

Research Article

Study on Resource Configuration on Cloud Manufacturing

Yanlong Cao,^{1,2} Zijian Wu,¹ Qijian Zhao,¹ Huiwen Yan,¹ and Jiangxin Yang²

¹The State Key Laboratory of Fluid Power Transmission and Control, School of Mechanical Engineering, Zhejiang University, Hangzhou 310027, China

²Key Laboratory of Advanced Manufacturing Technology of Zhejiang Province, School of Mechanical Engineering, Zhejiang University, Hangzhou 310027, China

Correspondence should be addressed to Jiangxin Yang; yangjx@zju.edu.cn

Received 27 March 2015; Revised 15 July 2015; Accepted 2 August 2015

Academic Editor: Lihui Wang

Copyright © 2015 Yanlong Cao et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

The purpose of manufacturing is to realize the requirement of customer. In manufacturing process of cloud system, there exist a lot of resource services which have similar functional characteristics to realize the requirement. It makes the manufacturing process more diverse. To develop the quality and reduce cost, a resource configuration model on cloud-manufacturing platform is put forward in this paper. According to the generalized six-point location principle, a growth design from the requirement of customers to entities with geometric constraints is proposed. By the requirement growing up to product, a configuration process is used to match the entities with the instances which the resources in the database could supply. Different from most existing studies, this paper studies the tolerance design with multiple candidate resource suppliers on cloud manufacturing to make the market play a two-level game considering the benefit of customers and the profit of resources to give an optimal result. A numerical case study is used to illustrate the proposed model and configuration process. The performance and advantage of the proposed method are discussed at the end.

1. Introduction

Since the term cyber-physical systems (CPS) was coined in the USA in 2006 [1], the concept of cyber-physical system has had great influences on the revolution and development of the manufacturing industry which has taken a chance to bridge the cyber-world of computing and communications with the physical world [2]. Cyber-physical systems are integrations of computation and physical processes. Embedded computers and networks monitor and control the physical processes, usually with feedback loops where physical processes are affected. As the concept of cloud manufacturing was put forward in 2010, cloud manufacturing implies an integrated cyber-physical system that can provide on-demand manufacturing services, digitally and physically, at the best utilization of manufacturing resources [3]. Cloud manufacturing, called CMfg for short below, focuses on the good service as well as high efficiency and low energy consumption [4, 5]. CMfg is a new service-oriented grid manufacturing mode with the integration of cloud-computing [6] and Internet of Things (IOT) [7] to make the manufacturing resource

virtualized with service-oriented tendency. There are many attempts in the manufacturing areas to take advantage of the Internet technology in the recent years. Typical examples are computer-integrated manufacturing (CIM), agile manufacturing (AM) [8], networked manufacturing (NM) [9], virtual manufacturing (VM) [10], global manufacturing [11], application service provider (ASP), collaborative manufacturing network, lean manufacturing [12], digital manufacturing [13], industrial product-service system (IPS²) [14], manufacturing grid (MGrid) [15], and so on. Although the networked manufacturing, application service provider (ASP), and manufacturing grid have the same problem of how to select the optimal resources chain, there are differences among them. The networked manufacturing is a self-contained system with certain resources and fixed-function solution, which is static and lacks intelligent strategies. So the resources in the chain of networked manufacturing are static and fixed up. ASP focuses on the rental mode of teleservice [16] but lacks intelligent strategies and data security. Manufacturing grid is integration of distributed manufacturing resources with data mining and optimal allocation [17]. The manufacturing mode only takes

the profit of customer into account. Cloud manufacturing concerns not only the integration of distributed resources, but also the distributed service of concentrated resources. CMfg could dispatch and schedule the resources and manufacturing requirement considering the benefit in both sides service suppliers and service demanders, of which the model and algorithm would be discussed in the rest of this paper.

A services encapsulation and virtualization access technique [18] for manufacturing machine provides the real-time, accurate, value-added, and useful manufacturing information, which satisfies the new requirement of CMfg. Virtual resource pool, the so-called manufacturing cloud, organizes the resource to provide service on demand. Users could dynamically add or detract a resource anytime and anywhere according to their requirements. Currently most research focuses on the comparison between CMfg and the existing manufacturing [19], semantic modeling [20], concept and framework building [21], net modeling of manufacturing enterprises [22, 23], resource selection strategy [24, 25], integration of relevant technology [26, 27], knowledge management of manufacturing resources [28], and so on.

A product configuration process is used for reference to describe the whole life cycle of manufacture-product in this paper. Usually, configurable products are assembled from predefined, parameterizable, and interconnectable components where the set of legal parameter constellations and component connections is bound by a set of technical or marketing related constraints [29]. It is a solution to realize the customized mass production at a reasonable cost [30]. By configuring the structure and allocating the resources in the design stage, it is a new breakthrough to take full advantage of the CMfg platform to realize manufacturing automation on the Internet. The knowledge of configuration is the key solution to the organization and management which could have a substantial influence on efficiency and accuracy of cloud platform [31, 32]. Configuration on rules [33], on instances [34], on constraints [35], on ontology [36], and so on is the existing expression of the configuration knowledge in the recent years. Over the last decades, several techniques from the field of artificial intelligence have therefore been applied to product configuration problems, starting with early 1980s rule-based techniques, over logic-based and resource-based approaches to more recent constraint-based solutions and preference programming. Most of the existing approaches have the assumption that there is a simple knowledge base so that the configuration systems are the central elements in any mass-customization strategy. However, the decentralized manufacturing mode on the CMfg platform integrates a wealth of knowledge as well as advanced interfirm modularity [37] and multimarket calls for a more advanced product configuration process.

Similarly, also new challenges arise for the product configuration on the CMfg platform. The disperse resource makes the assembly more sensitive to the requirements of the customers. The purpose of precision as well as efficiency and acceptable price raises the demand for advanced solution of tolerance design. Reusing the concept of product configuration by module design is not enough for the CMfg; it is necessary to expand the function by taking the resource

and the benefit of suppliers into account. After Narahari et al. [38] set up the function-assembly-behavior model, many researchers studied the function-structure mapping relationship [39]. To avoid waste and conflict of resource in the subsequent manufacturing process in the scenario of CMfg, the factor of resource chain needs more attention.

According to the characteristic of the CMfg and product configuration, modeling the configuration process from customer's requirements to the assembly of the product and optimizing the resource chain result based on the constraint of tolerance are what we discuss in this paper.

In Section 2, CMfg resource configuration framework and some definitions are given; a CMfg configuration model based on structure growth design and tolerance design using game theory is put forward in Section 3; in Section 4, a simulation example is presented to illustrate the whole process of the configuration discussed in this paper.

2. CMfg Resource Configuration Model

CMfg resource configuration model is to build a relationship among the requirements of the customers, the structures of products, and the resources of the service suppliers. By bridging the communication gap from the requirement to the market, the CMfg platform could realize the aim of TQCSEFK (i.e., fastest time to market, highest quality, lowest cost, best service, cleanest environment, greatest flexibility, and high knowledge) for manufacturing enterprise.

It is assumed that configuration requirements of resource demander and a set of resources on cloud as well as the optimizing standards are given; the target is to obtain a configuration result under these conditions (the result is a subset of resources which could be null).

At the same time, the process must satisfy some limiting conditions: the model does not produce any new resources; the geometric constraints are fixed; the results are not only the resource chain but also the space-time or other constraints between them.

The process of configuration on CMfg based on the growth design and game theory is shown in Figure 1.

