SPS2020
K. Säfsten and F. Elgh (Eds.)
© 2020 The authors and IOS Press.
This article is published online with Open Access by IOS Press and distributed under the terms of the Creative Commons Attribution Non-Commercial License 4.0 (CC BY-NC 4.0). doi:10.3233/ATDE200194

# Collaboration of Smart Device in Cloud Manufacturing: A Case of Active Recommendation Model Based on Service Agent

Chun ZHAO<sup>a,b</sup>, Mo CHEN<sup>b</sup>, Lihui WANG<sup>b,1</sup> and Lin ZHANG<sup>c</sup> <sup>a</sup>Computer School, Beijing Information Science and Technology University (BISTU), Beijing 100101, China <sup>b</sup>Department of Production Engineering, KTH Royal Institute of Technology, Stockholm 100 44, Sweden <sup>c</sup>School of Automation Science and Electrical Engineering, Beihang University, Beijing 100192, China

Abstract. By the developing and the application of cloud manufacturing, many new business modes based on the IoT, CPS, cloud computing are springing up. It is worthwhile to model a new business mode simulate it to verify the feasibility. In this paper, with a demand of smart devices from a fresh-mix drink company, service agents are used for modeling and a basic comparison model is proposed for the negotiation between service providers to vote for the service provider that best suits the consumer's demand. A simulation platform that originates from cloud manufacturing is used to simulate the drink ordering, provider negotiation and provider recommendation. Simulation results show that consumers can obtain expected drinks after the negotiations and votes made by providers. Moreover, the system can recommend drinks with different freshness depending on the expiration parameter set by the company.

Keywords. Cloud manufacturing, Smart device, Service agent

# Introduction

The concept of cloud manufacturing [1] is developed rapidly and is being verified around the world in recent years [2]. Many new technologies and concepts related information are flourishing such as Industry [3], IoT [4][5], CPS [6], Big data [7], Digital-twin [8][9]. These technologies and concepts are constantly integrated into the cloud manufacturing environment gradually from theory to application of cloud manufacturing [10]. Particularly, the development of cloud technology and CPS technology makes manufacturing to a ubiquitous service [11][12]. The smart devices were also born in this environment. The smart device is part of the smart product-service oriented system [13]. In this paper, the behavior of the smart devices, which are production service terminals in the cloud manufacturing environment, is studied to optimize the strategy of product supply. In order to reproduce or predict all kinds of behaviors in the cloud manufacturing

<sup>&</sup>lt;sup>1</sup> Corresponding Author. lihuiw@kth.se

environment such as interactive processes, business processes, algorithms, business modes, the architecture and application of cloud manufacturing simulation platform are constantly proposed [14][15]. The simulation of business mode is different between cloud manufacturing and other environments, that the manufacturing resource provides the service to users in cloud manufacturing, so the function of the manufacturing resource is encapsulated as a service. In the simulation of smart devices, a smart device is defined as an agent that offers the service and thus can be called a service agent. How smart devices effectively offer the services is focused in this paper. In the cloud manufacturing, there are three typical characteristics of the scheduling including operation mode-related, resource/service- and requirement-related, technology-enabled and knowledge-related [16]. Using the unified optimization of the service platform is a complex system engineering. The application of the decentralized idea of service agent to optimize can save a lot of computing resources of the platform.

The case in this paper comes from a freshly-mixed drink company. The requirement of the company is that there are many smart machines distributed in the city. Different types of drinks are available in these machines and some of the machines offer the same type of drink. When a consumer submits a demand (i.e., an order) for a type of drink on an App, the system recommends the smart machine that best suits the consumer's need. To meet this demand, the cloud computing methods unified scheduling in the cloud are usually used in traditional solutions. But the approach needs more high-performance servers and high-speed and stable network. In this case, the customer does not need to rent a high-performance server and the network can be unstable and slow. According to the requirement and environment, this paper proposed a model of active recommendation based on the service agent. By using the model, after receiving the order from a consumer, the smart machines can negotiate, and the system will choose the most suitable machine through voting and recommend it to the consumer. In the simulation platform, there are two types of service agents. One is the service user which represents the consumer. Another is the service provider which represents the smart machine.

