An Optimal Production Scheme for Reconfigurable Cloud Manufacturing Service System

Min Wang, Shanchen Pang, Shihang Yu, Sibo Qiao, Xue Zhai, and Hao Yue

***Abstract*—Cloud manufacturing (CMfg) platform consists of the cloud services, manufacturing technology, and the Internet of Things, which provides solutions for large-scale personalized customization through the service model. However, the service flexibility and resource allocation of CMfg are two factors that restrict the production time and cost of CMfg. A CMfg service model based on rewritable Petri nets (RPNs) is established, where the reconfiguration process of personalized customization is described by the rewritable rules of RPN. On this basis, the performance of the reconfiguration of the personalized customization service process is analyzed (this model analysis method can analyze the soundness of the reconfiguration process). In addition, we establish the resource allocation strategy of CMfg based on nondominated sorting genetic algorithm to obtain the best personalized customization scheme in terms of time and cost. The results of simulation and comparison experiments show that the method proposed in this article can obtain the optimal solution for both production time and cost.**

***Index Terms*—Cloud manufacturing (CMfg), personalized customization, resource allocation, rewritable Petri nets (RPN).**

I. Introduction

CLOUD manufacturing (CMfg) is a service-oriented network manufacturing pattern [1], which manages the decentralized, virtualized and service manufacturing resources, manufacturing capabilities and manufacturing knowledge, and provides customers with on-demand manufacturing cloud services [2]. On the CMfg platform, the demand of customer evolves from a single demand to a personalized demand with multidimensional requirements such as product configuration, attributes, quality and service attributes, service time, and so on. The model of CMfg service can combine the dispersed manufacturing resources, which eliminates the limitation that traditional manufacturing services only provide a single service. Another advantage is that the industrial structure is optimized and the usage of resources, in turn, becomes more reasonable and understandable. At the same time, advanced computing technology, virtualization technology, embedded technology, Internet of Things (IoT) technology, and other high-performance technologies can solve the bottleneck problem of manufacturing systems. For example, virtualization technology constructs a virtual manufacturing environment for simulating the resources and manufacturing capacity in the manufacturing system, which can reduce the waste of resources.

CMfg customized service is an intelligent manufacturing mode that integrates customers into the full life cycle process of product customization, design and development, manufacturing, logistics, and service [3], [4]. MindSphere is an IoT operating system based on a cloud platform launched by Siemens, which helps customers complete acquisition, transmission, storage, analysis, and application. This platform provides an open application interface, which helps users select the appropriate application to analyze the operation data of the factory to realize intelligent control of the production process. COSMOplat is an industrial Internet platform researched and developed by Haier industrial. The platform architecture includes a resource layer, platform layer, application layer, and model layer. The platform layer includes seven modules: user interaction customization platform, design and marketing platform, open design platform, purchasing platform, intelligent production platform, smart logistics platform, and intelligent service platform. Users participate in designing, procurement, manufacturing, logistics, and experience of the production process, which forms a trinity of users, enterprises, and resources. These studies designed the architecture of personalized services in CMfg. To improve production efficiency and reduce production costs, most studies focus on the optimization of CMfg resources, and there is still a lack of

Manuscript received 29 January 2022; revised 3 April 2022; accepted 12 April 2022. Date of publication 26 April 2022; date of current version 30 September 2022. This work was supported in part by the Major Science and Technology Innovation Project of Shandong Province under Grant 2019TSLH0214, in part by the Tai Shan Industry Leading Talent Project under Grant tscy20180416, in part by the Fundamental Research Funds for the Central Universities under Grant 20CX05016A, in part by the Major Scientific and Technological Projects of CNPC under Grant ZD2019-183-007, and in part by the graduate innovation projects of China University of Petroleum (East China) under Grant YCX2020097. Paper no. TII-22-0493. *(Min Wang and Shanchen Pang are co-first authors.) (Corresponding author: Shanchen Pang.)*

Min Wang is with the College of Control Science and Engineering, China University of Petroleum (East China), Qingdao 266580, China (e-mail: minwang2121@163.com).

Shanchen Pang, Sibo Qiao, Xue Zhai, and Hao Yue are with the College of Computer Science and Technology, China University of Petroleum (East China), Qingdao 266580, China (e-mail: pangsc@upc.edu.cn; siboqiao@126.com; 2417792534@qq.com; hyue@upc.edu.cn).

Shihang Yu is with the School of Mechanical Engineering, Tiangong University, Tianjin 300387, China (e-mail: shihang_yu@126.com).

Color versions of one or more figures in this article are available at https://doi.org/10.1109/TII.2022.3169979.

Digital Object Identifier 10.1109/TII.2022.3169979

effective solutions for the reconstruction and optimization of production processes.

The production process needs to be modeled to analyze the performance of production systems. The modeling approaches include integration definition for function modeling, which is a design and analysis method of structure-oriented. However, this model does not support the description of processes, nor can it describe the time-order constraint between activities. The event-driven process chain (EPC) modeling method combines the data, events, and other resources in the manufacturing process to create a dynamic model of the system [5]. This method has been applied in resource planning, business reorganization, and workflow management, but these modeling methods have shortcomings in mathematical description and analysis. Amjad *et al.* [6] combined EPC with Petri nets (PNs) to solve the deficiency of EPCs in modeling complex discrete systems, but its modeling complexity is high, and its visibility and understandability are insufficient.

PN is a modeling tool used to describe and analyze the system, with strict mathematical expression and intuitive graphical expression [7]. PN modeling technology has been widely applied in intelligent manufacturing systems, flexible manufacturing units, agile manufacturing units, and automatic production processes. A timed PN is proposed in [8], which is suitable for evaluating the time performance of an actual discrete system. In industrial production, the production of products is usually completed by a combination of multiple processes. A timed PN can model and analyze the performance of production systems. However, the traditional manufacturing system is not satisfied with the needs of customized manufacturing. The traditional manufacturing system needs to be reconstructed. The ability of PN to describe and model the dynamic system is insufficient, so rewritable Petri net (RPN) [9] was proposed to model the dynamic system.

In this article, the main contributions are as follows.

1) The optimal production scheme framework for CMfg customization is designed.
2) The CMfg service model based on RPN is established.
3) The soundness of the CMfg service model is verified.
4) The resource allocation strategy of CMfg based on nondominated sorting genetic algorithm II (NSGA-II) is established.

The rest of this article is organized as follows. Section III introduces the framework of the optimal production scheme for mass customization of CMfg. The optimal production scheme for CMfg based on RPN and NSGA-II is introduced in Section IV. Experiments and results analyses are shown in Section V. Finally, Section VI concludes the article.

II. Related Work

CMfg platform provides a CMfg business model with optimal allocation of cross-enterprise resource capabilities, product data management, and production execution management. CMfg platform is able to organize manufacturing service resources according to user needs and provide on-demand manufacturing services to users. An intelligent manufacturing platform is proposed in [10], which uses 5G edge computing technology to realize data collection, feature extraction, and emergency response and builds a cloud monitoring platform to realize message transmission, data storage, and status visualization in the manufacturing. Zhou *et al.* [11] design a large-scale personalized technical architecture, which digs market demand and predicts user satisfaction to help users make better production decisions. Due to the real-time arrival of personalized orders, the production line needs to be reorganized in real time according to different orders. To achieve rapid personalized customized production, a real-time edge scheduling model based on order-level requirements is proposed [12], which improves user satisfaction and resource utilization. The work [13] proposes a framework of 3-D printing service platform, which can manage and schedule distributed 3-D printing services intelligently on the cloud platform. CMfg services allow users to request manufacturing services on the Internet, which improves the usage of resources and reduces production costs [14].