In the first part of the configuration process, the requirements of the customers would be materialized by transferring the function to the functional surface. The initial surface grows up into an entity or a specific component which could undertake a certain function and some necessary tolerance or local dimension chain must be attached. After several modules fulfilling all the requirements of the customer, the product has been created to the different possibilities. Different choice of entity gives the design stage different direction to create the required product. The functional surfaces, the components, the module, or the whole product could be called configuration units. The process from the requirement to the entity is called the growth design process.

In the second part, while creating the surfaces, entities, modules, or products, the geometric constraints are attached to geometries. After the information extraction of the instances from the resource database in the third part, the geometries have been matched in the database. In the matching process, if any condition could not be satisfied,

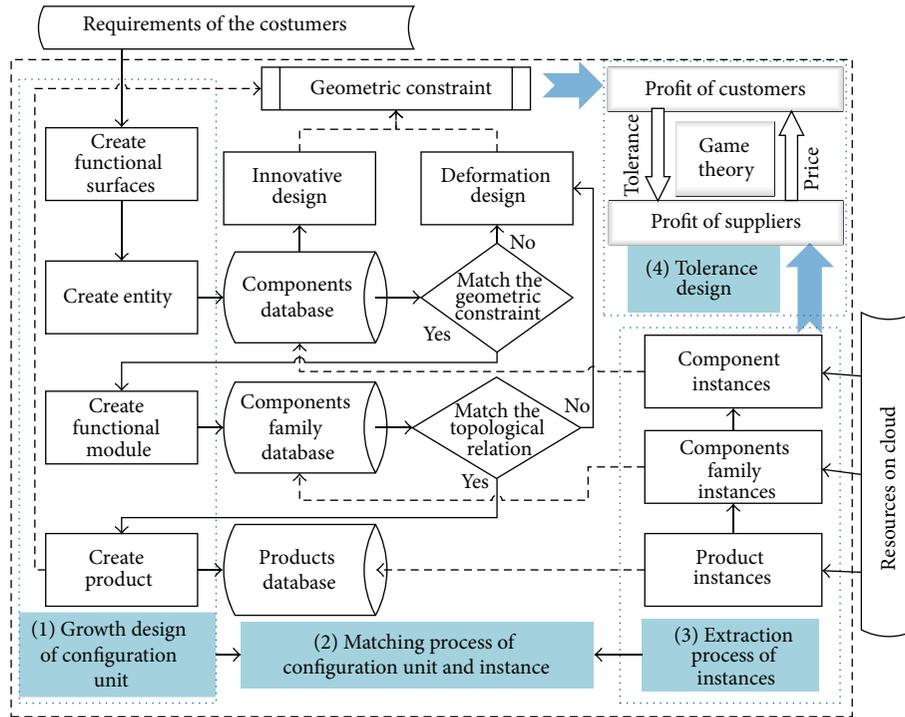


FIGURE 1: The process of configuration on CMfg.

innovative design or deformation design should be invoked by calling design service as soft resource.

In the fourth part, the resource information gives the reference to optimize the structure according to the cost and quality in the tolerance design stage. The tolerance design in this paper is modeled as a two-level game considering the profit of customers and resource suppliers. The game theory makes the market optimize itself to maximize the welfare of both sides.

When reusing the instant, invoking a component family is more efficient than invoking every single component. The concept of customized product and component family are important when discussing the configuration process in the rest of this paper.

Component family is a set of components which has the following characteristics:

- (i) It has the same market positioning.
- (ii) It satisfies the same requirements.
- (iii) It has a similar construction.
- (iv) It has a similar function.
- (v) It has the same external interface.
- (vi) It could be assembled to a product according to parameter given by the platform.
- (vii) It could describe the geometrical and functional relationship with the other parts.

There are three key issues in the configuration process of CMfg: building the relationship from the requirements of

customer to the structures, attaching the candidate resources according to the structures, and selecting the optimal structure and resource to maximize the interests of both customers and service suppliers. These three issues would be discussed in Section 3.

3. CMfg Resource Configuration Process

3.1. *The Growth Design in CMfg Scenario.* How to organize the manufacturing resource is a pivotal problem in the research of CMfg platform. A growth design based on function-construction-resource is proposed in this section to solve the resource organization.

The process of growth design is shown in Figure 2(a) [40]. A set of functional surfaces are materialized from the requirements given by the customers and then the surfaces grow up into certain executive components which could carry out a task. According to the six-point location principle, when forming a surface, the six degrees of freedom should be fixed up to locate the surface in space. The generalized six-point location principle assumes that every component of the product must be located in the four dimensions by eliminating twelve degrees of freedom.

That is, three pairs of translational degrees and three pairs of rotational degrees are eliminated by giving the static position or every position in the trajectory. As for a product, not all of the components are static. The kinematic pair would be regarded as the generalized location while determinate movement traits are generalized location as the location of surface or component is dynamically explicit. The concept of location is expanded by determined movement trajectories.

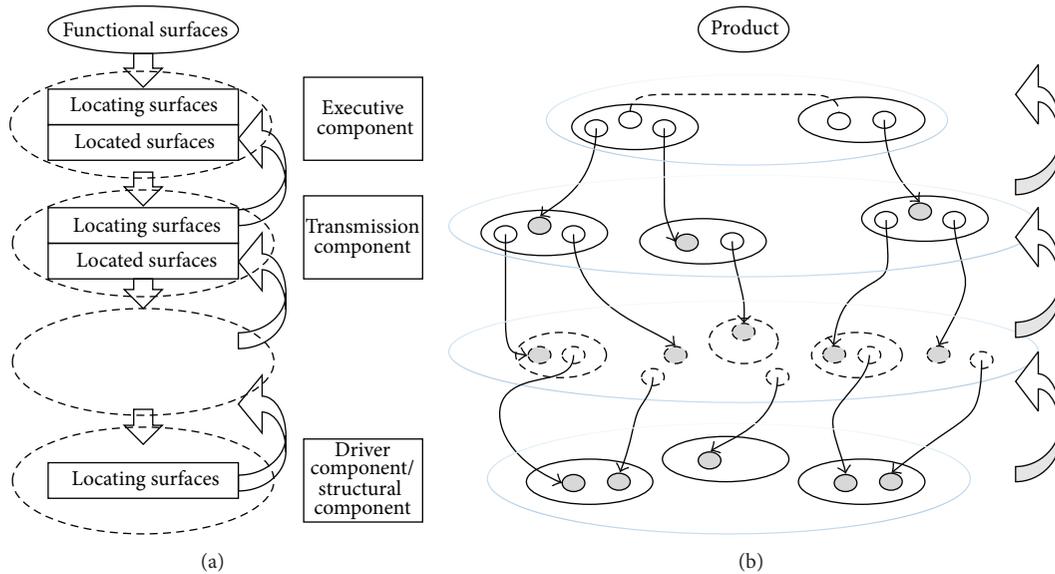


FIGURE 2: The process of incremental growth design from requirements to constructor.

According to different function, there are four kinds of components in the product: executive components, driving components, transmission components, and structural components. This classification gives references when the surfaces grow up to entity. According to the existing study of directed graph, gray circles represent locating surfaces of features while white circles represent located surfaces. One big circle stands for a component. Every directive arc carries out some attributive information of the two surfaces in the assembly and is a node which labels the constraint requirements. The directed graph would be ameliorated in this paper so that assembly directional graph is layered according to the function patterns of growth design. The relationship is shown in Figure 2(b), where the dashed line presents the function requirements and the biggest circle contains the components from the same function pattern. It is a simplified structure graph only carrying out one simple functional requirement. As we know, Brep decomposes an entity into different surfaces while the instantiation is the inverse process.