The remainder of this paper is organized as follows. Firstly, the simulation architecture based on the service agent is introduced in Section 1. Secondly, the modeling of active recommendation mode is presented in Section 2. In Section 3, the process of the simulation is described in detail and the analysis of the simulation is presented. Finally, Section 4 concludes the paper.

## 1. Simulation Architecture based on Service-Agent

### 1.1. Modeling of Service-Agent

A simple agent model is proposed to solve some independent, distributed, cooperative problems[17]. It focuses on the BDI model including Belief, Desire and Intention. The agent can simulate the behavior of a group to predict the tendency of the environment.

As shown in Figure 1, an agent receives information through the interpretation of the environment, and delivers the information to the processor. The processor gets the information and processes it according to the rules defined by the agent. Finally, the processing results are feeded as responds to the environment. The environment includes random events and other agents. Most agents can be combined to form an agent network. Each agent may have different rules, and through the definition of these rules, the mapping to the real world can be realized.



Figure 1. The Model of simple agent

In the environment of cloud manufacturing, a service is formed by the capabilities that one or more manufacturing resources have [18]. For the definition of resources in the cloud manufacturing environment, most of themanufacturing resources are intelligent to deal with problems under certain rules. Therefore, a manufacturing service should also be intelligent. In the cloud manufacturing simulation platform, the service agent (SA) can be formed by encapsulating the service to simulate the behaviors in the cloud manufacturing environment.



Figure 2. The Conceptual model of service agent

As shown in Figure 2, a conceptual model of service agent is presented [19]. A service agent can be either a service provider or a service consumer. An agent can be both service provider and consumer in most instances. A service agent is divided into two parts: intelligent module and manufacturing service. The model of manufacturing

service includes a service interface (input and output), a description of service function, a data set recording service static data and the temporary data generated during service execution, and a handle for invoking manufacturing resources. The expression for the model of service is:

SE<sub>cmfg</sub> = <SID, RSID, Info<sub>State</sub>, Interface, Info<sub>Basic</sub>, Func<sub>Templ</sub>, Func<sub>Data</sub>, Func<sub>Call</sub>, Func<sub>order</sub>> (1)

The intelligent module includes: an interpreter which receives other service agents' message, an actuator which sends messages to other service agents, a message queue which stores the message, a clock which triggers events, a knowledge library which stores the data such as state, rules, basic information and functional data, and a processor which consists of two function units in the condition of the service agent being a service provider and a service consumer respectively. The expression for the model of service agent is as follow:

SAcmfg = <SAID, SID, Infostate, Infobasic, Msgsa, ClksA, Funcreq, Functrans, Funcquery, Funcrespond> (2)

# 1.2. Simulation Platform Architecture based on SA

The cloud manufacturing simulation platform represents the behaviors and commercial activities in the cloud manufacturing environment [11]. The cloud manufacturing simulation platform conducts various researches and experiments on the cloud manufacturing system, such as: cloud manufacturing business model analysis, cloud service characteristic analysis, cloud service matching and combination of supply and demand, cloud service scheduling, cloud service network characteristic analysis, cloud service network evolution, etc. An architecture of simulation based on service agent is divided into two parts including service center and service agent network, is as follow:



Figure 3. The Conceptual model of service agent

(1) Model of Service Center: The service center is used to manage the service agents in cloud manufacturing simulation platform. The model of the service center is given as:

## SC<sub>cmfg</sub>=<Clk<sub>SC</sub>, Msg<sub>sc</sub>, Info<sub>env</sub>, Bulletin<sub>req</sub>, Bulletin<sub>service</sub>, Func<sub>sample</sub>, Func<sub>reqpub</sub>, Func<sub>spub</sub>

#### , $Func_{getinfo} > (3)$

In the Equation 3, *Clk<sub>SC</sub>* is a clock of the service center, used to simulate the clock of environment. *Msg<sub>sc</sub>* is a message controller which processes the messages between service agents. *Info<sub>env</sub>* is used to store the real-time data and historical data of the environment. *Bulletin<sub>req</sub>* is a bulletin for publishing demand information, which stores the demand information publish by the service consumer. *Bulletin<sub>service</sub>* is a bulletin for publishing service information, which stores the service information from the service provider. *Func<sub>sample</sub>* is a sampling function which collects and stores real time data about the environment. *Func<sub>reqpub</sub>* is a function which manages the publishment of demand. *Func<sub>spub</sub>* is a function which used to manage the publication of service. *Func<sub>getinfo</sub>* is a function which gets information from the service center.