Due to the individual needs of users, it is required that the manufacturing system can adapt to user needs quickly and achieve high-quality productions. However, CMfg platform has limitations in adapting to market needs, such as low flexibility and high customization costs. Therefore, it is necessary to study the reconstruction optimization of the CMfg system. A dynamic service reconfiguration model is proposed in [15] to address abnormal services. The goal of service reconfiguration is to improve the processing quality and reduce the cost, and the validity of the service reconfiguration model is verified. In order to realize the adaptive scheduling of CMfg system, [16] studied the service mode of a multi manufacturing cloud in a dynamic environment and proposed a new service selection paradigm. A functional block-based integration mechanism for integrating various types of manufacturing equipment is proposed in [17], providing a flexible architecture and adaptive and integration methods for CMfg systems. A task allocation method is proposed in [18] and analyzes in detail the four main factors that affect task allocation in the CMfg environment. Guo *et al.* [19] proposed the concept of resource service composition (RSC) and the idea of optimizing selection based on flexible RSC, so that RSC can adapt to dynamic changes.

CMfg centralizes manufacturing resources scattered in different geographical locations through servers and provides manufacturing services to users on demand. NSGA-II is a multiobjective optimization algorithm that obtains the resource allocation strategies quickly [20]. The optimized configuration of CMfg resources improves the utilization of resources and saves production costs. A new swap-shuffled leap-frog algorithm is proposed in [21] to optimize the four parameters of cost, time, quality, and risk in CMfg, which improves the efficiency of resource scheduling and provides the optimal production scheme. Hu *et al.* [22] take task load, task reliability, manufacturing efficiency, and manufacturing resource quantity as indicators that affect the quality of CMfg services. Additionally, chaotic optimization algorithms are used to optimize these indicators to achieve optimal manufacturer scheduling. A mathematical model is proposed to achieve the optimal combination method of logistics and operators, reducing the total production and logistics costs [23]. A multilayer cloud coordination optimization model that includes public cloud and private cloud is established to solve the problem of logistics service resource allocation and coordination optimization in CMfg [24]. The hybrid artificial bee colony algorithm is proposed to solve the problem of service optimization selection [25], which considers the quality of service and the CMfg environment. The probability model of Kimide copula distribution estimation algorithm and the chaos operator that guides the artificial bee colony to generate optimal individual offspring is to get optimal service.

Modeling the CMfg system can analyze and verify the property and performance of the manufacturing system [26]. A multidimensional information modeling method for manufacturing capabilities in CMfg systems is proposed in [27], which realizes the sharing of manufacturing capabilities in CMfg systems. Ahn *et al.* [28] propose a Markov model to analyze the potential cooperation between manufactures to improve customer satisfaction with customized services. Moreover, Markov chain is used to model the structure of the production line and analyze the throughput rate and steady-state rate of the production system [29], [30]. In addition, using automata to model manufacturing systems to analyze system anomalies and detect faulty equipment in the system is proposed in [31]. Automata is also used to model the dynamic behavior of physical manufacturing resources, and an adaptive control mode for resource allocation in production systems is proposed in [32].

However, these modeling methods cannot visualize the resource process in the system. PN is a method to visualize the operation process and resource flow of a manufacturing system and provide an analysis method of system properties and performance [33]. The work [34] proposes the resource allocation PNs to model resource allocation systems and studies the deadlock problem in resource allocation systems. Zhang *et al.* [35] propose a new reconfigurable control method, which uses PN to model the assembly system and combines the method of integer linear programming to calculate the shortest legal trigger sequence from the state with reconfiguration requirement to the reconfigurable state. Dynamic reconfiguration of the system is achieved within the maximum allowable reconfiguration delay. RPN is an extended PN to make up for the lack of modeling ability of traditional PN for dynamic system reconstruction, and it is suitable for modeling and simulating dynamically changing concurrent systems [36]. Six well-formed reconfigurable stochastic Petri net modules are built in [37], and the reconfigured manufacturing system based on these modules can maintain the properties of the original system, such as activity, boundedness, and reversibility.

The above research results have made great progress in CMfg model, CMfg production architecture, and CMfg production optimization in intelligent factories. The modeling and analysis of CMfg focus on the production system in the factory, but the research on the CMfg dynamic service is limited. Simultaneously, the production optimization of CMfg services focuses on resource optimization, without considering the reconfiguration of the production process to further optimize the CMfg service system (CMSS).

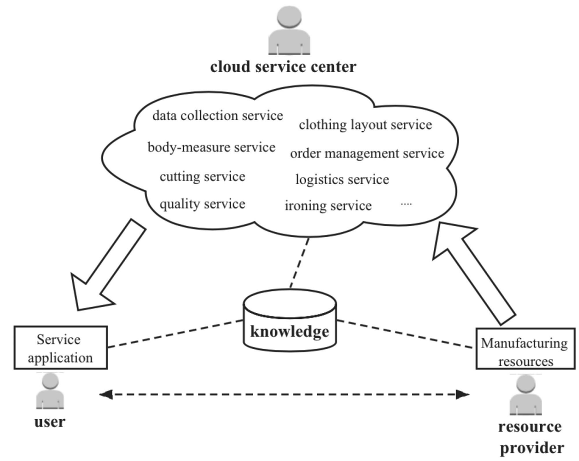

Fig. 1. Framework of CMSS.

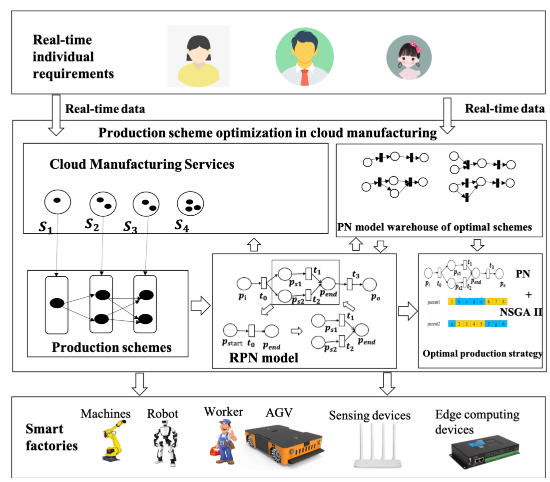

Fig. 2. Optimal production scheme framework for CMfg customization.

III. Proposed Framework

A. CMfg Service System

A CMSS includes three roles: resource provider, cloud service center, and user. Resource providers perceive and virtualize access to manufacturing resources and capabilities and provide the services to cloud service centers. The service center manages and operates cloud services efficiently and provides service for the users according to the user's request. Users use all kinds of services according to their needs with the support of a cloud service center. Knowledge supports virtualized access and service-oriented encapsulation of manufacturing resources and capabilities as well as efficient management and intelligent search of cloud services. Additionally, users also request the services from providers and the providers can provide the services to users directly. The framework of CMSS is shown in Fig. 1.

B. Customized Optimal Production Scheme Framework

As shown in Fig. 2, the optimal production scheme framework for CMfg customization proposed in this article consists of three layers.