In the growing up process, some constraints are introduced in this prototype as some features are restrained by the location principle. In the constraints propagation, when a surface has been selected, it is necessary to judge whether it has a requirement of precision. If the surface has the requirement, a new surface which could provide a datum or a TTRS is acquired. Search the inherited surface and continue the analysis until the surface is the final datum or the surface of pedestal. After forming a string of function surfaces and restraining the relationships between them, the surfaces grow up to some components. These relationships form the initial constraints or global constraints. To express the function requirements, the global tolerance chains are fixed and tolerances could be allocated according to the global constraints.

Using oriented graph in Figure 3, some global or fundamental constraints are introduced in the construct. Because

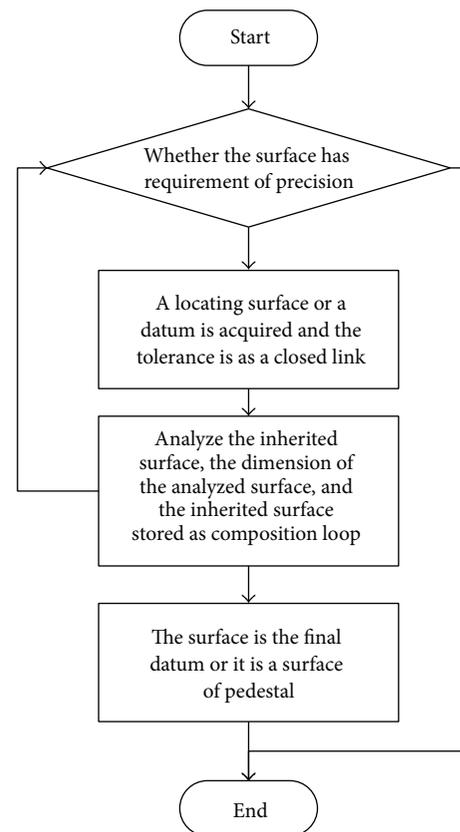


FIGURE 3: The cycle program according to growth design.

of the initial constraints and constructions, instances which match up with the prototype are picked up by searching the database on CM platform, and the node of constraints would provide the basic information while searching the geometric constraint net.

3.2. Resource Configuration Process in CMfg Scenario. When the surfaces grow up into entities, the essential question is matching procedure from the functional surfaces to the instances in the database. The matching procedure contains three elements: information of instance, information of configuration unit, and matching rules.

3.2.1. Information of Instance. The expression of instance (alternative component) is shown in

$$S = (\text{title, description, CSP}). \quad (1)$$

In this equation, title is the encoding when the component came out of the factory on the CM platform and is the unique identifier; the instance is described by description in a textualized way. It is executive information of the instance which could increase searching efficiency; CSP is the constraint including variables, values, and constraints.

3.2.2. Information of Configuration Unit. The expression of configuration unit is shown in

$$C_{\text{Unit}} = (O_{\text{id}}, O_{\text{sel}}, O_{\text{type}}, \text{Rep}(\text{Attr}_{\text{Name}}, \text{Data}_{\text{Type}}, \text{In_Cons}[\text{Domain}])). \quad (2)$$

O_{id} is the unique identifier of the configuration unit; O_{sel} is the selectivity of this unit. When $O_{\text{sel}} = 0$, this unit could be selected while $O_{\text{sel}} = 1$ means this unit must be selected; O_{type} is the type of this unit. When $O_{\text{type}} = 0$, this unit could be manufactured directly; when $O_{\text{type}} = 1$, this unit could be manufactured with more constraints in the following configuration; when $O_{\text{type}} = 2$, this unit could not be manufactured directly and needs to be innovated. $\text{Attr}_{\text{Name}}$ is the name of one attribute in this unit. There are two kinds of attributes: inner attribute is related to the material or geometric relationship inside the component while external attribute is function-related. $\text{Data}_{\text{Type}}$ is the value type of attribute. In_Cons is the constraint while the domain may be given by the customers or calculated by the other constraints.

3.2.3. Matching Rules. In the face of abundant information or knowledge in the manufacturing cloud, many instances could be reused directly from a tiny feature to even an entire product. Satisfying every configuration unit is not the final purpose and even one configuration could match up with a large number of instances, or in other words many resources are attached to the unit as candidates.

Step 1. Input the formalized information of customers' requirements; detect the functional surfaces and original constraints; make these structures grow up and match the instances in the database of cloud manufacturing; pick up the instances to constitute the subset.

Step 2. Check through every configuration unit from every functional surface to ensure the availability: if $O_{\text{sel}} = 1$, the unit is one node of product; if $O_{\text{sel}} = 0$, go to Step 3.

Step 3. Judge whether there is a constraint of requirements based on global constraint; if there is, the unit needs instantiation; otherwise, assign a default value.

Step 4. Merge the possible resources with the growing up of construction.

Step 5. Search the subset which is selected by the primary election to pick up all the possible resources.

Step 6. Work through all the surfaces from the functional surfaces to driving link/stationary part to complete the process.

3.3. The Tolerance Design Based on Game Theory in CMfg Scenario. In the existing methods of tolerance design, the manufacturing scenario is given as a prerequisite while the information of manufacturing resources on CMfg platform is dynamic. In the model of Section 2, several configuration units and the subsets of candidate resources which could meet the requirement of the corresponding configuration unit are given as a result. Different resources combinations give different strategies of manufacturing process, which leads to a complicated scenario. It is difficult to compute all the possible tolerance strategies in this specific circumstance.

In the existing distributed configuration problems solving algorithms, the solution architecture is based on having a centralized main configuration. But since an abundance of knowledge is involved in CMfg, a centralized main configuration might not deal with all the constraints in distributed resources. According to function requirements from the customers and the requirements from profit of the resource suppliers, the game of quality and cost could lead to an equilibrium so that the customers and resource suppliers would not change their choice or strategy. Making the market play a game itself to reach the equilibrium to give an optimal solution of manufacturing design is an intelligent breakthrough in the CMfg area.

After the process in Sections 3.1 and 3.2, several candidate resources are given by the matching result and some constraints are fixed up. But a number of tolerances still need to be allocated. It is assumed that the configuration unit has a tolerance demand and all the units in one game cooperate with each other to satisfy an assembly precision. The price which a certain supplier submits changes with the tolerance demand. The suppliers broadcast information of manufacturing capability and price to the demanders, and every configuration sends the demand of tolerance to the suppliers. Let $J = \{1, 2, 3, \dots, J\}$ denote the set of the resource suppliers and $I = \{1, 2, 3, \dots, I\}$ denote the set of the configuration units. We propose a scenario with multiple resource suppliers and multiple configuration units and formulate the tolerance design problem as a two-level game, that is, an evolutionary game for the configuration units and a noncooperative game for the resource suppliers.

Figure 4 shows the system architecture with multiple resource suppliers and multiple configuration units.

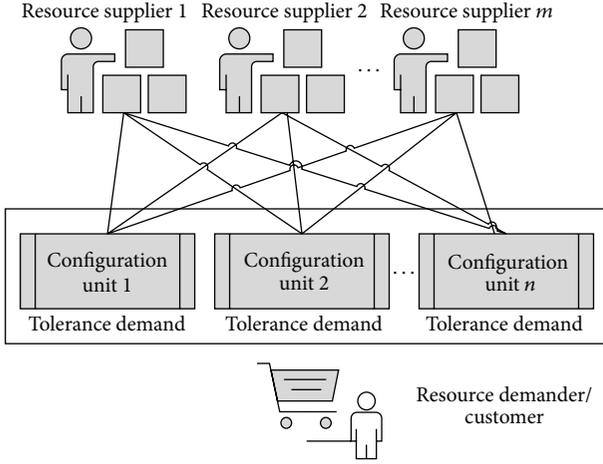


FIGURE 4: The interaction of resource supplier and customer on CMfg platform.