(2) Model of Simulation Platform: The cloud manufacturing simulation platform is extended on the basis of the service center model to enable its functions:

P=<SAcmfg, SCcmfg, Lib, Iinternal, Iexternal> (4)

In the Equation 4,  $SA_{cmfg}$  is the service agent,  $SC_{cmfg}$  is service center, *Lib* is the platform supporting database. In order to ensure the data stability in the simulation platform and facilitate the data acquisition, the platform data is managed in a database. This can reduce the redundancy of data, improve the efficiency of data storage, and speed up the query of related data. Lib is given as:

Where,  $Lib_{rs}$  is a manufacturing resource library based on meta-data.  $Lib_{se}$  is a manufacturing service library.  $Lib_{sa}$  is a service agent library.  $Lib_{sc}$  is a service center library.  $Lib_{case}$  is a manufacturing case library.

There are two interfaces for the platform. *I*<sub>internal</sub> is the internal interface for communication between modules. The internal interface only serves the internal work of the simulation platform.

*I*<sub>external</sub> is the external interface for interaction among users, realization data input and visualization of data. The data input is displayed through the static interface, and the data display is updated in real time by the visualization module. The models of the two interfaces are:

## *Iinternal*=<*Class, Function, <Param1, Param2, ....., Paramn>, <result>>*

*I<sub>external</sub>=<Class, Function, <Param<sub>1</sub>, Param<sub>2</sub>, ...., Param<sub>n</sub>>, <result>> (6) Through the simulation platform, we can simulate the business mode in cloud* 

manufacturing. For the case in this paper, we can classify service agents and create the rules of service agents to verify the active recommendation mode.

#### 2. Modeling of Active Recommendation Mode

## 2.1. Modeling of Work Flow

In this case, the participants are consumers and smart machines, represented by service consumers and service providers of service agents respectively. The service consumer proposes the demand, the service provider negotiates according to the demand, and finally a service provider responds and pushes the service. In this case, according to the service agent model in Section 1, a service agent has only one role. area consumer is only

a service consumer and not service provider, and smart machine is a service provider and not service consumer. The work flow is shown in Figure 4.



Figure 4. Work Flow of Active Recommendation mode

# 2.2. Models in this Case

The service consumer SA randomly publishes the demand to the bulletin board service center. The model of the bulletin board is shown in Equation 7.

$$Msg_{reg} = \langle SAID, Si_{param} \rangle \tag{7}$$

Equation 7 is the standard information published by requirements, and  $Si_{param}$  is the output interface parameter of service consumer requirements. The definition of the  $Si_{param}$  parameter for this case is:

$$Si_{param} = \langle \omega_{type}, \omega_{dist}, \omega_{wait} \rangle$$
 (8)

In Equation 8,  $\omega_{type}$  the type of drink required,  $\omega_{dist}$  is the weight of the service consumer's attention for the distance from the service provider, and  $\omega_{wait}$  is the weight of the service consumer's attention to the length of the queue, in which  $\omega_{dist} + \omega_{wait} = 1.0$ . The demand varies from consumer to consumer. Besides different preferences on drinks, they also pay different attention to the distance and the number of potential customers. For example, some consumers prefer to buy a drink from a place far away rather than wait a long line in a place nearby, or they prefer to wait in a long line rather than walk a long distance. Therefore, after each smart machine obtains the consumption preferences

of consumers, it can negotiate with other smart machines according to the needs. A basic comparison model is proposed in Equation 9 for the negotiation with j initiated by service provider *i*:

$$U_{(i,j)} = \sum_{m}^{k} \left[ \left(1 - \frac{Param_{m(i,c)}}{Param_{m(i,c)} + Param_{m(j,c)}}\right) \times \omega_{m_c} \right]$$
(9)