Specifically, the first layer is the user layer that collects real-time personalized requirements of users and generates user orders. The CMfg services layer is in the middle layer, which decomposes each order into multiple independent task modules according to the personalized requirements of users. First, the optimal production scheme is found by matching with the PN model warehouse of optimal schemes if the model warehouse has the same product scheme. Furthermore, there are two cases when user requirements change.

Case 1: If no optimal production scheme that produces the same product in the PN model warehouse of optimal schemes while there is an optimal scheme with the same type of product. Next, the RPN is used to model this optimal scheme and determine the services that need to be reconfigured in the scheme. Furthermore, the services to be reconstructed are sent to the CMfg service center to find the matching CMfg services. Finally, we can get an optimal production scheme that meets the new demand.

Case 2: If no production scheme with the same product type in the PN model warehouse, each task of the order will be matched with the service in the CMfg service to generate an optimal production scheme. Further, the newly generated production scheme is modeled and analyzed with PN and the PN model of the production scheme is saved in the PN model warehouse. In the end, NSGA-II is used to optimize the optimal production scheme to get the optimal production strategy.

The bottom of the framework is the intelligent factory, which sends manufacturing tasks to each factory for production according to the optimal production scheme obtained from the CMfg service layer.

IV. Optimal Production Scheme for CMfg Based on RPN and NSGA-II

The optimal production scheme of CMfg consists of three parts: RPN construction of CMfg service model, verification of the soundness of the reconfiguration process, and optimal resource allocation strategy.

A. CMfg Service Model Built by RPN

Definition 1: A four-tuple $S = (P, K, Q, R)$ is a CMfg service where we have the following.

1) P is the provider of CMfg service.
2) K is the knowledge of CMfg service.
3) $Q = \{c_i, t_i, w_i\}$ is the ability of CMfg service. c_i is the cost of service i, t_i is the execution time of a service i, and w_i is the waiting time of a service i.
4) R is the number of each service.

Definition 2: A four-tuple O = (id, type, q, sub) is an order in CMfg where id represents the order number, type represents the order type, q represents the product quantity of the order, and sub represents the subtype of the order.

Definition 3: (Petri nets [7]) A four-tuple $N = (P, T, F, M_0)$ is a prototype PN.

1) P is a finite set of places.
2) T is a finite set of arcs. $P \cap T = \varnothing$.
3) $F \subseteq (P \times T) \cap (T \times P)$ is a finite set of arcs.
4) M_0 is the initial marking. $M{:}P \rightarrow \mathbb{N}$ is a set of markings. $\mathbb{N}$ represents the set of a nonnegative integer. For example, $M_0(p) = 1$ denotes place p has one token at marking M_0.
5) $\forall x \in P \cup T$, ${}^\bullet x = \{y|(y, x) \in F\}$ denotes the preset of a node x. $x^\bullet = \{y|(x, y) \in F\}$ denotes the postset of a node x.

Transition t is enabled when $\forall p \in {}^\bullet t$ and $M(p) \geq 1$, which is denoted as $M\,[t\rangle$.

If transition t fires at marking M, then a new marking M' will generate, which is denoted as $M\,[t\rangle\, M'$. The new marking is as follows:

$$M' = \begin{cases} M(p) - 1 & \text{if } p \in^\bullet t - t^\bullet; \\ M(p) + 1 & \text{if } p \in t^\bullet -^\bullet t; \\ M(p), & \text{other} \end{cases}. \tag{1}$$

Definition 4: (Rewritable Petri nets [9]) A two-tuple $\Sigma = (\Re, \Sigma_0)$ is an RPN where we have the following.

1) $\Re = \{r_1, r_2, .., r_n\}$ is a finite set of rewritable rules.
2) Σ_0 is the initial PN.
3) $r \in \Re$ is a five-tuple $r = (L, R, {}^\bullet\tau, \tau, \tau^\bullet)$ where we have the following.
   a) $L = (P_L, T_L, F_L)$ and $R = (P_R, T_R, F_R)$ are the left-hand and the right-hand sides of r, respectively. L is the full subnet of the Σ_0. R is the rule for replacing L. P_L, T_L, and F_L are the places, transitions, and arcs of the left-hand side, respectively. P_R, T_R, and F_R are the places, transitions, and arcs of the right-hand side, respectively.
   b) $\tau \subseteq (P_L \times P_R) \cup (T_L \times T_R)$ is the transitive relation of r. The left-hand places (transitions) are related to the right-hand places (transitions), that is, $P_L\tau \subseteq P_R$, $T_L\tau \subseteq T_R$, $\tau P_R \subseteq P_L$, $\tau T_R \subseteq T_L$.
   c) ${}^\bullet\tau \subseteq \tau$ and $\tau^\bullet \subseteq \tau$ are the input interface relation and the output interface relation.
   d) Transition t fires at marking M and a new marking will be generated $M' : (\Sigma_0, M) \xrightarrow{t} (\Sigma, M') \Leftrightarrow (\Sigma = \Sigma_0) \wedge M[t\rangle$.
   e) The arc is marked by the rewritable rule $r = (L, R, {}^\bullet\tau, \tau, \tau^\bullet)$; there exists a full embedding $f : L \rightarrow \Sigma_0$ that has $\forall x \notin f(L) \wedge \forall y \in L : x \in^\bullet f(y) \Rightarrow y \in Dom({}^\bullet\tau) \vee x \in f(y)^\bullet \Rightarrow y \in Dom(\tau^\bullet)$.

$P = P_0 - f(P_L) + P_R$, $P_L\tau = P_R$, $T = T_0 - f(T_L + T_R)$, $T_L\tau = T_R$ where + (-) indicates adding (removing) place or transition. The relation of arc F is denoted as follows:

$$F(x,y) = \begin{cases} F_0(x,y), & x \notin R \wedge y \notin R; \\ F_R(x,y), & x \in R \wedge y \in R; \\ \cup_{y_i \in^\bullet \tau y} F(x, f(y_i)), & x \notin R \wedge y \in R; \\ \cup_{x_i \in \tau^\bullet x} F(f(x_i), y), & x \in R \wedge y \notin R \end{cases}. \tag{2}$$

Place $p \in P$, marking $M'(p)$ is as follows:

$$M'(p) = \begin{cases} M_0(p), & p \notin R; \\ M_R(f(p)), & p \in R \end{cases}. \tag{3}$$

Theorem 1: Given an RPN $\Sigma = (\Re, \Sigma_0)$, if Σ_0 is liveness and the right-hand side of rule R is well-structured [38], then the RPN Σ is liveness.

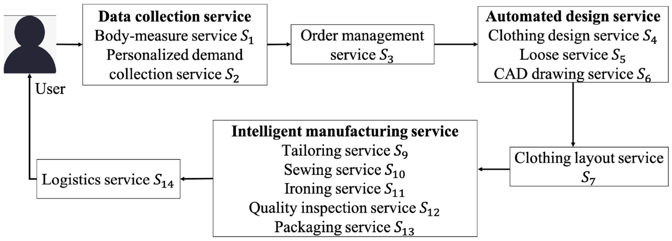

Fig. 3. CMfg process of clothes customization.

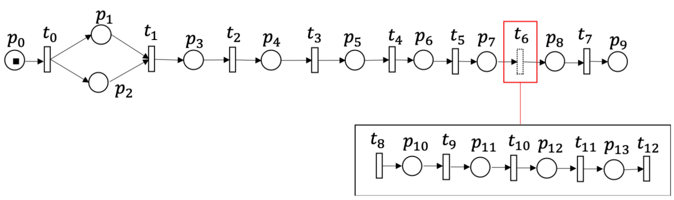

Fig. 4. PN model of CMfg process of clothes customization.

