Hindawi
Mathematical Problems in Engineering
Volume 2018, Article ID 9702910, 13 pages
https://doi.org/10.1155/2018/9702910

Research Article

Bi-Level Programming Model of Cloud Manufacturing Services Based on Extension Theory

Jinhui Zhao[1] **and Yu Zhou**[2]

[1] *Network Information Security Laboratory, Hebei GEO University, Shijiazhuang 050031, China*
[2] *School of Electric Power, North China University of Water Resources and Electric Power, Zhengzhou 450011, China*

Correspondence should be addressed to Jinhui Zhao; zhaojh9977@126.com

Received 23 May 2018; Revised 11 September 2018; Accepted 20 September 2018; Published 9 October 2018

Academic Editor: Giuseppe D'Aniello

In order to select the proper cloud manufacturing services to satisfy both sides of supplier and demander, a bi-level programming model is proposed based on extension theory in this paper. Firstly, the cloud model is employed to convert qualitative concepts of language description into quantitative values. Then, according to multicriteria assessment information, the satisfactions of both providers and demanders are calculated by extension evaluation. Finally, respectively taking satisfactions of the demanders and providers as optimization goal of upper layer and lower layer, a mathematical model of bi-level programming is constructed which is solved by linear programming. Compared with traditional recommended methods, the proposed method takes full account of the interests of service providers which have long been neglected. Moreover, extension evaluation is applied to reflect the demand and preference of each subject in detail. Experimental results verify the effectiveness and applicability of the proposed model.

1. Introduction

With the rise of cloud computing and the technological maturity, cloud computing provides a new way to solve the problems existing in manufacturing informatization [1, 2]. Combined with cloud computing, internet of things, web services technology, virtualization, and other new technologies, researchers proposed the concept of cloud manufacturing (CMfg) based on the existing advanced manufacturing models, such as application service providers, grid manufacturing, and agile manufacturing [3]. CMfg is a new service-oriented mode of networked manufacturing. By means of the network and cloud manufacturing services platform, CMfg organizes online manufacturing resources according to user requirements and provides all kinds of on-demand manufacturing services for users [4]. On the one hand, gathering all kinds of distributed resources and services, cloud manufacturing service platform encapsulates, assembles, designs, and develops the services according to user requirements and centrally organizes and manages massive cloud manufacturing services. On the other hand, the user dynamically gets and uses the required service on the cloud manufacturing service platform and pays for it without focusing on the specific implementation details of the services. Between uncertain, dynamically changing user needs and various distributed, heterogeneous manufacturing resources, the cloud service platform establishes a pattern of allocation and usage for resources, to optimize the allocation of manufacturing resources and promote the value added and efficiency of resources.

With the development and maturity of cloud manufacturing services business mode, there are huge amounts of manufacturing resources in cloud manufacturing service platform, in which some cloud manufacturing services provide the similar or identical functionalities with different nonfunctional attributes. The demanders naturally are willing to choose the services with good quality of service (QoS) and low prices, whereas in the market economy environment, the providers of manufacturing services often have their market positioning according to their strength or specialty and hope to serve a certain market group at good prices. It is very difficult to match a service that needs to be satisfied with both parties in a vast cloud manufacturing resources. Then, how to quickly find satisfied service resources for both demanders and providers has been the concern of the people.

In this work, with the purpose of improving the satisfaction for both demanders and providers, we propose a novel selection method based on bi-level programming model [5–7] to meet interests of both sides. In the bi-level programming selection method, both demanders and providers are evaluated by opposite side according to their individual needs or interests. Firstly, some semantic evaluation indexes are quantified by cloud model [8–10] which is an effective quantitative method with the full consideration of fuzziness and randomness in semantic evaluation. Then, the extension evaluation method [11–14] is used to evaluate the demanders and providers. Finally, the bi-level programming mathematical model is established, which is solved by linear programming method. In this paper, the proposed model can select suitable services in vast amount of cloud manufacturing resources to maximize the satisfaction of both demanders and providers with high efficient, which not only solves the actual problem of resources allocation, but also enriches the research of extension.

The rest of this paper is organized as follows. Section 2 presents related works. In Section 3, the relevant theoretical basis and concepts are introduced. In Section 4, based on bi-level programming, a research model is proposed and the solving process is described in detail. Section 5 reports on some experiments and discusses some observations. Finally, concluding comments and future research work are offered in Section 6.

2. Related Works

The study on selections and compositions of services is very extensive in the domain of web services, but it has only just begun in recent years on cloud manufacturing services. At present, the researches on the selection of cloud manufacturing services mainly are summarized into two aspects: One is form the functional view, and it includes various resources and capabilities in the whole manufacturing lifecycle, for example, production resources, design resources, and simulation resources. The other is nonfunctional view, and it is composed of throughput, response time, price, security, availability, and so on. The bi-level programming model studies the interaction between two subjects with different objectives in an orderly and noncooperative manner. We summarize the related works form three aspects in the following.

(1) *About Functional View*. Formal description of manufacturing resources and capabilities is the base of evaluation, selection, matching, and composition. Wang et al. [15] suggested CMfg task semantic modeling and description based on ontology and studied the construction of general CMfg task ontology and the task semantic matching model. How to provide a natural knowledge representation strategy by semantics was discussed in [16], which uses semantic web services to find the matched service resources, and proposed system was applied in a mechanical parts factory. In [17], the tools information was transformed to OWL-DL ontology to store in an ISO 14649 file, and the similarity between what is offered and what a user expects was calculated by the most appropriate tool. Based on the models of hyper network, a manufacturing resource supply-demand matching simulator was put forward in [18], which can dynamically analyze the supply and demand matching process to improve the efficiency of resources. In the semantic web-based framework, [19] studied the knowledge-based service composition and adaptive resource planning to develop an integrated networked environment allowing fast resource allocation for the given service request. Through creating a business vocabulary reflecting common service selection criteria and defining a textual domain specific language to let any user describe services easily, [20] implemented a novel brokering and matchmaking component to support users' service selection process.

(2) *About Nonfunctional View*. Since manufacturing services have an exponential increase in recent years, the nonfunctional attributes have become the focus of attention, which influence the overall performance of the CMfg systems. QoS, which was a domain-specific metric to evaluate cloud services, was taken into consideration as a crucial factor in selected cloud services. Reference [21] proposed that the computation and evaluation mechanism of QoS for software services should be different from that for hardware services. Lu et al. [22] put forward that the QoS relied heavily on the appropriate group of cloud manufacturing services in the resource pool. In [23], after analyzing software and hardware features of cloud manufacturing services, the authors discussed the impacts of each factor of QoS on CMfg systems. With criteria TQCS (time, quality, cost, and service) being considered, a service selection and scheduling model is established in [24], which employs fuzzy decision-making theory to transform TQCS values into relative superiority degrees. In order to improve the accuracy of QoS evaluation, Ma et al. [25] established the QoS information awareness and quantitative mechanism and calculated the weights through variable precision rough set theory. Meanwhile, many optimization algorithms of selection were proposed. Reference [26] presented dynamic update method of the QoS attribute vector by feedback control and selected services based on the preference weight and QoS attribute vector by technique for order preference by similarity to ideal solution (TOPSIS). According to the fuzzy quality theory, the dominance degree of intuitionistic fuzzy value is taken into account in decision-making; [27] proposed a method for resource service optimal selection based on multivariate process indicator and dominance degree of intuitionistic fuzzy value. From geo-perspective, Lartigau et al. [28] adopted an improved artificial bee colony optimization for QoS to optimal select service compositions. Form nonfunctional view, [29] developed two algorithms to select services, where one leverages a genetic algorithm and the other combines global optimization with local selection. In order to achieve the real-time data-driven optimization decision, [30] put forward a dynamic optimization model for flexible job shop scheduling based on game theory to provide a new real-time scheduling strategy and method. Focusing on the social collaboration feature of manufacturing services, [31] proposed a service selection model that

maximizes the overall synergy effect based on collaboration requirement.

(3) *About Bi-Level Programming Model*. In [6], a bi-level programming model was used to study the distribution center problem where the upper and lower layers are to find the minimum transportation cost from factories to distribution centers and from distribution centers to customers, respectively, which was solved by four algorithms to find an optimum balance between the two layers. In order to determine the optimal schemes of routing and spectrum assignments, Xuan et al. [7] established a bi-level programming model with the energy consumption of the optical networks and the maximum index of used frequency slots as the upper's and lower's objectives to be minimized, respectively, and designed a genetic algorithm with tailor-made crossover, mutation, and local search operator to solve the model. In the study of the proactive countermeasure selection problem for complex information and communication technology systems, Mahjoub et al. [32] proposed a bi-level programming model with a compact formulation based on primal-dual optimality conditions and an extended formulation employing an exponential number of path constraints.