Where **Param**<sub>m(i,c)</sub> is a number related to consumer *c* and provider *i*, and  $\omega_{mc}$  is a weight related to consumer *c*. After a requirement is published, the service provider fetches the corresponding requirement from the bulletin board. In this case, the service interface is equivalent to only one interface parameter  $\omega_{type}$ . If the drink available in a service provider matches  $\omega_{type}$ , the service provider can apply for offering the service to the consumer. After the service providers confirm the requirements, they begin to negotiate with each other. That is, each service. In this case, from the basic comparison model (Equation 9), a particular model can be derived as shown in Equation 10.

$$U_{(i,j)} = \left(1 - \frac{\text{Dist}_{(i,c)}}{\text{Dist}_{(i,c)} + \text{Dist}_{(j,c)}}\right) \omega_{dist_c} + \left(1 - \frac{\text{Wait}_i}{\text{Wait}_i + \text{wait}_j}\right) \omega_{wait_c} + \left(1 - \frac{\text{Exp}_i}{\text{Exp}_i + \text{Exp}_j}\right) P_{exp}$$
(10)

In Equation 10, the service provider i that initiates the comparison is compared with the service provider j, and the result  $U_{(i,j)}$  is the score of i when i is compared with j, and c is the service consumer. In the first term on the right,  $Dist_{(i,c)}$  is the distance between service provider i and service consumer c,  $Dist_{(j,c)}$  is the distance between service provider j and service consumer c, and  $\omega_{distc}$  is the attention weight of service consumer c for the distance. The shorter the distance, the higher the evaluation score.

In the second term, *Wait*<sub>(*i*,*c*)</sub> is the queue length of service provider *i* before *c*, *Wait*<sub>(*j*,*c*)</sub> is the queue length of service provider *j* before *c*, and  $\omega_{waitc}$  is the weight of attention of service consumers c on the queue length. The shorter the queue, the higher the evaluation score.

Finally,  $Exp_i$  is the number of days to expire date of drink in provider *i*. The drinks sold at the smart machine are freshly mixed drink, which we assume it has a uniform expire time. Drinks cannot be sold if they have expired.  $Exp_i$  and  $Exp_j$  are the numbers of days to the expire dates of the drinks available in service provider *i* and *j* respectively, and  $Pex_p$  is the weight of attention for the number of days to the expire date of drinks.  $Pex_p$  is initially set by the platform.

#### 3. Analysis of Simulation

In this case, there are 53 smart devices throughout the city. Each smart device is a machine of freshly mixed drink and can provide only one type of drink. There are three types of drink available. The simulation platform generates many virtual consumers to attend the process.

In the simulation, the numbers of demands from 50 consumer agents for the three types of drink are 16, 14, and 20 respectively and thus the probabilities of the demands are 0.32, 0.28, and 0.40. The numbers of provider agents that supply the three types of drink are 18, 18, and 17 respectively and thus the probabilities of supply are 0.34, 0.34, and 0.32. Therefore, the matching degree between supply and demand is:

$$\boldsymbol{C} = \sum_{i=1}^{n} |\boldsymbol{P}_{ci} - \boldsymbol{P}_{si}| \tag{11}$$

Where,  $P_{ci}$  is the demand probability of consumer agent *c* for drinks *i*,  $P_{si}$  is the supply probability of provider agent s for drinks *i*. *C* is the overall matching degree of supply and demand. The smaller the overall matching degree, the stronger the connection between supply and demand. In this case C = 0.16. Therefore, the data generated in the simulation are reasonable, and the resources supported by the provider agents can meet the needs of consumer agents for drinks without large bias.



Figure 5. The initialization of simulation

# 3.1. Process of simulation

As shown in Figure 5, the red points are the real positions of the smart devices distributed in the city. In this simulation, we create 53 service agents as providers and 50 service agents as consumers. The platform collects five hundred pieces of sampling data including application, negotiation, recommendation, states of the service agent, etc. The sampling frequency is 2,000 milliseconds.



#### Figure 6. Two types of service agent

As shown in Figure 6, the platform generates 50 consumer agents and distribute them to the city map randomly during the initialization. The consumer agents, which are

represented by blue points in the map, can generate demands of drinks randomly after the initialization and publish the demands to the service center.