Proof: We discuss two conditions: the interface of rule R is places or transitions. The proof is as follows. ∎

Condition 1. The interface of rule R is places. P_{Ri} and P_{Ro} are the input and output places of the right-hand side of rule R. We suppose that $T^1 = \{{}^\bullet P_{Ri} \cup P_{Ro}^\bullet\}$, and then we deduce that $T^1 \in T_0$. According to the condition that Σ_0 is liveness, it can be deduced that $\forall t \in T_0, \exists M \in R(M_0) : M[t\rangle$. Since the right-hand side of rule R is well-structured, then we can infer that $\forall t \in T_R, \exists M_2 \in R(M_1) : M_2[t\rangle$. Further, according to $T_\Sigma = T_R \cup T^1$, we can conclude that $\forall t \in T_\Sigma, \exists M' \in R(M_0) : M'[t\rangle$. Hence, RPN Σ is liveness.

Condition 2. The interface of rule R is transitions. T_{Ri} and T_{Ro} are the input and output transitions of the right-hand side of rule R. We suppose that $P^1 = \{{}^\bullet T_{Ri} \cup T_{Ro}^\bullet\}, T^1 = \{({}^\bullet P^1 \cup P^{1\bullet}) \cap T_0\}$. According to the condition that Σ_0 is liveness, we can deduce that $\forall t \in T^1, \exists M \in R(M_0) : M[t\rangle$. We assume that $T^2 = \{T_R \backslash (T_{Ri} \cup T_{Ro})\}$. Since the right-hand side of rule R is well structured, then we can infer that $\forall t \in T^2, \exists M_2 \in R(M_1) : M_2[t\rangle$. Further, according to $T_\Sigma = T^1 \cup T^2$, we can conclude that $\forall t \in T_\Sigma, \exists M' \in R(M_0) : M'[t\rangle$. Hence, RPN Σ is liveness.

We take the service of clothes customization as an example to introduce CMSS. First, the data collection service includes the smart body-measure service S_1 and personalized demand collection service S_2. Next, the order is generated by the order management service S_3 and the process enters the automated design services, which include clothes design service S_4, loose service S_5, and CAD drawing service S_6. Then, the clothes enter the clothing layout service S_7. There are five services in the intelligent manufacturing service, namely tailoring service S_9, sewing service S_{10}, ironing service S_{11}, quality inspection service S_{12}, and packaging service S_{13}. Finally, customized clothes will be sent to customers through logistics service S_{14}. The production process is shown in Fig. 3. The PN model of CMfg process is shown in Fig. 4, where the intelligent manufacturing transition t_6 contains a nested subnet of the manufacturing process in factories. Table I gives the meaning of the places and transitions in Fig. 4.

TABLE I
MEANING OF EACH PLACE AND TRANSITION IN FIG. 4

P	Meaning	T	Meaning
p_0	user	t_0	collect data service
p_1	smart body-measure service data	t_1	order management service
p_2	demand collection service data	t_2	clothes design service
p_3	order data	t_3	loose service
p_4	clothes design data	t_4	CAD drawing service
p_5	loose service data	t_5	clothing layout service
p_6	CAD drawing data	t_6	Intelligent manufacturing service
p_7	clothing layout data	t_7	logistics service
p_8	product	t_8	tailoring service
p_9	end	t_9	sewing service
p_{10}	well tailoring cloth	t_{10}	ironing service
p_{11}	well sewing cloth	t_{11}	quality inspection service
p_{12}	well ironing product	t_{12}	packaging service
p_{13}	product after quality inspection		

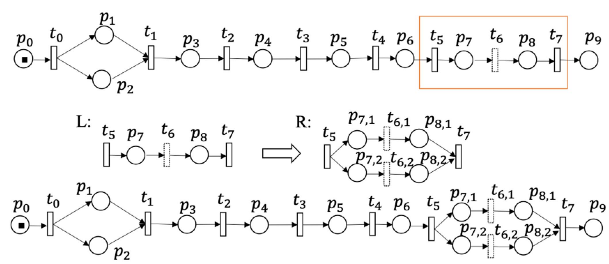

Fig. 5. RPN model of production scheme of reconfigurable clothes customization.

The production line needs to be restructured if the current production line cannot meet the new demand. For example, the current smart manufacturing factory does not meet the demand for new products, and two different factories are required to cooperate to complete the new production task. Therefore, the current production scheme does not meet the new demand, and the production scheme needs to be reconfigured. The reconfiguration process is shown in Fig. 5. The reconfigured scheme added the new needed factory.

B. Soundness of the Reconfiguration Progress

Definition 5: (Soundness [7]) A PN is soundness, when it satisfies all of the following conditions.

1) For every marking M reachable from initial marking M_0, there exists firing sequences σ_1 and σ_2 leading from M to terminal marking M_{end}. Formally, $\forall M (M_0[\sigma_1\rangle M) \Rightarrow M[\sigma_2\rangle M_{\text{end}}$.
2) The terminal marking M_{end} is the only making reachable from initial marking M_0 with at least one token in terminal output place p_O.
3) Each transition is live in PN. That is, $\forall t \in T, \exists M, M' \Rightarrow M_0[\sigma\rangle M[t\rangle M'$.

The reachability graph is the tool used to analyze the properties of PN. Each node is the marking of the system. When a transition sequence or a transition fires, the marking will be changed. The reachable graph of reconfigurable production

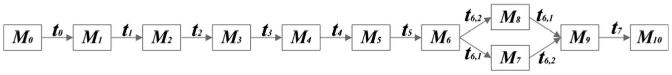

Fig. 6. Reachability graph of RPN model in Fig. 5.

scheme of clothes customization in Fig. 5 is shown in Fig. 6. We can see that the reachability graph of reconfigurable production scheme in Fig. 6 has a unique termination marking M_{10}. Any nontermination marking can reach the termination marking through the transition sequence, and each transition can fire in the reachability graph. Therefore, the reconfigurable production scheme is soundness.

C. Optimal Resource Allocation Strategy

The goal of the optimal allocation strategy is to obtain a production scheme with the shortest production time and the lowest production cost according to the order type, product quantity, and production capacity of each CMfg service. In this article, NSGA-II and RPN are combined to obtain the optimal production strategy.

Production time is the total time required to produce an order, and production cost is the total cost of CMfg services required to produce an order. The mathematical formula of the total production time T_{total} and the total production cost C_{total} is as follows:

$$\text{Minimize } T_{\text{total}} = \sum\nolimits_{j=1}^{m} \sum\nolimits_{i=1}^{n} \sum\nolimits_{k=1}^{K} (t_{jik}+w_{jik}) \quad (4)$$

$$\text{Minimize } C_{\text{total}} = \sum\nolimits_{j=1}^{m} \sum\nolimits_{i=1}^{n} \sum\nolimits_{k=1}^{K} (c_{jik}). \quad (5)$$

In formulas (4) and (5), t_{jik} represents the production time of the *j*th product in the *k*th service provider of the *i*th service. w_{jik} represents the waiting time of the *j*th product in the *k*th service provider of the *i*th service. c_{jik} represents the production cost of the *j*th product in the *k*th service provider of the *i*th service.