3.3.1. The Noncooperative Game Model of Suppliers. The cost function $f_j(T_j)$ is defined to specify the cost of supplier $j \in \mathcal{J}$, which provides the manufacturing service. Without loss of generality, functions are considered here.

Cost-tolerance model for ex-circle is the following:

$$f_j(T_j) = 15.1138e^{-42.2874T_j} + \frac{T_j}{0.8611T_j + 0.01508}. \quad (3)$$

Cost-tolerance model for inner-hole is the following:

$$f_j(T_j) = 12.6691e^{-37.5279T_j} + 2.486e^{0.000978/T_j}. \quad (4)$$

Cost-tolerance model for locating feature is the following:

$$f_j(T_j) = \begin{cases} 8.2369e^{-35.8049T_j} + 1.3071e^{0.0083/T_j} & T_j \ll 0.13 \\ 1.23036 & T_j > 0.13. \end{cases} \quad (5)$$

Let $p_j(T_j)$ denote the price of the service which decreases with the value of T_j . Then the welfare function is given by

$$U_j(T_j) = p_j(T_j) - f_j(T_j). \quad (6)$$

3.3.2. The Evolutionary Game Model of Configuration Units. The utility function quantifies the utility that unit receives when it purchases the manufacturing service. It coincides with our common sense that the utility is the quality. Without loss of generality, this paper adopts the quadratic utility function as

$$u_i(T_i) = -Q_{\text{Loss}} = -\frac{\alpha}{2}(T_i)^2. \quad (7)$$

If a unit i selects a supplier j as the resource provider, it has to pay p_j when consuming service. The payoff function can be described as

$$\omega_i(T_i) = p_j(T_j) - u_i(T_i) = p_j(T_j) + Q_{\text{Loss}}. \quad (8)$$

In an assembly, each composed link tolerance should satisfy the constraints in

$$T_{i \min} \leq T_i \leq T_{i \max},$$

$$\sum_{i=1}^I \varepsilon_i T_i \leq T_0. \quad (9)$$

3.3.3. Interaction between Resource Supplier and Configuration Unit. For the configuration unit, the main purpose is to purchase a manufacturing service with higher precision at a lower price to achieve higher welfare. On the other hand, from the perspective of the supplier, they aim at selling service at a higher price to achieve higher welfare. Therefore, appropriate strategies should be designed for both suppliers' side and customers' side to maintain the balance between supply and demand.

To characterize the selection process, the configuration unit possesses the following behaviors:

- (i) Each configuration unit has to choose one candidate resource to purchase manufacturing service.
- (ii) Each configuration unit can gradually adjust its strategy on the selection of candidate resource.
- (iii) Each configuration unit behaves independently.

The interaction between the evolutionary game and the noncooperative game is shown in Figure 5.

3.3.4. Analysis of Two-Level Game Theory. It is necessary to prove that the equilibrium point exists in the tolerance game we proposed.

Let y_j denote the probability of a CU choosing resource j , where $0 \leq y_j \leq 1$. The equation $\sum_{j=1}^J y_j = 1$ would be held in every point of the evolution process.

The net payoff of the whole product is defined as the accumulated payoff when all the CUs choose RS j , which is shown in

$$\pi_j = \sum_{i=1}^I \omega_i(T_i). \quad (10)$$

The replicator dynamics which depicts the selection of the population is shown in

$$\frac{\partial y_j}{\partial t} = y_j (\pi_j - \bar{\pi}), \quad (11)$$

where

$$\bar{\pi}_j = \sum_{i=1}^I y_j \pi_j. \quad (12)$$

Let the vector $Y^* = [y_1^*, y_2^*, \dots, y_j^*, \dots, y_J^*]$ present condition in the evolutionary equilibrium point. According to Lyapunov method, the replicator dynamics could be structured.

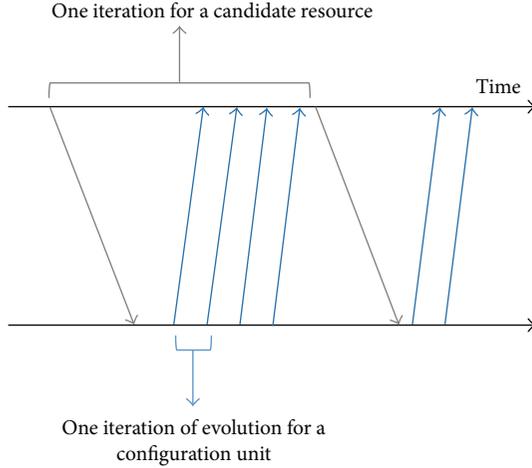


FIGURE 5: Interaction between the evolutionary game and the noncooperative game.

Lemma 1. Assume that there exists a scalar function V of the state \mathbf{x} , with continuous first-order derivatives such that

- (i) $V(\mathbf{x})$ is positive definite,
- (ii) $\dot{V}(\mathbf{x})$ is negative definite,
- (iii) $V(\mathbf{x}) \rightarrow \infty$ as $\|\mathbf{x}\| \rightarrow \infty$.

Then the equilibrium at the origin is globally asymptotically stable.

Define the tracking error function $e_j = y_j^* - y_j$ and the Lyapunov function $V_j(t) = (e_j/2)^2$. Since $V_j(t) \geq 0$, the Lyapunov function is positive definite.

The time derivative of $V_j(t)$ is

$$\begin{aligned} \dot{V}_j(t) &= \frac{\partial (e_j/2)^2}{\partial t} = e_j \frac{\partial e_j}{\partial t} = -e_j \frac{\partial y_j}{\partial t} \\ &= -y_j (y_j^* - y_j) (\pi_j - \bar{\pi}) \\ &= -y_j (y_j^* - y_j) \left(\pi_j - \sum_1^J y_j \pi_j \right). \end{aligned} \quad (13)$$

The net payoff of the resource j would not increase with respect to y_j . We take the resource n which has the lowest net payoff into account. Obviously, the inequations

$$\begin{aligned} y_n &\leq y_n^*, \\ \pi_n - \sum_1^J y_j \pi_j &= \sum_1^J y_j \pi_n - \sum_1^J y_j \pi_j > 0 \end{aligned} \quad (14)$$

could be satisfied.

Apparently, the time derivative of $V_j(t)$ is negative definite. The dynamics of CU choosing resource n will converge to the equilibrium. This process is analogous to the other choice. It is proved that the dynamics system is stable.

In the cooperative game of suppliers, the players are the resource suppliers, of which the strategies are price and precision. The same as with the evolutionary game, the equilibrium needs the proof of existence.

Lemma 2. If the following conditions are satisfied, there exist Nash equilibriums in the game:

- (i) The player set is finite.
- (ii) The strategy sets are closed, bounded, and convex.
- (iii) The utility functions are continuous and quasi-concave in the strategy space.

According to the literature [41], the cost-tolerance curve is concave as shown in Figure 6.

Lemma 3. A function $f(x)$ is concave over a convex set if and only if the function $-f(x)$ is a convex function over the set.