Figure 7. The schematic diagram of simulation processing

As shown in Figure 7, the demands are received by the service center and posted on the demand bulletin. When the service providers see the information on the bulletin, those who can offer the drink will start to negotiate with each other. After several rounds of negotiation, the service providers will recommend one provider to the service consumer. When the service consumer receives the recommendation, a transaction is created with the recommended service agent, and a black line is connected on the city map.

# 3.2. Process of negotiation

When a consumer agent sends a demand, the provider agents who satisfy the  $\omega_{type}$  begin to negotiate. The negotiation model is based on Equations 3. The weights of the consumer agent's attention for the distance  $\omega_{dist}$  are set from 0.4 to 0.6. The weights of attention for queue length  $\omega_{wait}$  are set to  $1-\omega_{dist}$ . The weight of attention for the expired time of drinks is set to 0.8.

Each provider agent j who can provide the drink of  $\omega_{ijpe}$  will send a request, and another provider agent i will vote. After the provider agent i receives the request, it will calculate  $U_{(i,j)}$  and  $U_{(j,i)}$ , and obtain  $U'_{(i,j)} = U_{(i,j)} - U_{(j,i)}$ . If  $U'_{(i,j)}$  is less than 0, it means that the score of provider agent i is less than the provider agent j and agent i will vote "yes"; otherwise, agent i will vote "no".

	agent01	agent02	agent03	agent04	agent05	agent06	agent07	agent08	agent09	agent10	agent11	agent12	agent13	agent14	agent15	agent16	agent17
agent01	0	0.0136	0.0717	0.1645	0.1645	-0.342	-0.584	0.2924	-0.589	-0.115	0.2896	0.2409	0.2702	-0.541	0.2473	-0.044	0.0344
agent02	-0.014	0	0.0329	0.1717	-0.021	-0.497	-0.585	0.2722	-0.664	-0.2	0.2801	0.2132	0.3158	-0.756	0.2694	-0.083	0.0408
agent03	-0.072	-0.033	0	0.129	-0.048	-0.387	-0.651	0.2416	-0.671	-0.233	0.2485	0.1882	0.2531	-0.63	0.2183	-0.098	0.0043
agent04	-0.164	-0.172	-0.129	0	-0.188	-0.611	-0.723	0.1127	-0.805	-0.329	0.1157	0.0492	0.1476	-0.865	0.1024	-0.243	-0.119
agent05	-0.01	0.0207	0.0479	0.1883	0	-0.484	-0.591	0.2966	-0.663	-0.135	0.3006	0.2341	0.3312	-0.738	0.2878	-0.059	0.0607
agent06	0.3416	0.4968	0.3874	0.6108	0.4835	0	0.0584	0.6227	-0.019	0.3008	0.6034	0.5381	0.7033	-0.252	0.685	0.4879	0.613
agent07	0.5845	0.585	0.6511	0.7234	0.5914	-0.058	0	0.8326	-0.04	0.4957	0.818	0.7822	0.8007	-0.211	0.7915	0.5074	0.563
agent08	-0.292	-0.272	-0.242	-0.113	-0.297	-0.623	-0.833	0	-0.883	-0.425	0.0109	-0.056	0.0379	-0.875	-0.012	-0.338	-0.215
agent09	0.5885	0.664	0.6706	0.8046	0.6626	0.0185	0.0404	0.8828	0	0.4877	0.8705	0.8238	0.883	-0.179	0.8728	0.5951	0.67
agent10	0.1146	0.2005	0.2335	0.3295	0.1354	-0.301	-0.496	0.4247	-0.488	0	0.4842	0.4217	0.4987	-0.523	0.4135	0.0662	0.1737
agent11	-0.29	-0.28	-0.248	-0.116	-0.301	-0.603	-0.818	-0.011	-0.871	-0.484	0	-0.067	0.0292	-0.862	-0.018	-0.342	-0.218
agent12	-0.241	-0.213	-0.188	-0.049	-0.234	-0.538	-0.782	0.0561	-0.824	-0.422	0.0668	0	0.0916	-0.798	0.0468	-0.275	-0.154
agent13	-0.27	-0.316	-0.253	-0.148	-0.331	-0.703	-0.801	-0.038	-0.883	-0.499	-0.029	-0.092	0	-0.944	-0.05	-0.388	-0.265
agent14	0.5406	0.7561	0.6297	0.8652	0.7381	0.2518	0.2105	0.8752	0.1792	0.5231	0.8619	0.7984	0.9442	0	0.9309	0.7397	0.8497
agent15	-0.247	-0.269	-0.218	-0.102	-0.288	-0.685	-0.791	0.0123	-0.873	-0.413	0.0184	-0.047	0.0497	-0.931	0	-0.341	-0.216
agent16	0.0441	0.0832	0.098	0.2435	0.059	-0.488	-0.507	0.3376	-0.595	-0.066	0.3423	0.2754	0.3877	-0.74	0.3412	0	0.1239
agent17	-0.034	-0.041	-0.004	0.1185	-0.061	-0.613	-0.563	0.2146	-0.67	-0.174	0.2178	0.1545	0.2648	-0.85	0.2161	-0.124	0

