestrate service compositions during the whole configuration process to meet more complex requirements needs to be further researched.

Acknowledgements The authors would like to acknowledge financial supports of National Science Foundation of China (51675441) and the 111 Project Grant (B13044).

References

Adamson, G., Wang, L. H., Holm, M., & Moore, P. (2015). Cloud manufacturing—A critical review of recent development and future trends. *International Journal of Computer Integrated Manufacturing*, *3052*, 1–34.

Ameri, F., & Patil, L. (2012). Digital manufacturing market: A semantic web-based framework for agile supply chain deployment. *Journal of Intelligent Manufacturing*, *23*(5), 1817–1832.

Ameri, F., Urbanovsky, C., & Mcarthur, C. (2012). A systematic approach to developing ontologies for manufacturing service modeling. In *7th international conference on formal ontology in information systems (FOIS 2012)*. Graz, Austria.

Barenji, A. V., Barenji, R. V., & Roudi, D. (2016). A dynamic multi-agent-based scheduling approach for SMEs. *The International Journal of Advanced Manufacturing Technology*,. doi:10.1007/s00170-016-9299-4.

Barenji, R. V., Barenji, A. V., & Hashemipour, M. (2014). A multi-agent RFID-enabled distributed control system for a flexible manufacturing shop. *The International Journal of Advanced Manufacturing Technology*, *71*(9), 1773–1791.

Buckholtz, B., Ragai, I., & Wang, L. H. (2015). Cloud manufacturing: Current trends and future implementations. *Journal of Manufacturing Science and Engineering*, *137*(4), 40902-1–40902-9.

Cao, Y., Wang, S. L., Kang, L., & Gao, Y. (2016). A TQCS-based service selection and scheduling strategy in cloud manufacturing. *International Journal of Advanced Manufacturing Technology*, *82*(1–4), 235–251.

Chen, F. Z., Dou, R. L., Li, M. Q., & Wu, H. (2016). A flexible QoS-aware web service composition method by multi-objective optimization in cloud manufacturing. *Computers and Industrial Engineering*, *99*, 423–431.

Chhim, P., Chinnam, B. R., & Sadawi, N. (2017). Product design and manufacturing process based ontology for manufacturing knowledge reuse. *Journal of Intelligent Manufacturing*,. doi:10.1007/s10845-016-1290-2.

Choi, C. R., & Jeong, H. Y. (2014). A broker-based quality evaluation system for service selection according to the QoS preferences of users. *Information Sciences*, *277*, 553–566.

Chongwatpol, J., & Sharda, R. (2013). RFID-enabled track and traceability in job-shop scheduling environment. *European Journal of Operational Research*, *227*(3), 453–463.

de Oliveira, V. C., & Silva, J. R. (2015). A service-oriented framework to the design of information system service. *Journal of Service Science Research*, *7*(2), 55–96.

Duron, C., Ould Louly, M. A., & Proth, J. M. (2009). The one machine scheduling problem: Insertion of a job under the real-time constraint. *European Journal of Operational Research*, *199*(3), 695–701.

Dymova, L., Sevastjanov, P., & Tikhonenko, A. (2013). An approach to generalization of fuzzy TOPSIS method. *Information Sciences*, *138*, 149–162.

Farsi, J. Y., & Toghraee, M. T. (2014). Identification the main challenges of small and medium sized enterprises in exploiting of innovative opportunities (Case study: Iran SMEs). *Journal of Global Entrepreneurship Research*, *4*(1), 4.

Haddad, J. El, Manouvrier, M., & Rukoz, M. (2010). TQoS: Transactional and QoS-aware selection algorithm for automatic web service composition. *IEEE Transactions on Services Computing*, *3*(1), 73–85.

Hall, N. G., & Potts, C. N. (2004). Rescheduling for new orders. *Operations Research*, *52*(3), 440–453.

Haupt, R. (1989). A survey of priority rule-based scheduling. *OR Spektrum*, *11*(1), 3–16.

He, W., & Da Xu, L. (2015). A state-of-the-art survey of cloud manufacturing. *International Journal of Computer Integrated Manufacturing*, *28*(3), 239–250.

Huang, B. Q., Li, C. H., Yin, C., & Zhao, X. P. (2013). Cloud manufacturing service platform for small- and medium-sized enterprises. *International Journal of Advanced Manufacturing Technology*, *65*(9–12), 1261–1272.