Although significant progresses were made, there are still several research topics to be investigated. Existing researches on service selection mainly focus on how to select and composite the best services to meet the needs of users, and few take into account the market positioning and interest needs of the service providers. In cloud manufacturing service platform, users are more used to evaluating indicators in language; the quantization of semantic evaluation is topic in selection and composite process. Based on QoS, this paper studies the service selection problem from these two aspects.

3. Theoretical Background

3.1. The Cloud Model. The cloud model was proposed by Li et al. [8, 9], which was a cognition model of reciprocal conversion between quantitative representation formed and qualitative conception by a specific structure algorithm. Based on the interaction between probability theory and fuzzy theory, the cloud model can reflect the uncertainty of the concept in natural language as well as the linkage between randomness and fuzziness. Because of less information loss for mutual mapping between qualitative concept and quantitative data [33], the cloud model has been successfully applied in many fields, such as wireless sensor networks [34], image segmentation [35], and decision[36].

Definition 1. Let Z be a universe set described by precise numerical data and $\widetilde{Z}$ be a qualitative concept related to Z. If $x \in Z$ is a random instantiation of concept $\widetilde{Z}$, which satisfies $x \sim \mathrm{N}(\mathrm{Ex}, \mathrm{En}'^2)$, $\mathrm{En}'^2 \sim \mathrm{N}(\mathrm{Ex}, \mathrm{He}^2)$, and the certainty degree of x belonging to concept $\widetilde{Z}$ satisfies

$$y = e^{-(\mathrm{x}-\mathrm{Ex})^2/2(En')^2} \quad (1)$$

Then the distribution of x in the universe Z is called normal cloud, and x is a cloud drop.

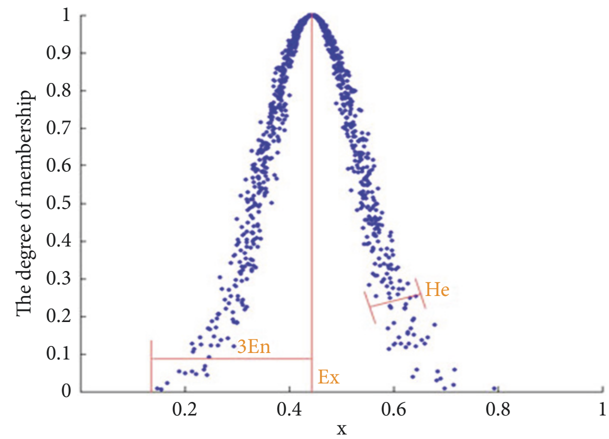


Figure 1: The digital features of cloud model.

Definition 2. The characteristics of a cloud y can be represented by the digital features (Ex, En, He): expectation Ex, entropy En, and superentropy He. Here, Ex is the center value of the qualitative concept domain, which is the best way to describe the fuzzy information. En measures the randomness and fuzziness of the qualitative concept. He reflects the uncertainty of the membership function and the dispersion degree of the cloud drops. The digital features are shown in Figure 1.

3.2. The Extension Theory. In 1983, Cai et al. proposed the extension theory to solve contradictions and incompatibility problems [11]. After years of unremitting research, a series of breakthroughs have been made in the extension theory, method research, and practical application, which has been widely applied to evaluation and selection, data mining, control decision, and other fields.

The hard core of extension theory includes two theoretical pillars: the matter element theory and the theory of extension set. The former is the basic tool to describe the variability of things, which not only take things, features, and values as a unified body to consider the relationship between quality and quantity but also change with the three elements and the internal structure. The latter is the quantitative tool of extension theory to represent the dependent degree of two matter elements through designed correlation function.

3.2.1. The Matter-Element Theory

Definition 3. Defining the name of a matter as N, one of the characteristics for this matter is c and the value of c is v. In extension theory, the matter element can be formally described as

$$R = (N, c, v) \quad (2)$$

Where N, c, and v are called fundamental elements of the matter element.

Definition 4. Assuming that the value of c has a range or a classical domain, the classical domain of the matter element is defined as

$$R = (N, c, v) = (N, c, (v_l, v_u)) \tag{3}$$

where v_l is the lower bound in classical domain and v_u is the upper one.

Definition 5. If the $R = (N, c, v)$ is a multidimensional matter element, $C = \{c_1, c_2, \cdots, c_n\}$ and $v = \{v_1, v_2, \cdots, v_n\}$, then a multidimensional matter is defined as

$$R = (N, C, V) = \begin{bmatrix} N, & c_1 & v_1 \\ & c_2 & v_2 \\ & \cdots & \cdots \\ & c_n & v_n \end{bmatrix} = \begin{bmatrix} R_1 \\ R_2 \\ \cdots \\ R_n \end{bmatrix} \tag{4}$$

where $R_i=(N, c_i, v_i)$ is defined as the submatter element of R, i=1,2,...,n.

3.2.2. Extension Set Theory

Definition 6. If we let U be a universe of discourse, an extension set B on U is defined as a set of ordered pairs described as

$$B = \{(x, y) \mid x \in U, y = \mu(x) \in (-\infty, +\infty)\} \tag{5}$$

where $y = \mu(x)$ is the membership function for extension set B.

The each element of U is mapped to a membership grade between $-\infty$ and $+\infty$ through $\mu(x)$. The higher the membership grade, the more the element belongs to B. According to the special situation, $0 \le \mu(x) \le 1$ accords with a normal fuzzy set, $-1 < \mu(x) < 0$ is an extension domain which means that the element x tends not to belong to B, and $\mu(x) \le 1$ implies that the element x has departed from B.

4. The Bi-Level Programming Model of Cloud Manufacturing Services

4.1. A Bi-Level Programming Research Model. Let D represent the set of demanders with the same or similar requirements, $D = \{d_1, d_2, \dots, d_m\}$. The d_j is the j^{th} demander in D, where j= 1,2,...,m. $S = \{s_1, s_2, \dots, s_n\}$is the set of services with same or similar functions. The s_i is the i^{th} service in S, where i is in (1,2,...,n). The d_j wants to select one manufacturing service to complete his manufacturing task, which must meet his requirements. $C = \{c_1, c_2, \dots, c_f\}$ is the set of assessment indexes that demanders select services. The c_k is the k^{th} index, where k=1,2,...,f, and $c_1, c_2, \dots, c_f$ are independent. $A = \{a_1, a_2, \dots, a_g\}$ is the set of indexes that providers of services evaluate demanders and tasks. The a_t is the t^{th} index where t=1,2,...,g, and $a_1, a_2, \dots, a_g$ are, respectively, independent too.

The requirements of two sides are called constraints. The constraints that have to be satisfied are called hard constraints, and the rest are called soft constraints. The soft constraints include three types:

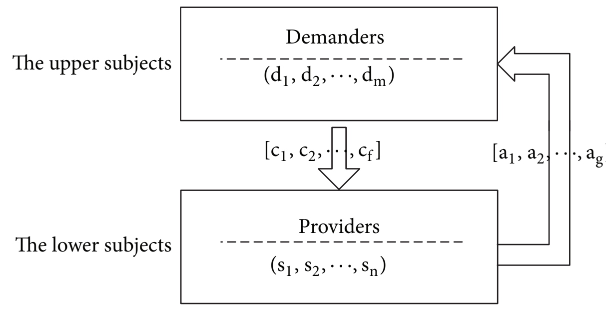


Figure 2: The bi-level programming research model.

(1) Benefit constraints: bigger values in these constraints are better, such as quality of service and reliability.

(2) Cost-based constraints: the values are as small as possible in this type, for instance, the cost.

(3) Interval constraints: the indexes change within a range, manufacturing period, for example.

Because the interests of both demanders and providers need to be considered, based on actual selection process, we structure a bi-level optimization model with a master-slave hierarchical structure from the perspective of QoS, as shown in Figure 2.

The upper subjects represent the set of demanders while the lower subjects stand for the set of services. The upper layer firstly starts to select services and deliver optimization results to the lower layer. Then, according to the results sent by upper layer, the lower layer makes its own optimization choices and sends (through feedback) the optimization results to the upper layer. This process is repeated until the optimal solution is reached. In this way, win-win results meet the requirements of the demanders and fully consider the interests of the service suppliers. These two selection processes are relatively independent and interdependent, which work together to facilitate the whole service selection process of cloud manufacturing services.