The production scheme matching algorithm is shown in Algorithm 1. The process of production scheme matching algorithm is as follows: If the subtype of the order is the same as the subtype of a production scheme in the warehouse, then the same subtype production scheme in the warehouse is the optimal scheme. If there is no production scheme of the same subtype in the warehouse, and there is a production scheme of the same type, the production scheme with the same type is the similar scheme. The similar scheme can be reconfigured to satisfy the new order.

The optimal production scheme algorithm is shown in Algorithm 2. The steps of the optimal production scheme are as follows:

Step 1. According to the subtype of the order, it is first matched with the optimal scheme warehouse. The production scheme will be used if there is a production scheme of the same subtype in the optimal schemes. Since there are multiple products for each order, the NSGA-II is used to obtain the optimal production allocation strategy to maximize the usage of CMfg service resources and reduce the waiting time for order processing.

Algorithm 1: Production Scheme Matching Algorithm.

```
Input: Order sequence O[], Production scheme
       warehouse scheme[]
Output: the scheme matching result
for (j=0; j<count(O[]); j++) do
    for (i=0; i<count(scheme[]); i++) do
        if the type of scheme[i]==the type of O[j] then
            if subtype of scheme[i]==subtype of O[j]
             then
                return scheme[i]
            end
            else
                return similar scheme[i]
            end
        end
        else
            return null
        end
    end
end
```

Step 2. When the demand changes, if there is no production scheme of the same subtype in the optimal scheme warehouse and there is a production scheme of the same type. Next, RPN is used to reconfigure and analyze the production scheme of the same type to obtain an optimal production scheme that adapts to the new product. Next, NSGA-II is used to allocate resources for the reconfigured production scheme.

Step 3. A new production scheme will be generated if there is no production scheme of the same type in the optimal warehouse. PN is used to model and analyze the new production scheme. Meanwhile, the new production scheme will be stored in the optimal scheme warehouse. At the same time, NSGA-II is used to optimize CMfg resources.

V. Experimental Program

A. Experimental Case

The experimental program in this article takes the CMfg process of clothing customization as an example that the specific production process is shown in Fig. 3 in Section IV. The production time and cost of body-measure service S_1 and personalized demand collection service S_2 are ignored. The specific parameter settings of order management service S_3, clothing design service S_4, loose service S_5, CAD drawing service S_6, clothing layout service S_7, and intelligent manufacturing service S_8 are shown in Table II. There is only one available service from each provider. Assume that the produced products include A, B, and C. The production scheme for products A and B is in the warehouse of the optimal production scheme. When the production demand changes to the production of A and C, the production scheme for products A and B needs to be reconstructed. The reconfiguration process is shown in Fig. 7. We can see that the initial PN is liveness and the right-hand rule is well-structured; so the RPN is liveness. The meaning of places

Algorithm 2: Optimal Production Scheme Algorithm.

Input: crossover probability c, mutation probability m, Iteration number G, population size S, Production scheme warehouse *scheme[]*, objection function T_{total}, C_{total}, order sequence O, similar order $\overline{O}$

Output: optimal production scheme *optimal[]*

```
NSGA-II(scheme[O_i])
for (g=0; g<G; g++) do
    Generate the initial population P_j;
    Generate the new population Q_j with crossover
    and iteration;
    P_new=P_j ∪ Q_j;
    Frank=fast-non-dominated-sort(P_new);
    P_{j+1}=∅ and i=1;
    for (|P_{j+1}|+|Frank_i|≤S) do
        crowding-distance-assignment(Frank_i);
        P_{j+1}=P_{j+1} ∪ Frank_i;
        i=i+1;
    end
    Sort(F_i);
    P_{j+1}= P_{j+1} ∪ Frank_i[1:(S-|P_{j+1})|];
    Q_{j+1}=make-new-pop(P_{j+1}); j=j+1;
end
for o_i in O do
    if scheme[o_i] in scheme[] then
        NSGA-II (scheme[o_i]);
    end
    else if scheme[o_i] not in scheme[] and scheme[ō_i]
     in scheme[] then
        Rewrite the scheme[ō_i] by RPN to obtain the
         scheme[o_i];
        NSGA-II (scheme[o_i]);
        Add scheme[o_i] in scheme[];
    end
    else if scheme[o_i] not in scheme[] and scheme[ō_i]
     not in scheme[] then
        Model the scheme[o_i] by RPN;
        NSGA-II (scheme[o_i]);
        Add scheme[o_i] in scheme[];
    end
end
```

and transitions are shown in Table I. $t_{i,j}$ represents the service provided by the jth provider of the service type t_i. For example, $t_{1,1}$ represents the service provided by the first service provider of the order management service.

We use the reachability graph to analyze the production scheme of products A and C. Reachability graph is shown in Fig. 8. Each transition is live and each nonterminal marking M can reach the terminal marking M_{end}. Therefore, the reconfigurable production scheme of products A and C is soundness.

TABLE II
PARAMETERS OF EXPERIMENTATION

Cloud Services	Provider	A time	A cost	B time	B cost	C time	C cost
S_3	1	1	1	1	1	1	1
	2	0.8	1.2	0.8	1.2	0.8	1.2
	3	1.1	0.9	1.1	0.9	1.1	0.9
S_4	1	-	-	30	5	35	6
	2	10	12	-	-	28	8
	3	-	-	-	-	26	6
	4	15	10	35	4	30	7
S_5	1	3	1	3	1	3	1
	2	2	1.5	2	1.5	2	1.5
S_6	1	25	2	25	2	25	2
	2	28	1.6	28	1.6	28	1.6
	3	22	2.4	22	2.4	22	2.4
S_7	1	6	2	6	2	6	2
	2	8	1.6	8	1.6	8	1.6
S_8	1	50	15	40	20	-	-
	2	45	18	-	-	45	22
	3	-	-	35	24	50	20
	4	-	-	-	-	40	15

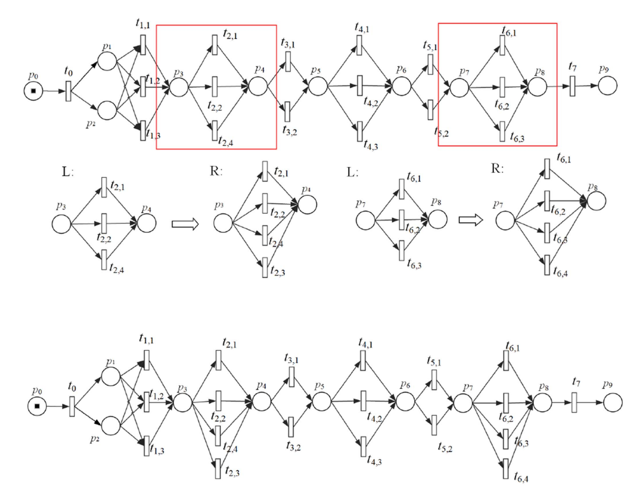

Fig. 7. RPN model for production scheme of products A and C.

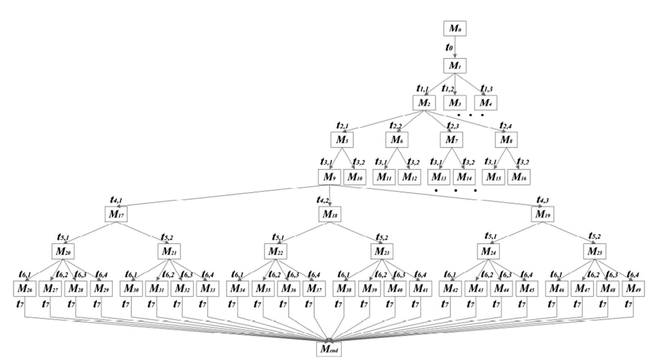

Fig. 8. Reachability graph of RPN model in Fig. 7.