Khalfallah, M., Figay, N., Da Silva, C. F., & Ghodous, P. (2014). A cloud-based platform to ensure interoperability in aerospace industry. *Journal of Intelligent Manufacturing*, *27*(1), 119–129.

Kubler, S., Holmström, J., Främling, K., & Turkama, P. (2016). Technological theory of cloud manufacturing. In *Service orientation in holonic and multi-agent manufacturing* (Vol. 640, pp. 267–276). Springer.

Lartigau, J., Nie, L. S., Xu, X. F., Zhan, D. C., & Mou, T. (2012). Scheduling methodology for production services in cloud manufacturing. In *Proceedings—2012 International joint conference on service sciences, service innovation in emerging economy: Cross-disciplinary and cross-cultural perspective*, IJCSS 2012 (pp. 34–39).

Lartigau, J., Xu, X. F., Nie, L. S., & Zhan, D. C. (2015). Cloud manufacturing service composition based on QoS with geo-perspective transportation using an improved Artificial Bee Colony optimisation algorithm. *International Journal of Production Research*, *53*(14), 4380–4404.

Li, B. H., Zhang, L., Wang, S. L., Tao, F., Cao, J. W., Jiang, X. D., et al. (2010). Cloud manufacturing: A new service-oriented networked manufacturing model. *Jisuanji Jicheng Zhizao Xitong/Computer Integrated Manufacturing Systems, CIMS*, *16*(2007), 1–7.

Li, W. X., Zhu, C. S., Yang, L. T., Shu, L., Ngai, E. C. H., & Ma, Y. J. (2015). Subtask scheduling for distributed robots in cloud manufacturing. *IEEE Systems Journal*, *PP*(99), 1–10.

Luo, H., Fang, J., & Huang, G. Q. (2015). Real-time scheduling for hybrid flowshop in ubiquitous manufacturing environment. *Computers and Industrial Engineering*, *84*, 12–23.

Luo, Y. L., Zhang, L., Tao, F., Ren, L., Liu, Y. K., & Zhang, Z. Q. (2013). A modeling and description method of multidimensional information for manufacturing capability in cloud manufacturing system. *International Journal of Advanced Manufacturing Technology*, *69*(5–8), 961–975.

Mai, J. G., Zhang, L., Tao, F., & Ren, L. (2016). Customized production based on distributed 3D printing services in cloud manufacturing. *International Journal of Advanced Manufacturing Technology*, *84*(1–4), 71–83.

Modekurthy, V. P., Liu, X. F., Fletcher, K. K., & Leu, M. C. (2015). Design and implementation of a broker for cloud additive manufacturing services. *Journal of Manufacturing Science and Engineering*, *137*(4), 40904-1–40904-10.

Oner, M., Ustundag, A., & Budak, A. (2016). An RFID-based tracking system for denim production processes. *The International Journal of Advanced Manufacturing Technology*,. doi:10.1007/s00170-016-9385-7.

Paolucci, M., Kawamura, T., Payne, T. R., & Sycara, K. (2002). Semantic matching of web services capabilities. *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)* (Vol. 2342, pp. 333–347).

Puttonen, J., Lobov, A., Soto, M. A. C., & Lastra, J. L. M. (2016). Cloud computing as a facilitator for web service composition in factory automation. *Journal of Intelligent Manufacturing*,. doi:10.1007/s10845-016-1277-z.

Quintanilla, F. G., Cardin, O., Anton, A. L., & Castagna, P. (2016). Engineering applications of artificial intelligence a modeling framework for manufacturing services in service-oriented holonic manufacturing systems. *Engineering Applications of Artificial Intelligence*, *55*, 26–36.

Rahmani, D., & Heydari, M. (2014). Robust and stable flow shop scheduling with unexpected arrivals of new jobs and uncertain processing times. *Journal of Manufacturing Systems*, *33*(1), 84–92.

Seghir, F., & Khababa, A. (2016). A hybrid approach using genetic and fruit fly optimization algorithms for QoS-aware cloud service composition. *Journal of Intelligent Manufacturing*,. doi:10.1007/s10845-016-1215-0.

Škulj, G., Vrabič, R., Butala, P., & Sluga, A. (2015). Decentralised network architecture for cloud manufacturing. *International Journal of Computer Integrated Manufacturing*,. doi:10.1080/0951192X.2015.1066861.

Sokolov, B., Benyamna, K., & Korolev, O. (2016). RFID technology for adaptation of complex systems scheduling and execution control models. In *Automation control theory perspectives in intelligent systems* (pp. 433–442).