4.2. Semantic Quantification Based on Cloud Model. In actual cloud manufacturing environment, perceived index values usually are described as quantitative values and uncertainty evaluation semantic. In order to facilitate the calculation automatically, the cloud model is employed to realize the uncertainty transformation from qualitative concepts of semantic description to quantitative numbers. This paper uses golden segmentation method [37] to generate n cloud evaluation scales. The basic idea of the golden segmentation method is that the semantic values are represented by the cloud model, and each semantic variable corresponds to a cloud. The middle cloud is represented by $E_0(Ex_0, En_0, He_0)$. Adjacent clouds are represented as $\cdots, E_{-2}(Ex_{-2}, En_{-2}, He_{-2}), E_{-1}(Ex_{-1}, En_{-1}, He_{-1}), E_0(Ex_0, En_0, He_0), E_1(Ex_1, En_1, He_1), E_2(Ex_2, En_2, He_2), \cdots$. The closer the cloud is to the central cloud, the smaller the entropy and the superentropy. The smaller is 0.618 times that of the larger in entropy and superentropy of neighboring

Table 1: The calculation methods of the digital feature and values for five clouds.

Semantic Information	Cloud	Ex	En	He
Very Good (VG)	$E_{-2}(Ex_{-2}, En_{-2}, He_{-2})$	x_{mix}	$En_{+1}/0.618$	$He_{+1}/0.618$
Good (G)	$E_{-1}(Ex_{-1}, En_{-1}, He_{-1})$	$Ex_0 - 0.383(x_{\min} + x_{\max})/2$	$0.382(x_{\min} - x_{\max})/6$	$He_0/0.618$
Common (C)	$E_0(Ex_0, En_0, He_0)$	$(x_{\min} + x_{\max})/2$	$0.618En_{+1}$	0.005
Poor (P)	$E_1(Ex_1, En_1, He_1)$	$Ex_0 + 0.383(x_{\min} + x_{\max})/2$	$0.382(x_{\min} - x_{\max})/6$	$He_0/0.618$
Very Poor(VP)	$E_2(Ex_2, En_2, He_2)$	$x_{\max}$	$En_{+1}/0.618$	$He_{+1}/0.618$

clouds. In the evaluation process of cloud manufacturing service, the evaluation phrase is generally divided into five levels. According to experts advices, $[x_{\min}, x_{\max}]$ is [0, 1] and He_0 is 0.005. The calculation methods of the digital feature for the five clouds are shown in Table 1.

The results are as follows: $E_{-2}(0.000, 0.104, 0.013)$, $E_{-1}(0.309, 0.064, 0.008)$, $E_0(0.500, 0.039, 0.005)$, $E_1(0.691, 0.064, 0.008)$, $E_2(1.000, 0.104, 0.013)$.

The positive cloud is a model that transforms the concepts of qualitative variable into quantitative value and its determination. The specific steps are as follows:

The inputs: eigenvector of the cloud, the number of cloud droplets (n).

The outputs: the quantitative values of n cloud droplets and their determination(y).

Step 1: Generate a normal random number (En') according to En and He.

Step 2: Create a normal random number (x) according to Ex and En.

Step 3: Calculate the determination $y = e^{-(x-Ex)^2/2(En')^2}$.

Step 4: (x, y) is a cloud droplet in the domain.

Step 5: Repeat Steps 1 ~ 4 until n droplets are generated.

4.3. Evaluation Based on Extension Analysis. According to the principle of extension, we use the matter-element to descript the cloud manufacturing services, which takes the evaluation indexes as the meta-features and the perceived state value as the corresponding eigenvalues, shown as follows.

$$R_S = (S, C, X) = \begin{bmatrix} S & c_1 & x_1 \\ & c_2 & x_2 \\ & \cdots & \cdots \\ & c_f & x_f \end{bmatrix} \tag{6}$$

According to d_j's requirements for QoS, classical domain matter-element for d_j is as

$$R_j = \left(N_j, C_k, X_{jk}\right) = \begin{bmatrix} N_j, & c_1, & x_{j1} \\ & c_2 & x_{j2} \\ & \cdots & \cdots \\ & c_f & x_{jf} \end{bmatrix} = \begin{bmatrix} N_j, & c_1, & \left\langle a_{j1}, b_{j1}\right\rangle \\ & c_2 & \left\langle a_{j2}, b_{j2}\right\rangle \\ & \cdots & \cdots \\ & c_f & \left\langle a_{jf}, b_{jf}\right\rangle \end{bmatrix} \tag{7}$$

where N_j is for the j^th demander and $x_{jk} = \langle a_{jk}, b_{jk}\rangle$ is the range of c_{jk}in d_j's requirements, which is called classical domain.

After confirming the classical domain, it is necessary to fix the possible ranges of indexes, which is called joint domain matter-element.

$$R_P = (P, C_k, X_{Pk}) = \begin{bmatrix} P, & c_1, & x_{P1} \\ & c_2 & x_{P2} \\ & \cdots & \cdots \\ & c_f & x_{Pf} \end{bmatrix} = \begin{bmatrix} P, & c_1, & \left\langle a_{P1}, b_{P1}\right\rangle \\ & c_2 & \left\langle a_{P2}, b_{P2}\right\rangle \\ & \cdots & \cdots \\ & c_f & \left\langle a_{Pf}, b_{Pf}\right\rangle \end{bmatrix} \tag{8}$$

where x_{P1},x_{P2},. . ., x_{Pf}, respectively, were the scopes of c_1, c_2,. . ., c_f about P.

Because all services properties have the same range and cooperation requirements of different users are different, there are many classical domain matter-elements, and only one joint domain matter-element.

In order to calculate the satisfaction degree of d_j to the k^{th} index, we design the membership function ($\mu_{jk}(x_k)$) of the k^{th} index in s_i as

$$\mu_{jk}\left(x_k\right) = \begin{cases} \dfrac{\rho\left(x_k, X_{jk}\right)}{D\left(x_k, X_{jk}, x_{pk}\right)}, & x_k \notin X_{jk} \\ -\dfrac{\rho\left(x_k, x_0, X_{jk}\right)}{\left|X_{jk}\right|}, & x_k \in X_{jk} \end{cases} \tag{9}$$

where j=1,2,. . .,n; k=1,2,. . .,f.
$\rho(x_k, X_{jk})$ is the distance of x_k to interval $X_{jk} = \langle a_{jk}, b_{jk}\rangle$.

$$\rho\left(x_k, X_{jk}\right) = \left|x_k - \frac{1}{2}\left(a_{jk} + b_{jk}\right)\right| - \frac{1}{2}\left(b_{jk} - a_{jk}\right) \tag{10}$$

$D(x_k, X_{jk}, x_{pk})$ is the distance from interval $x_{Pk} = \langle a_{pk}, b_{pk}\rangle$ to interval $X_{jk} = \langle a_{jk}, b_{jk}\rangle$.

$$D\left(x_k, X_{jk}, x_{pk}\right) = \rho\left(x_k, X_{pk}\right) - \rho\left(x_k, X_{jk}\right) \quad (11)$$

$\rho(x_k, x_0, X_{jk})$ is the distance of x_k to $X_{jk} = \langle a_{jk}, b_{jk}\rangle$ about x_0, where x_0 is the optimal value in this index. $\rho(x_k, x_0, X_{jk})$ includes the left distance and the right distance [11]. When $x_0 \in \langle a_{jk}, (a_{jk} + b_{jk})/2\rangle$, $\rho(x_k, x_0, X_{jk})$ is the left distance, which is marked as $\rho_l(x_k, x_0, X_{jk})$.

$$\rho_l\left(x_k, x_0, X_{jk}\right) = \begin{cases} a_{jk} - x_k, & \left(x_k \le a_{jk}\right) \\ \dfrac{b_{jk} - x_0}{a_{jk} - x_0}\left(x_k - a_{jk}\right), & \left(x_k \in \left\langle a_{jk}, x_0\right\rangle\right) \\ x_k - b_{jk}, & \left(x_k \ge x_0\right) \end{cases} \quad (12)$$

As a special case, when $x_0 = a_{jk}$, the denominator in formula (12) is 0, $x_0 - a_{jk} = 0$; $\rho_l(x_k, a_{jk}, X_{jk})$ is changed as

$$\rho_l\left(x_k, a_{jk}, X_{jk}\right) = \begin{cases} a_{jk} - x_k, & \left(x_k < a_{jk}\right) \\ x_k - b_{jk}, & \left(x_k > a_{jk}\right) \\ a_{jk} - b_{jk}, & \left(x_k = a_{jk}\right) \end{cases} \quad (13)$$

