Semantic matching of cloud resource service for user product design requirement

Minghai YUAN*, Chao SUN*, XianXian CAI*, Zhuo ZHOU* and Wenbin GU*
*College of Mechanical and Electrical Engineering, Hohai University,
200 Jinling North road, Xinbei District Changzhou, 213001, China
E-mail: ymhai@hhu.edu.cn

Received: 10 March 2019; Revised: 9 June 2019; Accepted: 18 November 2019

Abstract
Under the cloud manufacturing (CMfg) environment, a multilayer semantic matching model of design requirement and cloud service resource based on ontology was proposed to guarantee the rapid and accurate matching of resource services to meet the cloud user design requirement. First, the CMfg classification of demand and service was analyzed, and the ontology description was derived according to the service resource classification. From the aforementioned service resource classification, the multilayer matching model was established. The model included six matching stages, namely, base information matching, function information matching, state information matching, quality of service information matching, manufacturing capability information, and comprehensive matching. Then, the matching algorithm of each layer was quantitatively established and described to achieve intelligent matching of the service resource. Meanwhile, an improved analytic hierarchy process was presented to express the influence of each layer on its matching goal scientifically, ensuring that each affecting factor is reasonable. Finally, a case was used to prove the correctness and practicality of the proposed method.

Keywords: Cloud manufacturing (CMfg), Semantic matching model, Service resource, Analytic hierarchy process, Manufacturing systems

1.Introduction

In recent years, with each country’s emphasis on the industry, emerging technology such as cloud computing, network manufacturing, and Internet of Things rapidly developed and promoted the manufacturing industry, from manufacture to service, providing an advanced manufacturing service development model—cloud manufacturing (CMfg) (Wang at al., 2013). In order to promote the rapid development of the manufacturing industry and the implementation of the cloud manufacturing as soon as possible, a number of countries have initiated a large number of projects and plans, such as Industry 4.0 in Germany (Zhang at al., 2014), High Technology Development Program (863 Plan) (Wang at al., 2013) and Government Work Report 2025 (Li et al., 2016) in China, CAPP-4-SEMS Project (Zhang at al., 2014) and Salesforce CRM and sales cloud (Fei et al., 2014) in America, White Paper on Manufacturing (Zhang at al., 2014) in Japan.

CMfg resource service matching technology is one of the key technologies in the implementation of the service development model. The CMfg environment, according to the personalized resource service demands of the cloud user, relies on the CMfg service platform to achieve intelligent service matching from functional diversity and great variety of CMfg service resources (Wu et al., 2014). This manufacturing service model not only meets the service optimization demands of cloud users but also realizes the efficient and integrated management of CMfg resources. With CMfg gaining considerable interest in the academia and industry, an increasing number of researchers have started to investigate CMfg resource service matching. Lartigau et al. (2015) proposed an approach of CMfg service matching and composition based on quality of service with geo-perspective transportation. Belouadha et al. (2014) modelled business processes that could be performed by web services. They described a semantic matching algorithm for discovering Web Services based on BPMN and UML description in their paper. For manufacturing service aggregation in CMfg, Xu at al. (2016) introduced the population diversity preservation strategy and acceptance strategy of solution to avoid form local optimal.
Feng et al. (2017) proposed a semantics-based supply–demand classification matching method (SDCM) to obtain higher matching accuracy. Sheng et al. (2016) built an intelligent semantic matching engine of CMfg service based on OWL-S for small- and medium-sized enterprises. Zhou et al. (2017) proposed hybrid artificial bee colony (HABC) algorithm to solve the problem that requirements, composited CMfg service optimal selection (CCSOS).

From the aforementioned studies, the development of CMfg has clearly and gradually descended from the cloud alone to the preliminary application. However, most matching researches start from the overall perspective of technology, achieving global resource service matching. Studies on resource matching model, qualitative description, and other aspects are more in-depth, but works on specific resource matching processes and quantization of matching algorithm are limited. Besides, due to the diversity and complexity of manufacturing resources in cloud environment, the adaptability and practicability of matching methods need to be further explored. Therefore, from the user product design requirement perspective, the specific service matching process and quantitative matching algorithm are extensively investigated. After consulting relevant literature, the resource service matching research based on similarity can be divided into the following types (Rese et al., 2009; Zhang et al., 2010; Feng et al., 2017; Ye, and Zhang 2007; Yang et al., 2005; Xie et al., 2017; Huang et al., 2007). Details are shown in Table 1.