B. Contrast Experiments

To verify the effectiveness of the proposed method, we conducted two experiments. The iteration is 100, and the crossover probability and the mutation probability are all 100%.

The first experiment compares the shortest production time and lowest production cost between production schemes of pre-reconfiguring and post-reconfiguring where the number of product A is fixed to 50 and C varies with the number of products in one order. Under different product quantities in one number: Fig. 9(a) compares the shortest total production time by NSGA-II; Fig. 9(b) compares the minimum total production cost

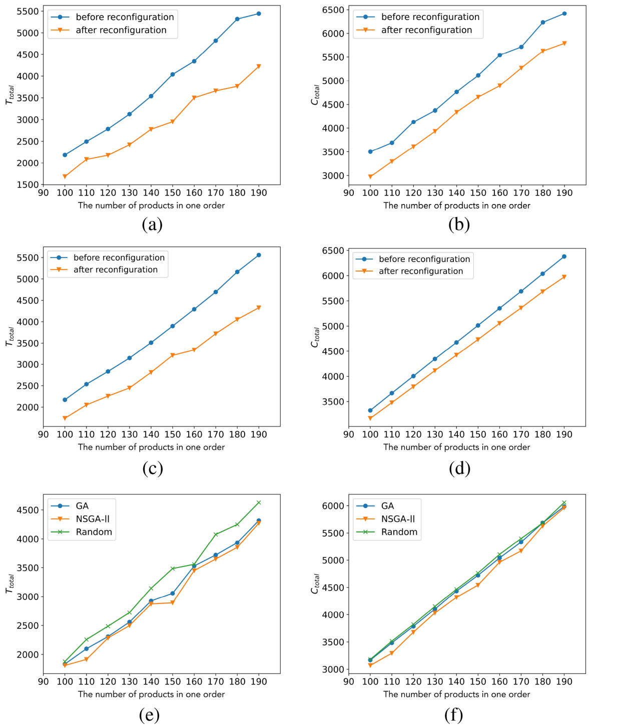

Fig. 9. Results of shortest production time and lowest cost with different number of product quantities in each order. (a) Optimal time by NSGA-II. (b) Optimal cost by NSGA-II. (c) Optimal time by GA. (d) Optimal cost by GA. (e) Optimal time by different algorithms after reconfiguration. (f) Optimal cost by different algorithms after reconfiguration.

TABLE III
RATIO OF PRODUCT A AND C

Order id	1	2	3	4	5	6	7	8	9
A:C	1:9	2:8	3:7	4:6	5:5	6:4	7:3	8:2	9:1

by NSGA-II; Fig. 9(d) compares the shortest production time by GA; Fig. 9(d) compares the minimum total production cost by GA. These two algorithms verified that minimum production time and production cost after reconfiguration are significantly lower than those before reconfiguration. In addition, Fig. 9(e) and (f) compare the shortest production time and lowest cost between three different resource allocation algorithms, NSGA-II, GA, and random strategy, respectively. The experimental results show that for the reconfigured production scheme, NSGA-II can achieve the resource allocation strategy with the lowest total production time and total production cost.

The second experiment is repeated on the similar process as experiment one, in which the ratio of product A to C is constantly changing under total amount of 100 for their sum. The ratio of product A and C is shown in Table III. Under different product ratios of A and C: Fig. 10(a) compares the shortest production time by NSGA-II; Fig. 10(b) compares the minimum production cost by NSGA-II; Fig. 10(c) compares the shortest total production time by GA; Fig. 10(d) compares the minimum total production cost by GA. These two experiments show that the minimum production time and production cost after reconfiguration are lower than those before reconfiguration.

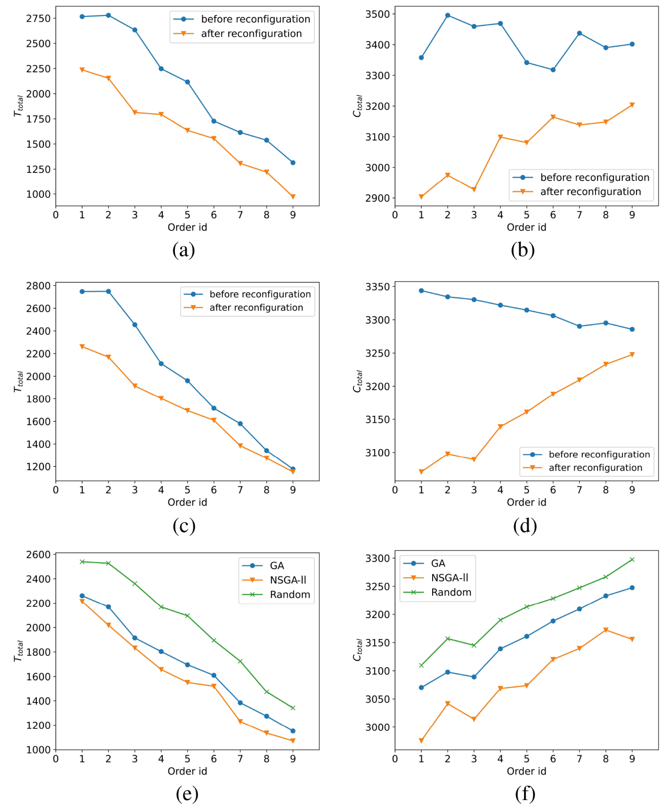

Fig. 10. Results of shortest production time and lowest cost with different ratios of products A and C. (a) Optimal time by NSGA-II. (b) Optimal time by NSGA-II. (c) Optimal time by GA. (d) Optimal cost by GA. (e) Optimal time by different algorithms after reconfiguration. (f) Optimal cost by different algorithms after reconfiguration.

Meanwhile, when product C takes up more ratio, the minimum production time and cost of reconfigured production scheme will be lower than the production scheme before reconstruction. The reason for this situation is that the production scheme before reconfiguration is to produce products A and B. Among them, the clothing design services for products A and B are S_{41}, S_{42}, and S_{44} and the manufacturing services for products A and B are S_{81}, S_{82}, and S_{83}. When products A and C have to be produced, the current production line can also produce products A and C, but the time and cost are higher. After reconfiguration, the clothing design service S_{43} and the manufacturing service S_{84} for producing product C are added to the production line. Therefore, the new production scheme can produce product C more efficiently. Fig. 10(d) and (e) compare the shortest production time and lowest cost between three different resource allocation algorithms, NSGA-II, GA, and random strategy, respectively. The experimental results show that, for the reconfigured production scheme with different ratios of A and C, NSGA-II is significantly better than the other two resource allocation algorithms. This experiment proves that regardless of the proportion of product C, the production time and cost of products A and C are reduced in the reconfigured production scheme.