When $x_0 \in \langle (a_{jk} + b_{jk})/2, b_{jk}\rangle$, $\rho(x_k, x_0, X_{jk})$ is the right distance, which is marked as $\rho_r(x_k, x_0, X_{jk})$.

$$\rho_r\left(x_k, x_0, X_{jk}\right) = \begin{cases} a_{jk} - x_k, & \left(x_k \le x_0\right) \\ \dfrac{a_{jk} - x_0}{b_{jk} - x_0}\left(b_{jk} - x_k\right), & \left(x_k \in \left\langle x_0, b_{jk}\right\rangle\right) \\ x_k - b_{jk}, & \left(x_k \ge b_{jk}\right) \end{cases} \quad (14)$$

As a special case, when $x_0 = b_{jk}$, the denominator in formula (14) is 0 too, $x_0 - b_{jk} = 0$; then $\rho_r(x_k, b_{jk}, X_{jk})$ is as

$$\rho_r\left(x_k, b_{jk}, X_{jk}\right) = \begin{cases} a_{jk} - x_k, & \left(x_k < b_{jk}\right) \\ x_k - b_{jk}, & \left(x_k > b_{jk}\right) \\ a_{jk} - b_{jk}, & \left(x_k = b_{jk}\right) \end{cases} \quad (15)$$

$|X_{ij}|$is the modulo of interval $X_{jk} = \langle a_{jk}, b_{jk}\rangle$.

$$\left|X_{jk}\right| = b_{jk} - a_{jk} \quad (16)$$

The graph of designed membership function is shown in Figure 3.

If w_k is the preference of d_j to the assessment index of s_i, and $\sum_{k=1}^{k=f} w_k = 1$, the comprehensive correlative degree that s_i meets the need of d_j can be calculated by

$$\overline{I}_{ji} = \sum_{k=1}^{k=f} w_k \mu_{jk} \quad (17)$$

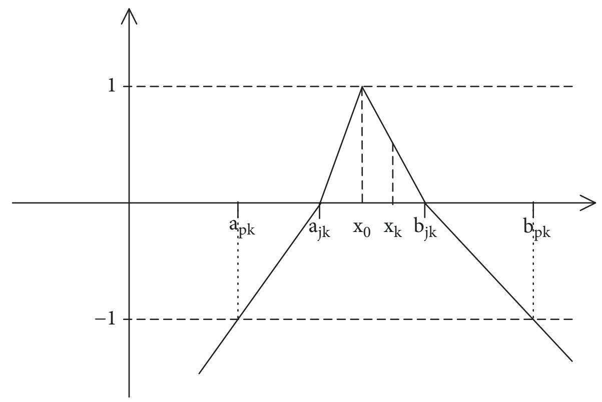


Figure 3: The membership function.

In the same way, comprehensive correlative degree that provider of s_i is satisfied with d_j is

$$\widetilde{I}_{ij} = \sum_{t=1}^{t=g} \omega_t \mu_{it} \quad (18)$$

where ω_t is preference of provider of s_i for the assessment index of d_j or task, $\sum_{t=1}^{t=g} \omega_t = 1$.

4.4. Establishment of Bi-Level Programming Mathematical Model. Aiming at the maximum satisfaction of each part, the bi-level programming mathematical model is established. The upper mathematical model is

$$\max \quad \sum_{i=1}^{n}\left(\sum_{j=1}^{m} x_{ij}\overline{I}_{ij}\right) \quad (19)$$

$$\text{s.t.} \quad \sum_{i=1}^{n} x_{ij} \le 1, \quad j = 1, 2, \cdots, m; \quad (20)$$

$$\sum_{s_i \in S} \overline{I}_{ij} x_{ij} \ge th_i \sum_{s_i \in S} x_{ij}, \quad \forall u_j \in U; \quad (21)$$

$$x_{ij} = \{0, 1\} \quad (22)$$

Formula (19) is the objective function of upper optimization, which means selecting the most satisfied services for the demand side. The constraint formula (20) indicates that demander only selects one service in a matching process. Formula (21) limits the range of services that meet the basic constraints. In formula (22), when $x_{ij} = 1$ it is selected and when $x_{ij} = 0$, it is not selected.

The lower mathematical model is

$$\max \quad \sum_{i=1}^{n}\left(\sum_{j=1}^{m} x_{ij}\widetilde{I}_{ij}\right) \quad (23)$$

$$\text{s.t.} \quad \sum_{j=1}^{m} x_{ij} \le \theta_i, \quad i = 1, 2, \cdots, n; \quad (24)$$

$$\sum_{u_j \in U} \widetilde{I}_{ij} x_{ij} \geq th_i \sum_{u_j \in U} x_{ij}, \quad \forall s_j \in S; \tag{25}$$

$$x_{ij} = \{0, 1\} \tag{26}$$

In the lower optimization model, formula (23) is the optimization function, which recommends the most advantageous task to the provider of service. The constraint formula (24) limits the load of each service. The market positioning and service group of each provider is described by constraint (25).

The proposed model is a multiobjective linear 0-1 integer programming model. In order to solve this model, the objective function is transformed into a linear single target by weighted sum method of membership function. Let fixed weight coefficients be w_1 and w_2, $w_1 + w_2 = 1$; the above multiobjective optimization model is transformed into a single-objective programming model:

$$\max \quad R = w_1 \sum_{i=1}^{n} \left(\sum_{j=1}^{m} x_{ij} \overline{I}_{ij} \right) + w_2 \sum_{i=1}^{n} \left(\sum_{j=1}^{m} x_{ij} \widetilde{I}_{ij} \right) \tag{27}$$

$$\text{s.t.} \quad \sum_{i=1}^{n} x_{ij} \leq 1, \quad j = 1, 2, \cdots, m; \tag{28}$$

$$\sum_{j=1}^{m} x_{ij} \leq \theta_i, \quad i = 1, 2, \cdots, n; \tag{29}$$

$$\sum_{s_i \in S} \overline{I}_{ij} x_{ij} \geq th_i \sum_{s_i \in S} x_{ij}, \quad \forall u_j \in U; \tag{30}$$

$$\sum_{u_j \in U} \widetilde{I}_{ij} x_{ij} \geq th_i (\theta_t) \sum_{u_j \in U} x_{ij}, \quad \forall s_j \in S; \tag{31}$$

$$x_{ij} = \{0, 1\} \tag{32}$$

The model can be solved by linear programming.

A task of the manufacturing industry is generally a combination of manufacturing services in a certain process. Because the composite service set is finally broken down into atomic services that meet certain constraints, this paper only analyzes the matching process of atomic services.

The bi-level programming selection process of the cloud manufacturing services is described as follows:

Step 1: Get demanders' desired vectors of services and the actual perceptive value vectors of services. At the same time, obtain desired vectors of providers to demanders and actual perceptive value vectors of demanders.

Step 2: Quantify the evaluation semantics to numerical values by positive clouds.

Step 3: Establish the classical domain of both supply and demand sides according to desired vectors.

Step 4: Build the joint domain of both providers and demanders according to actual perceptive value vectors.

Step 5: Calculate the comprehensive correlative degree of d_j for s_i and the comprehensive correlative degree of s_i to d_j, respectively.

Step 6: Structure the bi-level optimization model from comprehensive correlative degrees $\overline{I}$ and $\widetilde{I}$.

Step 7: Transform the multiobjective optimization model into a single target optimization model using linear weighted method.

Step 8: Solve the single objective programming model and output the selection result.

5. Experiment Study

In order to verify the rationality and practicability of our proposed model, the simulation platform is carried out in the laboratory environment. JDK8, eclipse4.3, and SQL Server2005 constitute the development environment. Tomcat7.0 is a server to build simulation platform of cloud manufacturing services. The simulation platform is written by java language to achieve the selection and call of cloud manufacturing services. To take some mold processing, for example, the simulation platform contains 200 cloud manufacturing services and 10 demanders to test the mold processing. The maximum load number for each service is 3. On the one hand, the providers publish their services to the simulation platform and set the requirements about the demanders and tasks. On the other hand, the demanders set the requirements of QoS about the manufacturing services according to the actual conditions.

The evaluation indexes of services include the reliability (c_1), credibility (c_2), technical level (c_3), time (c_4), and price (c_5). The providers evaluate the demanders and tasks from three aspects: reputation (a_1), technical difficulty (a_2), and payment speed (a_3).