U_j is comprised of two parts. Consider $p_j''(T_j) = 0$, so that the concavity of U_j depends on the concavity of $-f_j(T_j)$. According to Lemma 3, $-f_j(T_j)$ is strict concave, so that U_j is strict concave.

In addition, there are J resource suppliers involved in the game and the strategy sets of players are closed, bounded, and convex. According to Lemma 2, there exist Nash equilibriums in the game.

According to the game theory, the tolerance design could be carried out by the equilibrium.

4. Case Study

A simple assembly is used to illustrate the proposed process. The compression ratio of engine's cylinder should be kept constant at a certain value due to the deviation caused by manufacturing and assembly errors. For this purpose, the function surface is to ensure the space in the cylinder when it is used to compress the air. As the clearance affects the compression ratio directly, the clearance as closing link is 0.27 mm when surface A arrived at the upper dead point shown in Figure 7. So the structure grows up in the progress shown in Figure 8.

- (1) The function requirement is shown in Figure 5; surface A went up and down to inhale and compress with the volume changing of the space with B.
- (2) Surface A grows up into a piston which has a route to change the volume.
- (3) As the piston is a moving part, a transmission would be put into the structure which fastens the piston down in one end of the transmission and connects the motive part in the other end.
- (4) The motive part must be fastened in the same system to make sure the route is correct, so that there would be a fastened point in the box body which is the static structure.
- (5) Surface B must provide a sealed space with the box body so that a cabinet and an end-cover would be produced as a result.

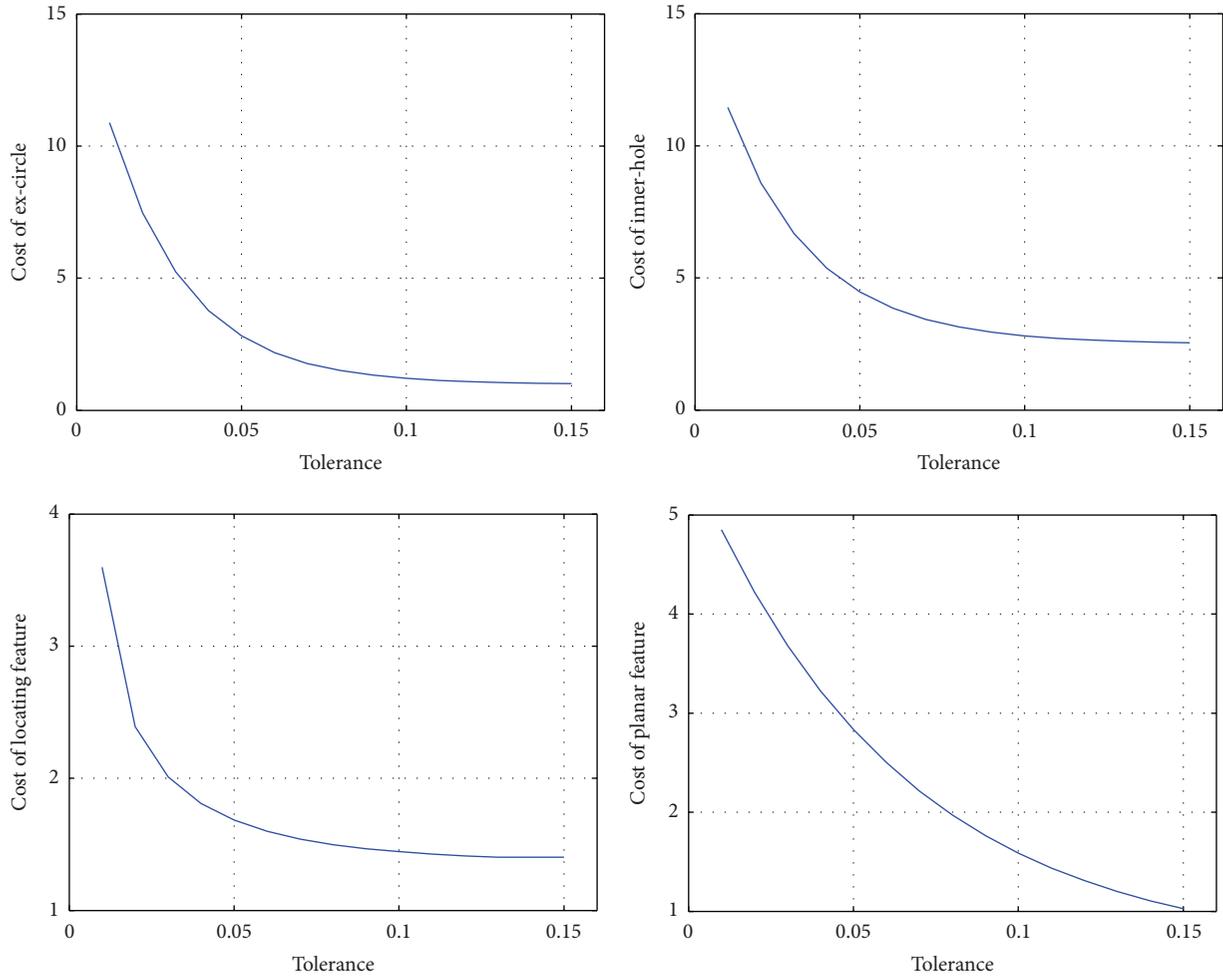


FIGURE 6: Cost-tolerance curve of different features.

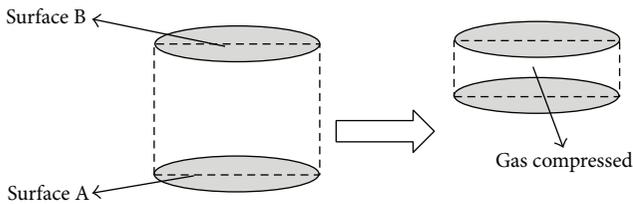


FIGURE 7: Functional requirement description.

(6) The graph shows the whole structure in the upper dead point and down dead point.

The model is figured out by these instances without any dimensions and tolerances except the closing link is 0.27 mm. Five sets of instances were picked up from the files collection of manufacturing resources by OPTIZ system as primary election to reduce the difficulty of the solving progress of the CSP. The results of election are only the structure and some dimension requirements shown in Figure 9 and Table 1. In the next stage, precise election would be done by solving the CSP of the configuration unit and instances. In this illustration,

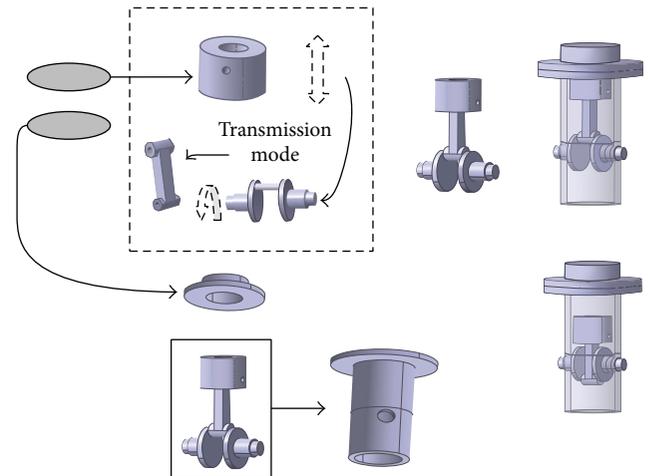


FIGURE 8: The construction growing up process.

there are only two functional surfaces: surface A and surface B. Surface A would be taken into account to show the solution of CSP process. According to the volume requirement from