We set two sets of different parameters to verify the method proposed in this article. Two different parameters are shown

TABLE IV
TWO SETS OF DIFFERENT PARAMETERS OF EXPERIMENTATION

Cloud Services	Pro-vider	A		B		C	
		time	cost	time	cost	time	cost
S_3	1	1/1	1/1	1/1	1/1	1/1	1/1
	2	0.1/1	2/1.5	0.1/1	2/1.5	0.1/1	2/1.5
	3	1/1	3/2	1/1	3/2	1/1	3/2
S_4	1	-	-	30/ 25	27/15	10/38	20/29
	2	30/10	40/15	-	-	15/30	20/30
	3	20/-	45/-	-/-	-/-	12/29	18/40
	4	15/13	30/10	25/28	25/10	18/40	15/25
S_5	1	2/4	5/8	3/4	5/8	5/4	2/8
	2	3/7	4/5	3/7	4/5	4/7	3/5
S_6	1	20/20	20/15	30/20	15/15	18/20	20/15
	2	25/38	20/10	30/38	10/10	20/38	15/10
	3	30/16	15/20	30/16	25/20	30/16	15/20
S_7	1	5/6	2/4	5/6	2/4	5/6	2/4
	2	3/7	5/3	3/7	5/3	3/7	5/3
S_8	1	60/30	50/20	50/40	40/18	-	-
	2	55/22	45/15	-	-	35/20	30/18
	3	-	-	48/30	40/20	30/33	28/25
	4	-	-	-	-	39/30	30/28

TABLE V
PRODUCTION TIME AND COST BEFORE AND AFTER RECONFIGURATION WITH DIFFERENT SETS OF PARAMETERS

Parameters	time/s before RPN	time/s after RPN	cost before RPN	cost after RPN
Table II	2289.3	1677.2	3415.8	3077.5
The left side of / in Table IV	2011	1786	8970	8967
The right side of / Table IV	1811	1491	6623.5	6399.5

in Table IV. The left side of / is a set of parameters, and the right side of / is another set of parameters. When the number of products A and C are both 50, the experimental results are shown in Table V. The results show that the production time and cost of the reconfigured production scheme based on our proposed method are significantly reduced than before reconfiguring.

VI. CONCLUSION

The main contributions of this article include proposing an optimal production strategy framework for CMfg. This method obtained the optimal CMfg production strategy quickly according to the real-time needs of users. Specifically, RPN modeled the reconstructed production process and analyzed the soundness of the reconstructed production process when the production demand changes. Then, NSGA-II was used to optimize the resource allocation strategy to minimize the total production time and total production cost. However, a limitation of this study is that we did not adopt new or improved existing resource allocation algorithms to optimize the resource allocation scheme.

In future work, the new resource allocation algorithms need to be carried out to combine with the RPN to access the optimal resource allocation scheme. In addition, this article studies the reconstruction of the production process and does not study the equipment reconstruction of the CMfg. Therefore, we intend to study the modeling method of the equipment reconstruction and study the well-structured and soundness analysis methods of the reconfigured system.

REFERENCES

[1] W. B. Wang, Y. F. Zhang, J. N. Gu, and J. Wang, “A proactive manufacturing resources assignment method based on production performance prediction for the smart factory,” *IEEE Trans. Ind. Informat.*, vol. 18, no. 1, pp. 46–55, Jan. 2021.

[2] F. Tao, L. Zhang, Y. K. Liu, Y. Cheng, L. H. Wang, and X. Xu, “Manufacturing service management in cloud manufacturing: Overview and future research directions,” *J. Manuf. Sci. Eng.*, vol. 137 no. 4, pp. 040912-1–040912-11, 2015.

[3] Y. P. Xu, G. X. Chen, and J. Zheng, “An integrated solution-KAGFM for mass customization in customer-oriented product design under cloud manufacturing environment,” *Int. J. Adv. Manuf. Technol.*, vol. 84, no. 1–4, pp. 85–101, 2016.

[4] Y. Q. Lu and X. Xu, “Cloud-based manufacturing equipment and big data analytics to enable on-demand manufacturing services,” *Robot. Comput.-Integr. Manuf.*, vol. 57, pp. 92–102, 2019.

[5] X. G. Miao, X. B. Ze, Y. B. Qu, and M. J. Zhang, “Research on the application of EPC encoding technology for plastic windows manufacturing,” in *Proc. 5th Int. Conf. Mech., Ind., Manuf. Technol.*, 2014, pp. 1527–1531.

[6] A. Amjad, F. Azam, M. W. Anwar, and W. H. Butt, “Verification of event-driven process chain with timed automata and time Petri nets,” in *Proc. 9th IEEE-GCC Conf. Exhib.*, 2017, pp. 1–6.

[7] T. Murata, “Petri nets: Properties, analysis and applications,” *Proc. IEEE*, vol. 77, no. 4, pp. 541–580, Apr. 1989.

[8] W. M. Zuberek, “Timed petri nets and preliminary performance evaluation,” in *Proc. 7th Annu. Symp. Comput. Architecture*, 1980, pp. 88–96.

[9] S. C. Pang and C. Lin, “Rewritable petri nets: Rewritable place and properties analysis,” *Chin. J. Comput.*, vol. 35, no. 10, pp. 2182–2193, 2012.

[10] H. B. Yang, Z. Sun, G. D. Jiang, F. Zhao, X. J. Lu, and X. S. Mei, “Cloud-manufacturing-based condition monitoring platform with 5g and standard information model,” *IEEE Internet Things J.*, vol. 8, no. 8, pp. 6940–6948, Apr. 2021.

[11] F. Zhou, Y. J. Ji, and R. J. Jiao, “Affective and cognitive design for mass personalization: Status and prospect,” *J. Intell. Manuf.*, vol. 24, no. 5, pp. 1047–1069, 2013.

[12] Y. Luo, W. F. Li, W. C. Yang, and G. Frtino, “A real-time edge scheduling and adjustment framework for highly customizable factories,” *IEEE Trans. Ind. Informat.*, vol. 17, no. 8, pp. 5625–5634, Aug. 2021.

[13] J. G. Mai, L. Zhang, F. Tao, and L. Ren, “Customized production based on distributed 3D printing services in cloud manufacturing,” *Int. J. Adv. Manuf. Technol.*, vol. 84, no. 1–4, pp. 71–83, 2016.

[14] W. X. Wang and F. Liu, “The research of cloud manufacturing resource discovery mechanism,” in *Proc. 7th Int. Conf. Comput. Sci. Educ.*, 2012, pp. 188–191.

[15] Y. K. Wang, S. L. Wang, L. Kang, and S. B. Wang, “An effective dynamic service composition reconfiguration approach when service exceptions occur in real-life cloud manufacturing,” *Robot. Comput.- Integr. Manuf.*, vol. 71, 2021, Art. no. 102143.

[16] C. Yang, T. Peng, S. L. Lan, W. M. Shen, and L. H. Wang, “Towards IoT-enabled dynamic service optimal selection in multiple manufacturing clouds,” *J. Manuf. Syst.*, vol. 56, pp. 213–226, 2020.

[17] X. V. Wang and L. H. Wang, “Function block-based integration mechanisms for adaptive and flexible cloud manufacturing,” in *Proc. Int. Manuf. Sci. Eng. Conf.*, 2015, pp. 1–8.

[18] M. W. Wang, J. T. Zhou, and S. K. Jing, “On-demand task assignment strategies for workflow-based applications in cloud manufacturing,” *J. Comput.-Aided Desigh Comput. Graph.*, vol. 24, no. 3, pp. 308–313, 2012.

[19] H. Guo, L. Zhang, F. Tao, L. Ren, and Y. L. Luo, “Research on the measurement method of flexibility of resource service composition in cloud manufacturing,” *Adv. Mater. Res.*, vol. 139, pp. 1451–1454, 2010.