For example, four demanders simultaneously request a service with the same function at the same time. The attribute values of demanders and QoS requirements of cloud manufacturing services are shown in Table 2.

In Table 2, the reputation (a_1) comes from perceptive module, which updates the data once half a month. The technical difficulty (a_2) is determined according to task through industry standards or demander's experiences and payment speed (a_3) indicates for how many days the demanders will be able to pay after delivery. The requirements of QoS are described by demander when applying for services. Because people tend to use semantics to express their needs, reliability (c_1), credibility (c_2), and technical level (c_3) are the evaluation semantics at five levels. The time (c_4) is the interval number of days that express the task that needs to be completed in a range after the submission of the task. The price (c_5) is the cost of goods reaching the buyer, whose units are Chinese Yuan.

The simulation platform finds 20 cloud manufacturing services that meet the requirements. The perceived values of s_i and cooperation intentions of providers are shown in Table 3.

In Table 3, the perceived values of QoS for each service come from perceptive module too; the reliability (c_1) is represented by the ratio of the number of satisfied tasks

TABLE 2: The attribute values of d_j and the requirements of QoS.

	The attribute values of d_j			the requirements of QoS				
u_j	a_1	a_2	a_3	c_1	c_2	c_3	c_4	c_5
d_1	Poor	Easy	25	Good	Common	Common	[10, 14]	2.5
d_2	Good	Hard	15	Good	Common	Good	[6, 10]	3.2
d_3	Common	Common	20	Good	Good	Good	[8, 12]	3.0
d_4	Good	Very hard	10	Good	Good	Very Good	[4, 8]	3.5

TABLE 3: The perceived values of s_i and cooperation intentions of providers.

	QoS					Cooperation intentions		
s_i	c_1	c_2	c_3	c_4	c_5	a_1	a_2	a_3
s_1	8/11	Common	Common	12	2.57	Good	Common	19
s_2	20/20	Very Good	Good	9	2.83	Common	Hard	15
s_3	0/1	Poor	Poor	20	2.02	Very Good	Easy	28
s_4	7/10	Common	Good	9	2.80	Good	Hard	15
s_5	9/9	Very Good	Common	13	2.47	Good	Common	20
s_6	8/8	Good	Very Good	4	3.68	Good	Very hard	10
s_7	16/20	Common	Common	15	2.29	Common	Common	22
s_8	18/19	Good	Very Good	5	3.58	Common	Very hard	11
s_9	31/31	Very Good	Common	16	2.25	Common	Common	23
s_{10}	28/28	Very Good	Good	6	3.20	Common	Hard	12
s_{11}	30/30	Very Good	Good	12	2.63	Good	Hard	19
s_{12}	1/2	Poor	Good	11	2.70	Common	Hard	18
s_{13}	9/9	Very Good	Common	14	2.33	Very Good	Common	21
s_{14}	15/17	Good	Very Good	8	2.96	Good	Very hard	14
s_{15}	10/15	Common	Very Good	6	3.31	Common	Very hard	12
s_{16}	3/5	Poor	Common	17	2.11	Good	Common	24
s_{17}	2/2	Common	Very Good	4	3.63	Common	Very hard	10
s_{18}	3/3	Good	Poor	18	2.04	Good	Easy	26
s_{19}	20/21	Very Good	Very Good	7	3.19	Good	Very hard	13
s_{20}	21/21	Good	Good	12	2.61	Good	Hard	19

to the number of received tasks; the technical level (c_3) is determined by the service providers according to their processes and techniques; the time (c_4) is the number of days to complete the task after receiving the order; the price (c_5) is the cost of goods reaching the demander, which consists of two parts: manufacturing costs and postage. The cooperation intentions show the market positioning of the service providers, which are submitted when the service is published. The reputation (a_1) and the technical difficulty (a_2) are represented by evaluation semantics while the payment speed (a_3) is the number of days that service providers expect payment after the orders.

5.1. Evaluate Services Based on Extenics. In order to describe the solving process of extension evaluation in detail, we take the selection services as examples to analyze the evaluation process. Firstly, we establish multidimensional matter element of each service from the information of QoS in Table 3. Because the bigger the benefit index is better and the smaller the cost index is better, we can obtain classical domain of each user from the requirements of QoS for four users in Table 2, which are shown in Table 4.

In order to automatically make the selections of cloud manufacturing services matching, the evaluation phrases are quantified by cloud models firstly. The optimal value is determined according to whether the index is benefit attribute or cost attribute excluding c_4. Because the delivery time (c_4) involves the transport, occupation of inventory, and other factors, the middle of the delivery time is optimal. Then, using formula (9) the satisfaction degree of d_j to the k^{th} index $\mu_{jk}(x_k)$ is calculated. The preferences of demanders for each attribute of services are different, but in order to facilitate the analysis, the weight vectors of d_j to s_i are given as $\{0.2, 0.2, 0.2, 0.1, 0.3\}$. According to formula (17), the calculation results of comprehensive correlative degree are shown in Table 5.

In order to analyze the advantages of extension evaluation in services evaluation, a comparative analysis is conducted with TOPSIS evaluation method [26], variable precision rough set method [25], and intuitionistic fuzzy evaluation method [27]. The results of other three methods are shown in Table 6.

The sorted results of the four methods are shown in Table 7.

Table 4: The classical domain of d_j's requirements for services.

The demanders	d_1	d_2	d_3	d_4
The classical domain	$\begin{bmatrix} S_{d1} & c_1 & \langle G, VG\rangle \\ & c_2 & \langle C, VG\rangle \\ & c_3 & \langle C, VG\rangle \\ & c_4 & \langle 10, 14\rangle \\ & c_5 & \langle 2.02, 2.5\rangle \end{bmatrix}$	$\begin{bmatrix} S_{d2} & c_1 & \langle G, VG\rangle \\ & c_2 & \langle C, VG\rangle \\ & c_3 & \langle G, VG\rangle \\ & c_4 & \langle 6, 10\rangle \\ & c_5 & \langle 2.02, 3.2\rangle \end{bmatrix}$	$\begin{bmatrix} S_{d3} & c_1 & \langle G, VG\rangle \\ & c_2 & \langle G, VG\rangle \\ & c_3 & \langle G, VG\rangle \\ & c_4 & \langle 8, 12\rangle \\ & c_5 & \langle 2.02, 3.0\rangle \end{bmatrix}$	$\begin{bmatrix} S_{d4} & c_1 & \langle G, VG\rangle \\ & c_2 & \langle G, VG\rangle \\ & c_3 & \langle VG, VG\rangle \\ & c_4 & \langle 4, 8\rangle \\ & c_5 & \langle 2.02, 3.5\rangle \end{bmatrix}$

Table 5: The calculation results of comprehensive correlative degree.

$d_j \setminus s_i$	s_1	s_2	s_3	s_4	s_5	s_6	s_7	s_8	s_9	s_{10}
d_1	-0.0038	0.3768	-0.1505	-0.0541	0.4430	0.1184	0.1434	0.1104	0.5214	0.2577
d_2	0.0631	0.5175	-0.1829	0.0906	0.4999	0.1697	0.1585	0.2078	0.5277	0.4000
d_3	-0.0025	0.4760	-0.2160	-0.0047	0.4926	0.0650	0.1008	0.0772	0.5253	0.2850
d_4	-0.0158	0.4285	-0.2418	-0.0552	0.4728	0.2200	0.0699	0.2997	0.4937	0.4188
$d_j \setminus s_i$	s_{11}	s_{12}	s_{13}	s_{14}	s_{15}	s_{16}	s_{17}	s_{18}	s_{19}	s_{20}
d_1	0.4751	-0.1404	0.5020	0.2446	0.2859	0.0537	0.0535	0.4189	0.3591	0.3570
d_2	0.5243	-0.0612	0.5259	0.4893	0.1011	0.0161	0.1183	0.3831	0.5887	0.4057
d_3	0.5110	-0.0943	0.5235	0.3149	-0.0555	-0.0155	0.0285	0.3108	0.4699	0.3170
d_4	0.4520	-0.1677	0.4931	0.4109	0.1888	-0.0576	0.2147	0.2801	0.6482	0.2560

Table 6: The calculation results of other three methods.