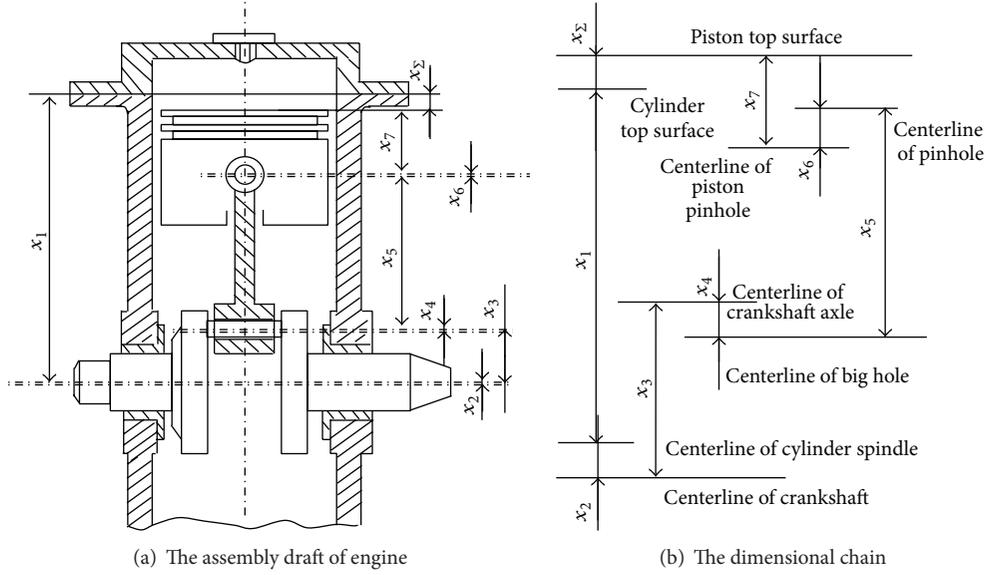


FIGURE 9: The assembly draft and dimensional chain.

TABLE 1: The structure and some dimensions of the product.

Dimension	Name	Tolerance code	Dimension	Name	Tolerance code
x_1	Distance between cylinder top surface and centerline of cylinder spindle	T_1	x_{21}	Diameter of crankshaft spindle	T_{21}
x_2	Eccentricity between centerline of cylinder spindle and axis diameter	T_2	x_{22}	Diameter of cylinder spindle	T_{22}
x_3	Center distance between crankshaft spindle and crankshaft axle	T_3	x_{41}	Diameter of crankshaft axle	T_{41}
x_4	Eccentricity between centerline of big hole and centerline of crankshaft axle	T_4	x_{42}	Diameter of connecting rod big hole	T_{42}
x_5	Center distance between centerline of big hole and centerline pinhole	T_5	x_{61}	Diameter of connecting rod pinhole	T_{61}
x_6	Eccentricity between centerline of pinhole and centerline of piston pinhole	T_6	x_{62}	Diameter of piston pinhole	T_{62}
x_7	Distance between piston top surface and centerline of piston pinhole	T_7	x_Σ	Distance between piston top surface and cylinder top surface	T_Σ

the customers, the configuration unit from surface A is shown in the graph. There is another constraint; that is,

$$x_\Sigma = x_2 + x_4 + x_6 + x_3 + x_5 + x_7 - x_1 \quad (15)$$

s. t.

$$\begin{aligned} T_1^2 + T_2^2 + T_3^2 + T_4^2 + T_5^2 + T_6^2 + T_7^2 &\ll 0.27^2, \\ T_{21}^2 + T_{22}^2 &\leq 0.05^2, \\ T_{41}^2 + T_{42}^2 &\leq 0.05^2, \\ T_{61}^2 + T_{62}^2 &\leq 0.05^2. \end{aligned} \quad (16)$$

But according to the theory of configuration, the constraint among attributes from different units should be avoided because this kind of constraint is so highly coupled that it is not good for the configuration process. To solve this problem, we should distinguish which of A and B is the more important functional surface to give a priority and the equation would be satisfied in the later part of the process to reduce the contradiction. On the other side, the attributes from the same unit also have the constraint with each other just like arrows in the graph; on this occasion, the sequence would follow the process of the structure growing up.

After working through all the attributes of the unit (some value would be solved in the global CSP), the CMfg platform resources data pool comes out, in which every resource

TABLE 2: The dimensions of the components.

Dimension	x_1	x_2	x_3	x_4	x_5	x_6	x_7	x_{21}	x_{22}	x_{41}	x_{42}	x_{61}	x_{62}	x_{Σ}
Nominal dimension	292.3	0	50	0	174	0	69	70	70	57	57	32	32	0.7

TABLE 3: Boolean matrix of $[R \times F(R)]$.

	Cylinder		Crankshaft			Connecting rod			Piston	
	x_1	x_{22}	x_{21}	x_3	x_{41}	x_{42}	x_5	x_{61}	x_{62}	x_7
R_1	0	1	0	1	0	0	0	0	0	0
R_2	1	0	1	0	0	1	0	0	0	0
R_3	0	0	1	0	0	0	0	0	1	1
R_4	1	0	0	0	1	0	1	0	0	0
R_5	0	0	0	0	1	1	0	0	0	0
R_6	1	1	0	0	0	1	0	0	0	0
R_7	0	0	0	0	0	0	1	0	1	0
R_8	0	0	1	1	1	1	0	0	0	0
R_9	0	0	0	0	1	1	1	1	0	0
R_{10}	0	1	0	0	0	0	0	1	0	0

TABLE 4: Boolean matrix of $[T \times F(T)]$.

	Cylinder		Crankshaft			Connecting rod			Piston	
	x_1	x_{22}	x_{21}	x_3	x_{41}	x_{42}	x_5	x_{61}	x_{62}	x_7
IT5	0	0	1	1	1	0	0	1	1	0
IT6	0	1	1	0	1	1	1	1	1	0
IT7	0	1	1	0	1	1	1	1	0	1
IT8	0	1	0	0	0	1	1	0	0	1
IT9	1	0	0	0	0	0	0	0	0	1
IT10	1	0	0	0	0	0	0	0	0	0

combination would satisfy the requirements of the customers. So in the next stage, all the combinations should be worked through to check the best resource allocation according to the objective functions calculated from the customers' requirements.

To show the process in a concise way, the dimensions are given directly in Table 2.

More often than not, features from one part should be manufactured in the same resources to reduce the cost and time caused by frequent resources changing except that some specific features have to be manufactured in specific resources. And for the same purpose, the resources should be as intensive as possible so that the logistics cost could be reduced.

The possible sets in which resources could combine to finish the manufacturing task are shown below. Polychromatic sets theory is introduced to figure out the possible resources chain.

There are parts such as piston, crankshaft, connecting rod, and cylinder and there are ten features to be manufactured. Boolean matrix of $[R \times F(R)]$, $[T \times F(T)]$ $[T \times R(T)]$ could be built to reveal the relationship between resources and features, tolerances and features, and tolerances and resources, and the relationship map could be concluded.

After the Boolean calculation from Tables 3–5, some possible resource chain could be given. To present the process of tolerance allocation without loss of generality, we assume that T_{41} and T_{42} need to be allocated with the candidate resources R_8 and R_9 , and all the other tolerances are fixed up in the configuration process. According to Figure 9, T_{41} and T_{42} belong to the crankshaft and connecting rod, respectively. $T_0 = g(T_1, T_2, \dots, T_l)$ is $T_{41}^2 + T_{42}^2 \leq 0.05^2$. There are two candidate resources which could satisfy the requirement of the tasks.