Srinivasan, N., Paolucci, M., & Sycara, K. (2004). An efficient algorithm for OWL-S based semantic search in UDDI. *Semantic Web Services and Web Process Composition*, *3387*, 96–110.

Tao, F., Laili, Y. J., Da Xu, L., & Zhang, L. (2013). FC-PACO-RM: A parallel method for service composition optimal-selection in cloud manufacturing system. *IEEE Transactions on Industrial Informatics*, *9*(4), 2023–2033.

Tapoglou, N., & Mehnen, J. (2016). Cloud-based job dispatching using multi-criteria decision making. *Procedia CIRP*, *41*, 661–666.

Valilai, O. F., & Houshmand, M. (2014). A platform for optimisation in distributed manufacturing enterprises based on cloud manufacturing paradigm. *International Journal of Computer Integrated Manufacturing*, *27*(11), 1031–1054.

Wang, J. W., Liu, D., Ip, W. H., Zhang, W. J., & Deters, R. (2014). Integration of system-dynamics, in system information modeling. *IEEE Transactions on Industrial Informatics*, *10*(2), 847–853.

Wang, J. W., Muddada, R. R., Wang, H. F., Ding, J. L., Lin, Y. Z., Liu, C. L., et al. (2016a). Toward a resilient holistic supply chain network system: Concept, review and future direction. *IEEE Systems Journal*, *10*(2), 410–421.

Wang, J. W., Wang, H. F., Furuta, K., & Ip, W. H. (2016b). On domain modelling of the service system with its application to enterprise information systems. *Enterprise Information Systems*, *10*(1), 1–16.

Welbourne, E., Battle, L., Cole, G., Gould, K., Rector, K., Raymer, S., et al. (2009). Building the internet of things using RFID: The RFID ecosystem experience. *IEEE Internet Computing*, *13*(3), 48–55.

Wilde, S. (2010). Small and medium-sized enterprises. In *Customer knowledge management* (pp. 11–18). Springer.

Wu, D. Z., Greer, M. J., Rosen, D. W., & Schaefer, D. (2013). Cloud manufacturing: Strategic vision and state-of-the-art. *Journal of Manufacturing Systems*, *32*(4), 564–579.

Xu, W. J., Tian, S. S., Liu, Q., Xie, Y. Q., De Zhou, Z., & Pham, D. T. (2016). An improved discrete bees algorithm for correlation-aware service aggregation optimization in cloud manufacturing. *International Journal of Advanced Manufacturing Technology*, *84*(1–4), 17–28.

Xu, W. J., Yu, J. J., De Zhou, Z., Xie, Y. Q., Pham, D. T., & Ji, C. Q. (2015). Dynamic modeling of manufacturing equipment capability using condition information in cloud manufacturing. *Journal of Manufacturing Science and Engineering*, *137*(4), 1–36.

Xu, X. (2012). From cloud computing to cloud manufacturing. *Robotics and Computer-Integrated Manufacturing*, *28*(1), 75–86.

Zhang, Y. F., Liu, S. C., Liu, Y., & Li, R. (2016a). Smart box-enabled product-service system for cloud logistics. *International Journal of Production Research*, *54*(22), 6693–6706. doi:10.1080/00207543.2015.1134840.

Zhang, Y. F., Qian, C., Lv, J. X., & Liu, Y. (2016). Agent and cyber-physical system based self-organizing and self-adaptive intelligent shopfloor. *IEEE Transactions on Industrial Informatics*,. doi:10.1109/TII.2016.2618892.

Zhang, Y. F., Ren, S., Liu, Y., & Si, S. B. (2016c). A big data analytics architecture for cleaner manufacturing and maintenance processes of complex products. *Journal of Cleaner Production*, *142*, 626–641.

Zhang, Y. F., Zhang, G., Liu, Y., & Hu, D. (2015). Research on services encapsulation and virtualization access model of machine for cloud manufacturing. *Journal of Intelligent Manufacturing*,. doi:10.1007/s10845-015-1064-2.

Zhang, Y. F., Zhang, G., Wang, J. Q., Sun, S. D., Si, S., & Yang, T. (2015b). Real-time information capturing and integration framework of the internet of manufacturing things. *International Journal of Computer Integrated Manufacturing*, *28*(8), 811–822.

Zissis, D., Lekkas, D., Azariadis, P., Papanikos, P., & Xidias, E. (2016). Collaborative CAD/CAE as a cloud service. *International Journal of Systems Science: Operations and Logistics*. doi:10.1080/23302674.2016.1186237.