Evaluation method	s_1	s_2	s_3	s_4	s_5	s_6	s_7	s_8	s_9	s_{10}
TOPSIS method	0.4158	0.6439	0.2500	0.4798	0.5627	0.6998	0.4114	0.7001	0.5447	0.6756
Variable precision rough set method	0.3807	0.5927	0.0513	0.4575	0.4678	0.6258	0.3385	0.6401	0.4224	0.6236
Intuitionistic fuzzy method	0.4228	0.5511	0.3397	0.4623	0.4996	0.6154	0.4137	0.6037	0.4886	0.5764
Evaluation method	s_{11}	s_{12}	s_{13}	s_{14}	s_{15}	s_{16}	s_{17}	s_{18}	s_{19}	s_{20}
TOPSIS method	0.6279	0.3943	0.5476	0.6464	0.5989	0.3348	0.6485	0.4429	0.7379	0.5643
Variable precision rough set method	0.5698	0.3471	0.4368	0.6057	0.5327	0.2199	0.5472	0.2690	0.6957	0.5208
Intuitionistic fuzzy method	0.5419	0.4217	0.4896	0.5477	0.5214	0.3655	0.5804	0.4346	0.5977	0.5160

As shown in Table 7, the sorting results of other three methods are basically the same, but there are four different results for four demanders in proposed method. Specific analysis is as follows.

Firstly, the TOPSIS method [26] takes the optimal and inferior ideal values as reference points. After the reference points are selected, there is only one sorted result for the fixed services. But the extension evaluation create a classical domain matter-element for each user according to the requirements and choose the optimal value (x_0) from the actual situation; each user has a sorting result. The d_1 and d_3, whose tasks are easy and the payment speeds are slow, want to select middle or upper level services, so the middle or upper level services in other sorted sequences are better services for them, while the d_2 and d_4 want better services, whose requirements of tasks and their QoS are high. Hence, there are slight differences between these four queues. The classical domain matter-elements and the optimal values (x_0) dynamically reflect the details of each demander's preference at the indexes. Generally, in the market environment the best service is not necessarily the right service, while the extension method only recommends the right services.

Secondly, in the variable precision rough set method [25], the weights in the services selections are derived from the historical data, and the similarities between the comprehensive evaluation of actual QoS and ideal selection are sorted. Its results are relatively objective. Because of the fixed weight and records of bi-level planning selection in this experiment, after analyzing the 700-1000 records, the weights calculated by the rough set start to approximate the given weights. The similarity formula $(1-\sqrt{\sum(q_j^{\max}\times t_j - Q_{ij}\times w_j)^2})$ [25] is to "$1-\sqrt{\sum[w_j(q_j^{\max}-Q_{ij})]^2}$" which has the same information with the positive ideal distance of the TOPSIS method. The weights in this method are determined by the experience of all demanders, which ignores demanders' preferences, and only apply to those who have no experience at the beginning.

Finally, compared with intuitionistic fuzzy methods [27], in intuitionistic fuzzy operations, the calculation will increase sharply with the increase of evaluation index, which is not suitable with the calculation for the big data. The proposed

Table 7: The sorted results of four methods.

Evaluation method		The sorted results
TOPSIS method		$s_{19} > s_8 > s_6 > s_{10} > s_{17} > s_{14} > s_2 > s_{11} > s_{15} > s_{20} > s_5 > s_{13} > s_9 > s_4 > s_{18} > s_1 > s_7 > s_{12} > s_{16} > s_3$
Variable precision rough set method		$s_{19} > s_8 > s_6 > s_{10} > s_{14} > s_2 > s_{11} > s_{17} > s_{15} > s_{20} > s_5 > s_4 > s_{13} > s_9 > s_1 > s_{12} > s_7 > s_{18} > s_{16} > s_3$
Intuitionistic fuzzy method		$s_6 > s_8 > s_{19} > s_{17} > s_{10} > s_2 > s_{14} > s_{11} > s_{15} > s_{20} > s_5 > s_{13} > s_9 > s_4 > s_{18} > s_1 > s_{12} > s_7 > s_{16} > s_3$
Extension evaluation	d_1	$s_9 > s_{13} > s_{11} > s_5 > s_{18} > s_2 > s_{19} > s_{20} > s_{15} > s_{10} > s_{14} > s_7 > s_6 > s_8 > s_{16} > s_{17} > s_1 > s_4 > s_{12} > s_3$
	d_2	$s_{19} > s_9 > s_{13} > s_{11} > s_2 > s_5 > s_{14} > s_{20} > s_{10} > s_{18} > s_8 > s_6 > s_7 > s_{17} > s_{15} > s_4 > s_1 > s_{16} > s_{12} > s_3$
	d_3	$s_9 > s_{13} > s_{11} > s_5 > s_2 > s_{19} > s_{20} > s_{14} > s_{18} > s_{10} > s_7 > s_8 > s_6 > s_{17} > s_1 > s_4 > s_{16} > s_{15} > s_{12} > s_3$
	d_4	$s_{19} > s_9 > s_{13} > s_5 > s_{11} > s_2 > s_{10} > s_{14} > s_8 > s_{18} > s_{20} > s_6 > s_{17} > s_{15} > s_7 > s_1 > s_4 > s_{16} > s_{12} > s_3$

algorithm in this paper has a good theoretical basis, a small amount of calculation, and flexible parameter setting and can be adapted to different environments. In addition, it can reflect the satisfaction in the details of the property, which is a good way to evaluate the QoS of cloud manufacturing services.

5.2. The Analyses of Bi-Level Programming on Selection. Because the locations of each service in other three sorted queues are very similar from Table 7 and the attributes of selected services by other three methods are not obviously different, the one-way selection algorithm of variable precision rough set method is selected to compare with the proposed bi-level programming model. The experiment designs that four concurrent demanders continuously choose services 10 times and the full load task (θ_i) of each manufacturing service is 3. The benefits of service providers are as important as demander's in this experiment, w_1=w_2=0.5. Then, the single-objective programming model is established from formula (27) to formula (32). Solving the proposed model, the optimal results for both sides are output.

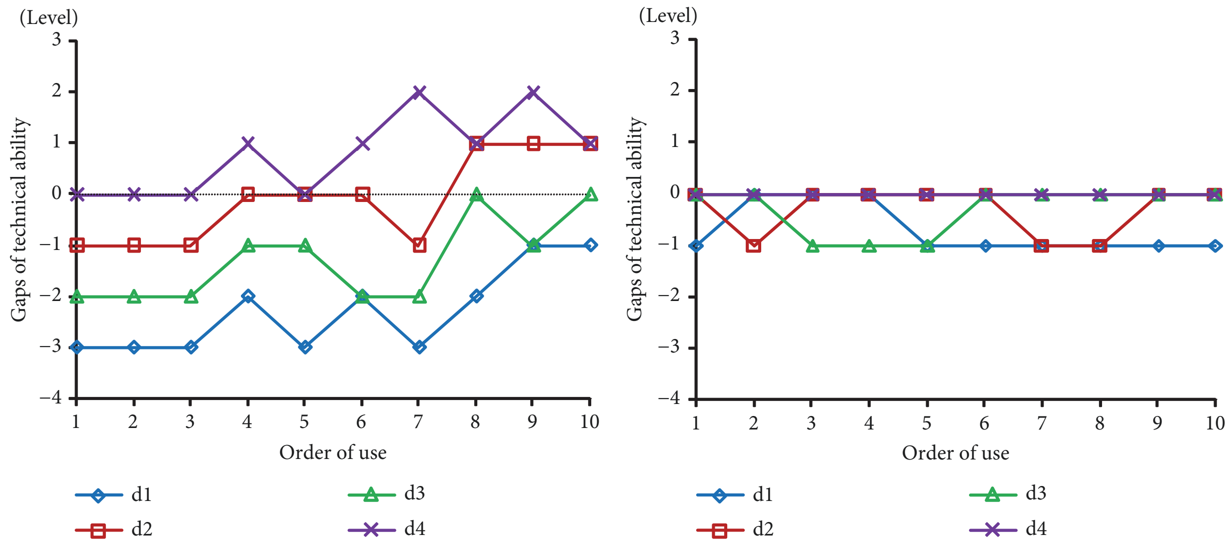
In order to analyze the impacts of one-way choice and two-way choice on the interests of service providers, we compare the market positioning of service providers with the actual situation of demanders and tasks requirements according to the orders in which the demanders use the services. Figures 4, 5, and 6 show the results form technical difficulty (a_2), reputation(a_1), and payment speed(a_3).