Figure 8. The schematic diagram of simulation processing

An example of adjacency matrix of service agent negotiation is shown in Figure 8. The matrix shows the results of all rounds of voting. Finally, the provider agent 13 is recommended to the consumer agent after negotiation. In order to increase the randomness of the service agents, the response time of each service agent is different. Therefore, the time of sending requests and the time of collecting votes are also different.



Figure 9. An example of adjacency matrix of service agent negotiation

As shown in Figure 9, each yellow block represents that a request is sent by a provider agent, and each red block represents that the votes are collected by a provider agent. For example, the agent 01 sends a request at time tn, and collects only some of the votes at time  $t_{n+1}$ . Then agent 01 collects the remaining votes at times  $t_{n+4}$ ,  $t_{n+6}$  and  $t_{n+8}$ .

#### 3.3. Analysis of Simulation

In this simulation, the service network of service agents is generated during simulation initialization. Service agents in the network are divided into provider agents (smart device) and consumer agents (consumers). In this simulation the service agent has only one role, that is, either the service provider or the service consumer. The relationship between service agents is service provider with service consumer, and there is no relationship of service composition or supply chain.

In this case, according to the requirements by drink manufacturers, provided that the needs of consumers are met, it is necessary to sell drinks before their expire date as far

as possible. Therefore, during the definition of the  $P_{exp}$ , it should be meet the following condition:

$$P_{exp} > \frac{\sum_{i}^{n} (\omega_{dist_{i}} + \omega_{wait_{i}})}{2n}$$
(12)

According to Equation 12, when  $P_{exp}$  meets this condition, the drink suppliers can guide the behavior of consumers to reduce the waste of drink while meeting the demands and preferences of consumers. In the simulation, based on the proposed models, every consumer can obtain a recommendation which matches the expectation of the consumer. In addition, the above experiments shows that  $P_{exp}$  directly affects the quality of drinks, and the higher the value of  $P_{exp}$  is, the fresher a drink is offered to a consumer.

# 4. Conclusions

In this paper, an active recommendation mode in cloud manufacturing based on service agent has been presented. Based on a real case of drink sales, the simulation of this business model has been carried out with the support of the cloud manufacturing simulation platform. For the freshly-mixed drink company in this paper, the sales model has been verified to be cost-efficient with reduced waste of drinks. The method of modeling and simulation in this paper can be used by other companies with the same requirement. The contribution of this study is concluded as follows:

- A collaborative approach for smart machines has been presented. The approach can be used in the scenario of distributed smart machines, low-performance cloud server resources and unstable and slow network.
- The proposed method of agent modeling and simulation can be used for verification of business model.
- This paper provides a reference to self-service product companies for rapid deployment and implementation of their systems.

With the development and implementation of cloud manufacturing, many new business models are emerging. In the future, the study will be focused on the serviceagent-based modeling and simulation. By extending the service agent into factory level or even equipment level, the co-simulation in cloud manufacturing environment will be realized.

## Acknowledgement

This research is supported by the National Key Research and Development Program of China (No. 2018YFB1701602).