The cost-tolerance curves of RS8 and RS9 are shown in Figure 10. It is obvious that RS9 is cheaper at low precision service and more expensive at high precision than RS8.

The algorithms are given as follows.

For the evolutionary game model of the configuration units which is T_{41} and T_{42} in this case, Algorithm 4 is executed.

Algorithm 4 (executed by configuration unit $i \in \mathbb{I}$). Consider the following.

- (1) Randomly choose one resource supplier j to buy service;
- (2) $n = 1$;

TABLE 5: Boolean matrix of $[T \times R(T)]$.

	R_1	R_2	R_3	R_4	R_5	R_6	R_7	R_8	R_9	R_{10}
IT5	1	0	1	0	0	1	1	1	1	1
IT6	1	0	1	1	1	1	1	1	1	0
IT7	1	1	1	1	1	1	1	1	1	1
IT8	1	1	1	1	1	1	1	1	1	1
IT9	0	1	1	1	1	1	0	0	0	0
IT10	0	0	0	1	0	1	0	0	0	1

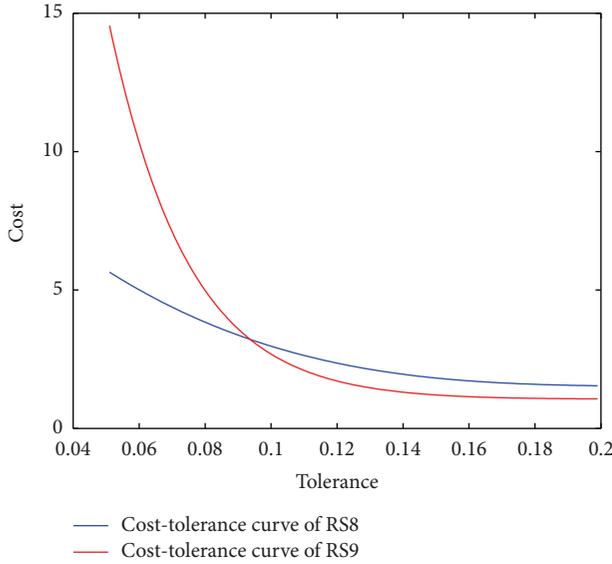


FIGURE 10: The cost-tolerance curves of RS8 and RS9.

- (3) **repeat;**
- (4) calculate the net payoff $\pi_j(n)$ of buying resource from resource supplier j according to $\pi_j = \sum_{i=1}^I \omega_i(T_i)$;
- (5) receive the whole net payoff, and then calculate the average net payoff $\bar{\pi}_j$ according to $\bar{\pi}_j = \sum_{i=1}^I y_j \pi_j$;
- (6) change the probability $y_j(n)$ according to $y_j(n+1) = y_j(n) + \sigma_1 y_j(n)(\pi_j(n) - \bar{\pi}_j(n))$;
- (7) $n = n + 1$;
- (8) **until** $|\pi_j(n) - \bar{\pi}_j(n)| < \varepsilon_1$ is satisfied.

In addition, ε_1 is a small positive constant, n is the iteration number, and σ_1 is the step length.

For the noncooperative game model of the resources suppliers which is RS8 and RS9 in this case, Algorithm 5 is executed.

Algorithm 5 (executed by resource supplier $j \in \mathcal{J}$). Consider the following.

- (1) Keep initial $T_j(1) = T_j^u$, $p_j(1) = C_j = f_j(T_j^u)$;
- (2) $n = 0$;
- (3) **repeat;**

- (4) $n = n + 1$;
- (5) update T_j according to $T_j(n+1) = T_j(n) - \sigma_2$;
- (6) $n = 0$;
- (7) **repeat;**
- (8) $m = m + 1$;
- (9) update p_j according to $\delta_j(m+1) = \delta_j(m) + \sigma_3$;
- (10) execute Algorithm 4 by the configuration unit;
- (11) **until** $W_j = \sum_{i=1}^I y_{ji} U_j(T_j) = \max W_j$ is satisfied,

where m is the iteration number and σ_2 and σ_3 are the step lengths. The initial value of T_j is the upper limit of the tolerance value range, and n and m are the iteration numbers.

For the resource j , the strategy is price and precision.

Given the parameters of quality loss and service price, the optimal result could be calculated.

5. Conclusion

We have studied the configuration process on CMfg platform with distributed resources and manufacturing tasks from customers. In order to realize the aim of faster time to market, higher quality, lower cost, better service, cleaner environment, greater flexibility, and higher knowledge, construction design and the resources allocation are set up in growth design. The interaction among the configuration units is formulated as an evolutionary game, while the interaction among the candidate resources is formulated as a noncooperative game. Simulation result is presented in qualitative analysis to illustrate the whole process from the requirements of the customers using growth design to the resources allocation using the tolerance constraints based on game theory.

Conflict of Interests

The authors declared that they have no conflict of interests regarding this work.

Acknowledgments

This research was supported by the Science Fund for Creative Research Groups of National Science Foundation of China (no. 51221004), the National Nature Science Foundation of China (no. 51275464), and the National Basic Research Program of China (973 Program, no. 2011CB706505).