Figure 4 shows the level differences of semantic evaluation for technical ability between actual situation of service providers and task demands in both cases. Because at each selection the best service is always recommended to the demander in the traditional one-way selection, services are allocated to d_1, d_2, d_3, and d_4 according to the order of application. At the beginning, d_1 and d_3 whose tasks are easy are allocated excellent manufacturing capacities which inevitably lead to waste of manufacturing capacities. When the demands are tight later on, the weak technical services are recommended to high-demand d_2 and d_4 which make the QoS unwarranted. As shown in Figure 4(a) the curves of d_1 and d_3 go from negative number to close to 0 while the difference of d_2 and d_4 goes from basic consistent to increasing difference. Because the proposed model recommends services according to demanders' requirements by two-way selection and the index of technical ability is a hard constraint, the suitable technical abilities are recommended for the right demanders. The curves shown in Figure 4(b) are mild which means the gaps of technical ability between expectation and reality are not large.

The level differences of semantic evaluation for reputation between cooperation intention of service providers and actual situation of demanders are shown in Figure 5. In the real market, service providers with high technical capability and good service quality often want to provide services for the demanders with good reputation, timely payment, and technical difficulty. But in the traditional one-way service selection process, the interests of service providers are ignored. As a result, the services are allocated to the demanders whose reputations are uneven, and the level gaps of reputation are large between the reputation requirement of service providers and actual situation of demanders in Figure 5(a). As shown in Figure 5(b), there are slightly larger gaps of reputation between d_1 with poor reputation and reputation requirement of selected services, which are also in the acceptable ranges.

From Figure 6 we can see that the differences of days for payment speed between cooperation intention of service providers and actual situation of demanders are big in Figure 6(a) while they are small in Figure 6(b). The payment period is usually related to the manufacturing duration. In the traditional one-way selection, the high quality services, selected at beginning, quickly complete those tasks while the demanders are slow to pay which have resulted in a serious lag in the payment period. As shown in Figure 6(a), only the payment period of d_4, whose requirements of QoS are high, meets the requirements at the beginning. With the increase of the tasks the QoS of selected services are getting worse. The payment periods of d_1, d_2, and d_3 with poor requirements for QoS begin to meet the requirements of payment period, but the d_4's payment periods are in advance. Figure 6(b) shows that the providers of selected services are basically satisfied with the actual payment periods of the demanders.

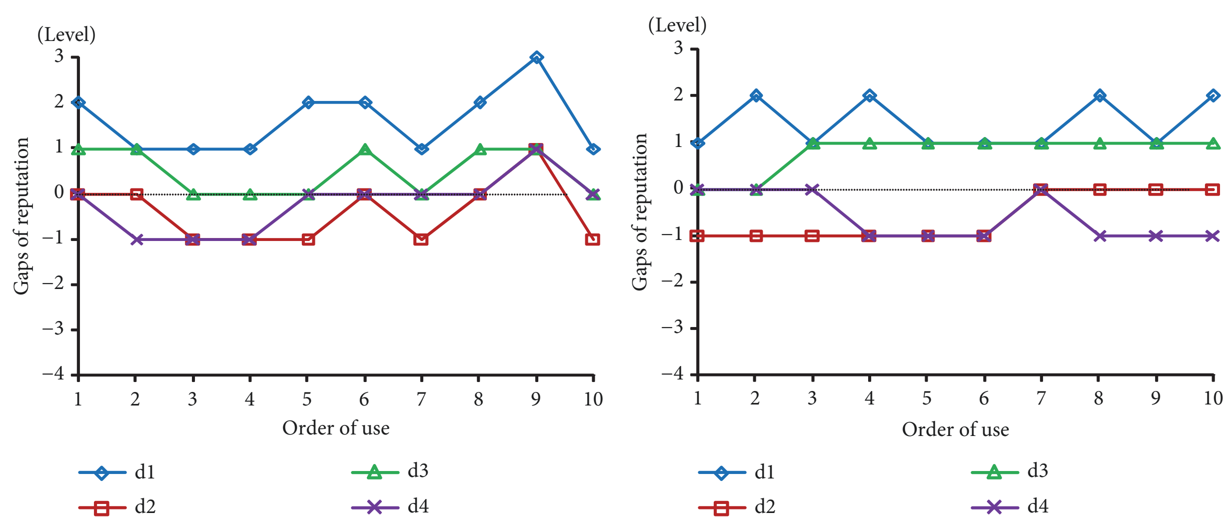
In fact, the selection process of the cloud manufacturing service is a consultation process of manufacturing outsourcing. Service providers have their own market positioning and hope to provide services for a group of demanders according to strength and market demand, while the demanders select different services according to the actual manufacturing task. In a realistic market environment the best one is not necessarily the right one. In the selection process of cloud manufacturing services, it is necessary to ensure the interests of both the providers and the demanders for the harmonious and healthy development of cloud manufacturing industry. The proposed model does that well.



(a) Services selection by one-way

(b) Services selection by bi-level programming model

Figure 4: The gaps of technical abilities.



(a) Services selection by one-way

(b) Services selection by bi-level programming model

Figure 5: The gaps of reputations.

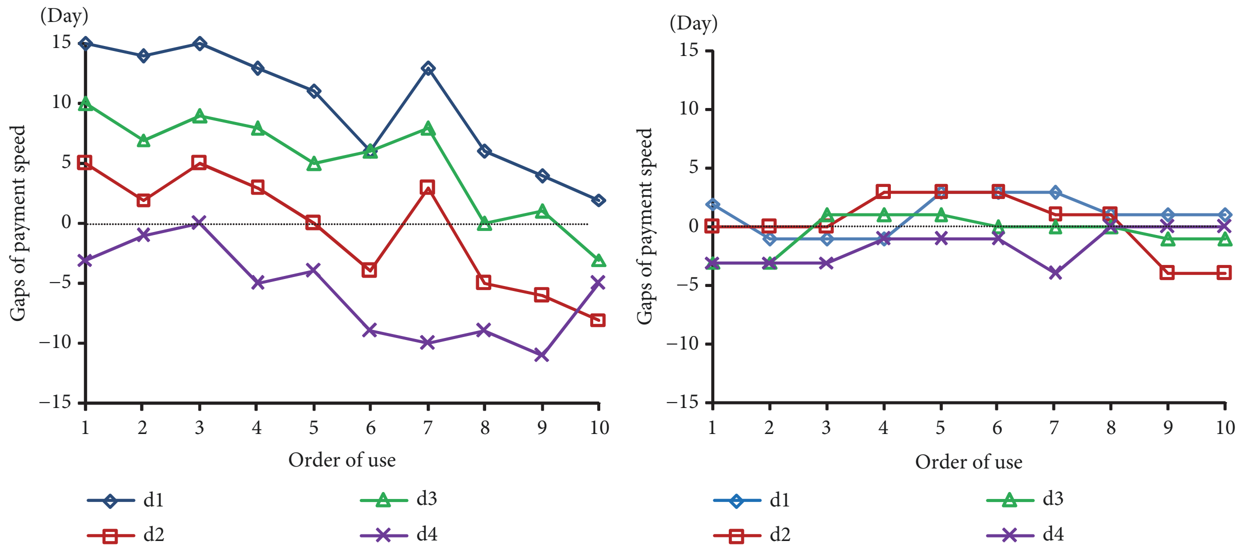
6. Conclusion

To better serve both sides of the supply and demand in the cloud manufacturing service platform, we propose a bi-level programming model of cloud manufacturing services to select services for users based on extension theory. The proposed model employs the cloud model to realize the uncertainty transformation from qualitative concepts of language description to quantitative numbers and extension method to assess both sides of the supply and demand, which is solved by linear programming. Compared with traditional recommended methods, our method takes full account of the interests of service providers which have long been neglected. Moreover, the cloud model can reflect the uncertainty of the concept in natural language and the linkage between randomness and fuzziness and the extension evaluation reflect the satisfaction in the details of the property with simplified calculation. Experimental results show that the proposed model is a better solution for the selection of cloud manufacturing services and more suitable with the actual transaction situation.

The next step is to focus on the two-way selection and matching of composite services in cloud manufacturing.

Data Availability

The data used to support the findings of this study are included within the article.



(a) Services selection by one-way

(b) Services selection by bi-level programming model

FIGURE 6: The gaps of payment speed.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

Acknowledgments

This paper is supported by National Natural Science Foundation of China (Grant no.U1504622) and Key Subjects of Social Science Development in Hebei Province (Grant nos. 201602020215, 201802020211).

References

[1] X. Xu, "From cloud computing to cloud manufacturing," *Robotics and Computer-Integrated Manufacturing*, vol. 28, no. 1, pp. 75–86, 2012.