# References

- [1] L. Zhang, Y. Luo, F. Tao, et al. Cloud manufacturing: a new manufacturing paradigm. *Enterprise Information Systems*, vol. 8, no. 2, 2014, pp. 167-187.
- [2] L. Ren, L. Zhang, L. Wang, et al. Cloud manufacturing: key characteristics and applications. *International Journal of Computer Integrated Manufacturing*, vol. 30, no. 6, 2017, pp. 501-515.
- [3] Y. Liu, X. Xu. Industry 4.0 and cloud manufacturing: A comparative analysis. *Journal of Manufacturing Science and Engineering*, vol. 139, no. 3, 2017, pp. 034701.

- [4] S. Li, L. D. Xu, S. Zhao. The internet of things: a survey. *Information Systems Frontiers*, vol. 17, no. 2, 2015, pp. 243-259.
- [5] F. Tao, Y. Cheng, L. D. Xu, et al. CCIoT-CMfg: cloud computing and internet of things-based cloud manufacturing service system. *IEEE Transactions on Industrial Informatics*, vol. 10, no. 2, 2014, pp. 1435-1442.
- [6] C. Yu, X. Xu, Y. Lu. Computer-integrated manufacturing, cyber-physical systems and cloud manufacturing–concepts and relationships. *Manufacturing letters*, vol. 6, 2015, pp. 5-9.
- [7] J. Li, F. Tao, Y. Cheng, et al. Big data in product lifecycle management. *The International Journal of Advanced Manufacturing Technology*, vol. 81, no. 1-4, 2015, pp. 667-684.
- [8] F. Tao, M. Zhang. Digital twin shop-floor: a new shop-floor paradigm towards smart manufacturing. *Ieee Access*, vol. 5, 2017, pp. 20418-20427.
- [9] D. Ngo, D. A. Guerrazubiaga, G. Gonzálezbadillo, R. V. Barenji. Towards a Digital Twin for Cloud Manufacturing: Case Study. ASME 2018 International Mechanical Engineering Congress and Exposition. American Society of Mechanical Engineers, 2018, pp. V002T02A072-V002T02A072.
- [10] Y. Liu, L. Wang, X. V. Wang. Cloud manufacturing: latest advancements and future trends. *Procedia Manufacturing*, vol. 25, 2018, pp. 62-73.
- [11] L. Wang, X. V. Wang. Cloud-Based Cyber-Physical Systems in Manufacturing. Springer International Publishing, 2018.
- [12] X. V. Wang, L. Wang, A. Mohammed, et al. Ubiquitous manufacturing system based on Cloud: A robotics application. *Robotics and Computer-Integrated Manufacturing*, vol. 45, 2017, pp. 116-125.
- [13] S. Chowdhury, D. Haftor, N. Pashkevich. Smart Product-Service Systems (Smart PSS) in Industrial Firms: A Literature Review. 10th CIRP Conference on Industrial Product-Service Systems, IPS2 2018, 29-31 May 2018, Linköping, Sweden, vol. 73, 2018, pp. 26-31.
- [14] F. Tao, J. Cheng, Y. Cheng, et al. SDMSim: a manufacturing service supply-demand matching simulator under cloud environment. *Robotics and computer-integrated manufacturing*, vol. 45, 2017, pp. 34-46.
- [15] C. Zhao, L. Zhang, Y. Liu, et al. Agent-based simulation platform for cloud manufacturing. *International Journal of Modeling, Simulation, and Scientific Computing*, vol. 8, no. 3, 2017, pp. 1742001.
- [16] Y. Liu, L. Wang, X. V. Wang, et al. Scheduling in cloud manufacturing: state-of-the-art and research challenges. *International Journal of Production Research*, 2018, pp. 1-26.
- [17] S. J. Russell, P. Norvig, Artificial Intelligence: A Modern Approach. Prentice-Hall, Englewood Cliffs, New Jersey. 1995, pp. 25-27.
- [18] L. Zhang, Y. L. Luo, F. Tao, et al. Key technologies for the construction of manufacturing cloud. *Computer Integrated Manufacturing Systems*, vol. 16, no. 11, 2010, pp. 2510-2520.
- [19] L. Zhang, B. P. Zeigler, Y. Laili, Model Engineering for Simulation, *Elsevier*, 2019.3.