[20] K. Deb, A. Pratap, S. Agarwal, and T. Meyarivan, “A fast and elitist multiobjective genetic algorithm: NSGA-II,” *IEEE Trans. Evol. Comput.*, vol. 6, no. 2, pp. 182–197, Apr. 2002.

[21] S. L. Wang, Z. Q. Zhu, and L. Kang, “Resource allocation model in cloud manufacturing,” *Proc. Inst. Mech. Engineers Part C-J. Mech. Eng. Sci.*, vol. 230, no. 10, pp. 1726–1741, 2016.
[22] Y. Hu, F. Zhu, L. Zang, Y. Lui, and Z. Wang, “Scheduling of manufacturers based on chaos optimization algorithm in cloud manufacturing,” *Robot. Comput.- Integr. Manuf.*, vol. 58, pp. 13–20, 2019.
[23] E. Aghamohammathadeh, M. Malek, and O. F. Valilai, “A novel model for optimisation of logistics and manufacturing operation service composition in cloud manufacturing system focusing on cloud-entropy,” *Int. J. Prod. Res.*, vol. 58 no. 7, pp. 1987–2015, 2020.
[24] H. Wang, Y. T. Song, R. Zhang, and Y. Li, “Multilevel coordinate optimization of cloud manufacturing,” in *Proc. IOP Conf. Ser.: Mater. Sci. Eng.*, 2018, Art. no. 012145.
[25] J. Zhou and X. Yao, “A hybrid artificial bee colony algorithm for optimal selection of QoS-based cloud manufacturing service composition,” *Int. J. Adv. Manuf. Technol.*, vol. 88, no. 9–12, pp. 3371–3387, 2017.
[26] Q. Zhang, C. Zhou, Y. C. Tian, N. Xiong, Y. Qin, and B. Hu, “A fuzzy probability Bayesian network approach for dynamic cybersecurity risk assessment in industrial control systems,” *IEEE Trans. Ind. Informat.*, vol. 14, no. 6, pp. 2497–2506, Jun. 2018.
[27] Y. Luo, Z. Lin, and T. Fei, “A modeling and description method of multidimensional information for manufacturing capability in cloud manufacturing system,” *Int. J. Adv. Manuf. Technol.*, vol. 69, no. 5–8, pp. 961–975, 2013.
[28] G. Ahn and S. Hur, “Dynamic estimation model for collaboration potential in cloud manufacturing based on Markov random fields,” *Ind. Eng. Manage. Syst.*, vol. 19, no. 2, pp. 301–307, 2020.
[29] K. Park and J. Li, “Improving productivity of a multi-product machining line at a motorcycle manufacturing plant,” *Int. J. Prod. Res.*, vol. 57, no. 2, pp. 470–487, 2019.
[30] X. Gu, “Modeling of reconfigurable manufacturing system architecture with geometric machines and in-stage gantries,” *J. Manuf. Syst.*, vol. 62, pp. 102–113, 2022.
[31] Z. Guo, Y. Zhang, X. Zhao, and X. Song, “CPS-based self-adaptive collaborative control for smart production-logistics systems,” *IEEE Trans. Cybern.*, vol. 51, no. 1, pp. 188–198, Jan. 2021.
[32] G. Ghosh, G. Wang, and J. Lee, “A novel automata and neural network based fault diagnosis system for PLC controlled manufacturing systems,” *Comput. Ind. Eng.*, vol. 139, 2021, Art. no. 106188.
[33] G. J. Liu, “Pspace-completeness of the soundness problem of safe asymmetric-choice workflow nets,” in *Proc. 41st Int. Conf. Appl. Theory Petri Nets Concurrency*,” 2020, Art. no. 196–216.
[34] G. J. Liu, “Complexity of the deadlock problem for petri nets modelling resource allocation systems,” *Inf. Sci.*, vol. 363, pp. 190–197, 2016.
[35] J. F. Zhang, H. Y. Li, and Z. W. Li, “Reconfiguration control of dynamic reconfigurable discrete event systems based on NCESs,” *IEEE Trans. Control Syst. Technol.*, vol. 28, no. 3, pp. 857–868, May 2020.
[36] M. Llorens and J. Oliver, “Structural and dynamic changes in concurrent systems: Reconfigurable petri nets,” *IEEE Trans. Comput.*, vol. 53, no. 9, pp. 1147–1158, Sep. 2004.
[37] S. Tigane, K. Kahloul, and S. Bourekkache, “Reconfigurable stochastic petri nets for reconfigurable manufacturing systems,” in *Proc. Int. Workshop Serv. Orientation Holonic Multi-Agent Manuf.*, 2017, pp. 383–391.
[38] W. M. Van Der Aalst, “Workflow verification: Finding control-flow errors using petri-net-based techniques,” in *Business Process Management*, vol. 1806, W. M. P. Aalst *et al.* Eds. Berlin, Germany: Springer, 2000, pp. 162–183.

Min Wang received the M.S. degree in software engineering in 2019 from the China University of Petroleum, Qingdao, China, where she is currently working toward the Ph.D. degree in control science and engineering.

Her research interests include theory and application of Petri net, trusted computing, and deep learning.

Shanchen Pang received the Ph.D. degree in computer software and theory from the Tongji University, Shanghai, China, in 2008.

From 2009 to 2011, he was a Postdoctoral Researcher with Tsinghua University, Beijing, China. He is currently a Professor and Ph.D. Supervisor with the China University of Petroleum, Qingdao, China. He has authored or coauthored more than 80 research papers in prestigious journals, including IEEE TRANSACTIONS ON INDUSTRIAL INFORMATICS, IEEE TRANSACTIONS ON SYSTEMS, MAN, AND CYBERNETICS: Systems, *Information Science*, IEEE JOURNAL OF BIOMEDICAL AND HEALTH INFORMATICS, and *Future Generation Computer Systems*. His research interests include (advanced) Petri net theory and applications, formal methods, model checking, artificial intelligence, distributed concurrent system analysis and verification, and other fields of teaching and research work.

Shihang Yu received the M.S. degree in software engineering from the China University of Petroleum (East China), Qingdao, China, in 2020. He is currently working toward the Ph.D. degree in mechanical engineering with the School of Mechanical Engineering, Tiangong University, Tianjin, China.

His research interests include transfer leaning and fault diagnosis.

Sibo Qiao received the M.S. degree in software engineering in 2020 from the China University of Petroleum, Qingdao, China, where he is currently working toward the Ph.D. degree in computer technology and resource information engineering.

His research interests include data mining, deep learning, and machine learning.

Xue Zhai received the B.S. degree in network engineering from the Qingdao University of Technology, Qingdao, China, in 2018. She is currently working toward the Ph.D. degree in computer technology and resource information engineering with the China University of Petroleum, Qingdao.

Her research interests include Petri net, blockchain, and microservices.

Hao Yue received the Ph.D. degree in computer software and theory from the Shandong University of Science and Technology, Shandong, China, in 2009.

From 2011 to 2013, he was a Postdoctoral Researcher with the State Key Laboratory for Manufacturing Systems Engineering, Xi’an Jiaotong University, Xi’an, China. He was also a Postdoctoral Researcher with the State Key Laboratory of Industrial Control Technology, Zhejiang University, Hangzhou, China, from 2015 to 2019. He is currently an Associate Professor with the China University of Petroleum, Qingdao, China. His research interests include logistics systems engineering, discrete event systems and their supervisory control techniques, Petri nets, and automated manufacturing systems.