[2] K.-B. Ooi, V.-H. Lee, G. W.-H. Tan, T.-S. Hew, and J.-J. Hew, "Cloud computing in manufacturing: The next industrial revolution in Malaysia?" *Expert Systems with Applications*, vol. 93, pp. 376–394, 2018.

[3] B. H. Li, L. Zhang, S. L. Wang et al., "Cloud manufacturing: a new service-oriented networked manufacturing model," *Computer Integrated Manufacturing Systems*, vol. 16, no. 1, pp. 1–16, 2010.

[4] F. Tao, L. Zhang, V. C. Venkatesh, Y. Luo, and Y. Cheng, "Cloud manufacturing: a computing and service-oriented manufacturing model," *Proceedings of the Institution of Mechanical Engineers, Part B: Journal of Engineering Manufacture*, vol. 225, no. 10, pp. 1969–1976, 2011.

[5] J. Bracken and J. T. McGill, "Mathematical programs with optimization problems in the constraints," *Operations Research*, vol. 21, pp. 37–44, 1973.

[6] S. Saranwong and C. Likasiri, "Bi-level programming model for solving distribution center problem: A case study in Northern Thailand's sugarcane management," *Computers & Industrial Engineering*, vol. 103, pp. 26–39, 2017.

[7] H. Xuan, Y. Wang, Z. Xu, S. Hao, and X. Wang, "New bi-level programming model for routing and spectrum assignment in elastic optical network," *Optical and Quantum Electronics*, vol. 49, no. 5, 2017.

[8] D. Li and Y. Du, *Artificial Intelligence with Uncertainty*, Chapman & Hall, Boca Raton, Fla, USA, 2017.

[9] D. Li, C. Liu, and W. Gan, "A new cognitive model: cloud model," *International Journal of Intelligent Systems*, vol. 24, no. 3, pp. 357–375, 2009.

[10] K.-Q. Wang, H.-C. Liu, L. Liu, and J. Huang, "Green supplier evaluation and selection using cloud model theory and the QUALIFLEX method," *Sustainability*, vol. 9, no. 5, pp. 1–17, 2017.

[11] W. Cai, "The extension sets and incompatible problems," *Journal of Science Exploration*, vol. 3, no. 1, pp. 83–98, 1983.

[12] J. Ye, "Application of extension theory in misfire fault diagnosis of gasoline engines," *Expert Systems with Applications*, vol. 36, no. 2, pp. 1217–1221, 2009.

[13] Y. Zhou, L. Tian, and L. Liu, "Improved extension neural network and its applications," *Mathematical Problems in Engineering*, vol. 2014, Article ID 593021, 14 pages, 2014.

[14] X. Zhang and J. Yue, "Measurement Model and its Application of Enterprise Innovation Capability Based on Matter Element Extension Theory," in *Proceedings of the 13th Global Congress on Manufacturing and Management, GCMM 2016*, pp. 275–280, China, November 2016.

[15] T. Wang, S. Guo, and C.-G. Lee, "Manufacturing task semantic modeling and description in cloud manufacturing system," *The International Journal of Advanced Manufacturing Technology*, vol. 71, no. 9-12, pp. 2017–2031, 2014.

[16] A. Garcia-Crespo, B. Ruiz-Mezcua, J. L. Lopez-Cuadrado, and J. M. Gomez-Berbis, "Conceptual model for semantic representation of industrial manufacturing processes," *Computers in Industry*, vol. 61, no. 7, pp. 595–612, 2010.

[17] C. Singh, Q. Shao, Y. Lu, X. Xu, and X. Ye, "Tool selection: a cloud-based approach," *Lecture Notes in Electrical Engineering*, vol. 301, pp. 237–245, 2014.

[18] F. Tao, J. Cheng, Y. Cheng, S. Gu, T. Zheng, and H. Yang, "SDMSim: A manufacturing service supply–demand matching simulator under cloud environment," *Robotics and Computer-Integrated Manufacturing*, vol. 45, pp. 34–46, 2017.

[19] Y. Lu and X. Xu, "A semantic web-based framework for service composition in a cloud manufacturing environment," *Journal of Manufacturing Systems*, vol. 42, pp. 69–81, 2017.

[20] M. Slawik, B. I. Zilci, A. Küpper, B. İ. Zilci, and A. Küpper, "Establishing user-centric cloud service registries," *Future Generation Computer Systems*, vol. 87, pp. 846–867, 2018.

[21] F. Tao, Y. Laili, L. Xu, and L. Zhang, "FC-PACO-RM: a parallel method for service composition optimal-selection in cloud manufacturing system," *IEEE Transactions on Industrial Informatics*, vol. 9, no. 4, pp. 2023–2033, 2013.

[22] N. Tajik, R. Tavakkoli-Moghaddam, B. Vahdani, and S. Meysam Mousavi, "A robust optimization approach for pollution routing problem with pickup and delivery under uncertainty," *Journal of Manufacturing Systems*, vol. 33, no. 2, pp. 277–286, 2014.

[23] D. Mourtzis and E. Vlachou, "Cloud-based cyber-physical systems and quality of services," *TQM Journal*, vol. 28, no. 5, pp. 704–733, 2016.

[24] Y. Cao, S. Wang, L. Kang, and Y. Gao, "A tqcs-based service selection and scheduling strategy in cloud manufacturing," *International Journal of Advanced Manufacturing Technology*, vol. 82, no. 1-4, pp. 1–17, 2016.

[25] W.-L. Ma, L.-N. Zhu, and W.-L. Wang, "Cloud service selection model based on QoS-aware in cloud manufacturing environment," *Computer Integrated Manufacturing Systems, CIMS*, vol. 20, no. 5, pp. 1246–1254, 2014.

[26] X. Q. He and Y. H. Wang, "Service selection algorithm based on dynamic assessment for web of things," *Acta Electronic Sinica*, vol. 41, no. 1, pp. 117–122, 2013.

[27] Y. Cao, Z. Wu, T. Liu, Z. Gao, and J. Yang, "Multivariate process capability evaluation of cloud manufacturing resource based on intuitionistic fuzzy set," *The International Journal of Advanced Manufacturing Technology*, vol. 84, no. 1-4, pp. 227–237, 2016.

[28] J. Lartigau, X. Xu, L. Nie, and D. Zhan, "Cloud manufacturing service composition based on QoS with geo-perspective transportation using an improved Artificial Bee Colony optimisation algorithm," *International Journal of Production Research*, vol. 53, no. 14, pp. 4380–4404, 2015.

[29] H. Wang, P. Ma, Q. Yu, D. Yang, J. Li, and H. Fei, "Combining quantitative constraints with qualitative preferences for effective non-functional properties-aware service composition," *Journal of Parallel and Distributed Computing*, vol. 100, pp. 71–84, 2017.

[30] Y. Zhang, J. Wang, S. Liu, and C. Qian, "Game theory based real-time shop floor scheduling strategy and method for cloud manufacturing," *International Journal of Intelligent Systems*, vol. 32, 2017.

[31] M. Ren, L. Ren, and H. Jain, "Manufacturing service composition model based on synergy effect: a social network analysis approach," *Applied Soft Computing*, vol. 70, pp. 288–300, 2018.

[32] A. Ridha Mahjoub, M. Yassine Naghmouchi, and N. Perrot, "A bilevel programming model for proactive countermeasure selection in complex ICT systems," *Electronic Notes in Discrete Mathematics*, vol. 64, pp. 295–304, 2018.

[33] L. Li, F. Fan, L. Ma, and Z. Tang, "Energy utilization evaluation of carbon performance in public projects by fahp and cloud model," *Sustainability*, vol. 8, no. 7, p. 630, 2016.

[34] T. Zhang, L. Yan, and Y. Yang, "Trust evaluation method for clustered wireless sensor networks based on cloud model," *Wireless Networks*, vol. 1, p. 21, 2016.

[35] K. Qin, K. Xu, F. Liu, and D. Li, "Image segmentation based on histogram analysis utilizing the cloud model," *Computers & Mathematics with Applications*, vol. 62, no. 7, pp. 2824–2833, 2011.

[36] J.-Q. Wang, J.-J. Peng, H.-Y. Zhang, T. Liu, and X.-H. Chen, "An uncertain linguistic multi-criteria group decision-making method based on a cloud model," *Group Decision and Negotiation*, vol. 24, no. 1, pp. 171–192, 2015.

[37] H. L. Wang and Y. Q. Feng, "On multiple attribute group decision making with linguistic assessment information based on cloud model," *Control Decision*, vol. 20, no. 6, pp. 679–681, 2005.