References

- [1] E. A. Lee, "Cyber-physical systems—are computing foundations adequate?" in *Proceedings of the NSF Workshop on Cyber-Physical Systems: Research Motivation, Techniques and Roadmap*, vol. 2 of *Position Paper*, Austin, Tex, USA, October 2006.
- [2] R. Rajkumar, I. Lee, L. Sha, and J. Stankovic, "Cyber-physical systems: the next computing revolution," in *Proceedings of the 47th ACM/IEEE Design Automation Conference (DAC '10)*, pp. 731–736, ACM, Anaheim, Calif, USA, June 2010.
- [3] L. Wang, M. Törngren, and M. Onori, "Current status and advancement of cyber-physical systems in manufacturing," *Journal of Manufacturing Systems*, 2015.
- [4] B.-H. Li, L. Zhang, L. Ren et al., "Further discussion on cloud manufacturing," *Computer Integrated Manufacturing Systems*, vol. 17, no. 3, pp. 449–457, 2011.
- [5] F. Tao, L. Zhang, H. Guo, Y.-L. Luo, and L. Ren, "Typical characteristics of cloud manufacturing and several key issues of cloud service composition," *Computer Integrated Manufacturing Systems*, vol. 17, no. 3, pp. 477–486, 2011.
- [6] R. Buyya, C. S. Yeo, S. Venugopal, J. Broberg, and I. Brandic, "Cloud computing and emerging IT platforms: vision, hype, and reality for delivering computing as the 5th utility," *Future Generation Computer Systems*, vol. 25, no. 6, pp. 599–616, 2009.
- [7] E. A. Lee, "Cyber physical systems: design challenges," in *Proceedings of the 11th IEEE Symposium on Object Oriented Real-Time Distributed Computing (ISORC '08)*, pp. 363–369, IEEE, Orlando, Fla, USA, May 2008.
- [8] H. Sharifi, G. Colquhoun, I. Barclay, and Z. Dann, "Agile manufacturing: a management and operational framework," *Proceedings of the Institution of Mechanical Engineers, Part B: Journal of Engineering Manufacture*, vol. 215, no. 6, pp. 857–869, 2001.
- [9] M. Mitsuishi and T. Nagao, "Networked manufacturing with reality sensation for technology transfer," *CIRP Annals-Manufacturing Technology*, vol. 48, no. 1, pp. 409–412, 1999.
- [10] C. F. Bremer, "From an opportunity identification to its manufacturing: a reference model for Virtual Enterprises," *CIRP Annals—Manufacturing Technology*, vol. 49, no. 1, pp. 325–330, 2000.
- [11] S. Kara, S. Manmek, and C. Herrmann, "Global manufacturing and the embodied energy of products," *CIRP Annals-Manufacturing Technology*, vol. 59, no. 1, pp. 29–32, 2010.
- [12] R. Shah and P. T. Ward, "Lean manufacturing: context, practice bundles, and performance," *Journal of Operations Management*, vol. 21, no. 2, pp. 129–149, 2003.
- [13] G. Chrysolouris, D. Mavrikios, N. Papakostas, D. Mourtzis, G. Michalos, and K. Georgoulas, "Digital manufacturing: history, perspectives, and outlook," *Proceedings of the Institution of Mechanical Engineers. Part B: Journal of Engineering Manufacture*, vol. 223, no. 5, pp. 451–462, 2009.
- [14] H. Meier, R. Roy, and G. Seliger, "Industrial product-service systems-IPS2," *CIRP Annals—Manufacturing Technology*, vol. 59, no. 2, pp. 607–627, 2010.
- [15] F. Tao, L. Zhang, and A. Y. C. Nee, "A review of the application of grid technology in manufacturing," *International Journal of Production Research*, vol. 49, no. 13, pp. 4119–4155, 2011.
- [16] L. Tao, "Shifting paradigms with the application service provider model," *Computer*, vol. 34, no. 10, pp. 32–39, 2001.
- [17] Y. Fan, D. Zhao, L. Zhang, S. Huang, and B. Liu, "Manufacturing grid: needs, concept, and architecture," in *Grid and Cooperative Computing*, vol. 3032 of *Lecture Notes in Computer Science*, pp. 653–656, Springer, Berlin, Germany, 2004.
- [18] Y. Zhang, G. Zhang, Y. Liu, and D. Hu, "Research on services encapsulation and virtualization access model of machine for cloud manufacturing," *Journal of Intelligent Manufacturing*, 2015.
- [19] D. Wu, D. W. Rosen, L. Wang, and D. Schaefer, "Cloud-based design and manufacturing: a new paradigm in digital manufacturing and design innovation," *Computer-Aided Design*, vol. 59, pp. 1–14, 2015.
- [20] H. Jiao, J. Zhang, J. H. Li, and J. Shi, "Research on cloud manufacturing service discovery based on latent semantic preference about OWL-S," *International Journal of Computer Integrated Manufacturing*, 2015.
- [21] L. Zhang, Y. Luo, F. Tao et al., "Cloud manufacturing: a new manufacturing paradigm," *Enterprise Information Systems*, vol. 8, no. 2, pp. 167–187, 2014.
- [22] P. Helo, M. Suorsa, Y. Hao, and P. Anussornnitisarn, "Toward a cloud-based manufacturing execution system for distributed manufacturing," *Computers in Industry*, vol. 65, no. 4, pp. 646–656, 2014.
- [23] B. Huang, C. Li, C. Yin, and X. Zhao, "Cloud manufacturing service platform for small- and medium-sized enterprises," *The International Journal of Advanced Manufacturing Technology*, vol. 65, no. 9–12, pp. 1261–1272, 2013.
- [24] Y. Cao, S. Wang, L. Kang, C. Li, and L. Guo, "Study on machining service modes and resource selection strategies in cloud manufacturing," *The International Journal of Advanced Manufacturing Technology*, 2015.
- [25] Y. Laili, F. Tao, L. Zhang, and B. R. Sarker, "A study of optimal allocation of computing resources in cloud manufacturing systems," *The International Journal of Advanced Manufacturing Technology*, vol. 63, no. 5–8, pp. 671–690, 2012.
- [26] F. Tao, Y. Cheng, L. D. Xu, L. Zhang, and B. H. Li, "CCIoT-CMfg: cloud computing and internet of things-based cloud manufacturing service system," *IEEE Transactions on Industrial Informatics*, vol. 10, no. 2, pp. 1435–1442, 2014.
- [27] S. Qanbari, F. Li, and S. Dustdar, "Toward portable cloud manufacturing services," *IEEE Internet Computing*, vol. 18, no. 6, pp. 77–80, 2014.
- [28] J. Liu, W. Xu, and H. Zhan, "Multi-source and heterogeneous knowledge organization and representation for knowledge fusion in cloud manufacturing," in *Proceedings of International Conference on Soft Computing Techniques and Engineering Application*, vol. 250 of *Advances in Intelligent Systems and Computing*, pp. 55–61, Springer, New Delhi, India, 2014.
- [29] D. Jannach and M. Zanker, "Modeling and solving distributed configuration problems: a CSP-based approach," *IEEE Transactions on Knowledge and Data Engineering*, vol. 25, no. 3, pp. 603–618, 2013.
- [30] G. Qi, X. Gu, and J. Tan, *Technology and Application of Design for Mass Customization*, China Machine Press, Beijing, China, 2003.
- [31] V. Gruhn, R. Ijioui, D. Peters, and C. Schäfer, "Configuration management for Lyee software," *Knowledge-Based Systems*, vol. 16, no. 7–8, pp. 441–447, 2003.
- [32] T. W. Carnduff and J. S. Goonetillake, "Configuration management in evolutionary engineering design using versioning and integrity constraints," *Advances in Engineering Software*, vol. 35, no. 3–4, pp. 161–177, 2004.

- [33] D. Sabin and R. Weigel, "Product configuration frameworks—a survey," *IEEE Intelligent Systems and Their Applications*, vol. 13, no. 4, pp. 42–49, 1998.
- [34] S. Wang, J. Tan, S. Zhang, X. Wang, and C. He, "Case-based product configuration and reuse in mass customization," *Chinese Journal of Mechanical Engineering*, vol. 17, no. 2, pp. 233–236, 2004.
- [35] J. A. W. Gut and J. M. Pinto, "Optimal configuration design for plate heat exchangers," *International Journal of Heat and Mass Transfer*, vol. 47, no. 22, pp. 4833–4848, 2004.
- [36] S. Myung and S. Han, "Knowledge-based parametric design of mechanical products based on configuration design method," *Expert Systems with Applications*, vol. 21, no. 2, pp. 99–107, 2001.
- [37] N. A. Staudenmayer, M. Tripsas, and C. L. Tucci, "Interfirm modularity and its implications for product development," *Journal of Product Innovation Management*, vol. 22, no. 4, pp. 303–321, 2005.
- [38] Y. Narahari, R. Sudarsan, K. W. Lyons, M. R. Duffey, and R. D. Sriram, "Design for tolerance of electro-mechanical assemblies: an integrated approach," *IEEE Transactions on Robotics and Automation*, vol. 15, no. 6, pp. 1062–1079, 1999.
- [39] H. Zhang, Y. Cao, Y. Wei et al., "A concurrent design method for functional tolerance and structure based on the principle of decomposition and reconstitution," *Procedia CIRP*, vol. 10, pp. 194–202, 2013.
- [40] J. Yang, K. Huang, and Z. Huo, "Function surface-based growth design theory," *Chinese Journal of Mechanical Engineering*, vol. 44, no. 4, pp. 29–35, 2008.
- [41] Z. Wu and J. Yang, *Computer-Aided Tolerancing*, Zhejiang University Press, Hangzhou, China, 1999.



Hindawi

Submit your manuscripts at
<http://www.hindawi.com>