| Algorithm type                  | Calculation index type                                                                 | Computational model                                                                 |
|---------------------------------|----------------------------------------------------------------------------------------|--------------------------------------------------------------------------------------|
| **Ontology concept word similarity** | Semantic similarity based on distance                                                   | $Sim(w_1, w_2) = \frac{2 \times (\text{len}(w_1, w_2) - \text{dis}(w_1, w_2))}{2 \times (\text{len}(w_1, w_2))}$ |
|                                 | Semantic similarity based on attributes                                                 | $Sim(w_1, w_2) = \alpha f(w_1 \cap w_2) - \beta f(w_1 - w_2) - \delta f(w_2 - w_1)$  |
|                                 | Semantic similarity based on information content                                        | $Sim(w_1, w_2) = \frac{2 \log^2 p(w_1, w_2)}{\log^2 p(w_1) + \log^2 w_2}$            |
|                                 | **Semantic distance**                                                                 | $Sim(w_1, w_2)_{\text{dist}} = \frac{2 \times \text{Depth}(T)}{2 \times \text{Depth}(T) - \text{dis}(w_1, w_2)}$ |
|                                 | **Semantic depth**                                                                     | $Sim(w_1, w_2)_{\text{dep}} = \frac{2 \times \text{Depth}(\text{Lca}(w_1, w_2))}{\text{Depth}(w_1, \text{Depth}(w_2))}$ |
|                                 | **Semantic coincidence**                                                                | $Sim(w_1, w_2)_{\text{cap}} = \frac{\text{Par}(w_1) \cap \text{Par}(w_2)}{\max(\text{Par}(w_1), \text{Par}(w_2))}$ |
|                                 | **Semantic density**                                                                   | $Sim(w_1, w_2)_{\text{den}} = \frac{\text{Wid}(w_1) + \text{Wid}(w_2)}{2 \times \max(\text{Wid}(w_1), \text{Wid}(w_2))}$ |
| **Word length similarity**      | $Sim(w_1, w_2)_l = 1 - \frac{\text{abs}(\text{Len}(w_1) - \text{Len}(w_2))}{\text{Len}(w_1) + \text{Len}(w_2)}$ |
| **Word property similarity**    | $Sim(w_1, w_2)_p = 2 \times \frac{\text{same}(w_1, w_2)}{\text{Len}(w_1) + \text{Len}(w_2)}$ |
| **Word order similarity**       | $Sim(w_1, w_2)_o = 1 - \frac{\text{Re}(w_1, w_2)}{\max \text{Re}(w_1, w_2)}$ |
| **Numerical parameter information similarity** | $Sim(A_i, B_j) = \begin{cases} 1 & (A_i \cap B_j = A_i \land A_i \cap B_j = B_j) \\ \frac{|A_i \cap B_j|}{|A_i|} & (A_i \cap B_j \neq A_i \land A_i \cap B_j \neq B_j) \\ 0 & (A_i \cap B_j = \emptyset) \end{cases}$ |
| **Fuzzy number similarity**     | $Sim(A_i, B_j) = \begin{cases} 1 & A_i = B_j \\ e^{-d^2(A_i, B_j)/\alpha} & A_i \neq B_j \end{cases}$ |

Note: Variables in the model can be referred to the corresponding references.

Based on Table 1, the matching process will have some limitations when an index factor is considered separately. For example, the semantic similarity based on distance is greatly influenced by ontology hierarchy. In order to clearly express the attributes and relationships of each concept, the semantic similarity based on attributes will require higher description of ontology concept attribute information. For the semantic similarity based on information content, the detailed differences of ontology hierarchy can not be handled well. Therefore, it is necessary to take all factors into account and
measure the similarity of ontology concepts based on multi-level matching of resource services. The rest of the paper is organized as follows: In Section 2, the matching model between user product design requirements and service resources is investigated and discussed. The specific matching process of multilayer matching is also presented. In Section 3, the specific algorithm for resource service matching is designed and illustrated. In Section 4, an improved analytic hierarchy process (I-AHP) is proposed to express the influence of each layer on its matching goal scientifically. In Section 5, an instance is demonstrated to validate the effectiveness and practicality of the proposed model and method. In Section 6, the conclusion is drawn.

2. Matching model between user product design requirements and cloud service resources

This study starts from the user product design requirement perspective, analyzes the classification of CMfg demand and service resource, and constructs the matching model to achieve fast and accurate matching of CMfg service resource and to meet the demands of cloud users. The model is a quantitative research on resource service requirement matching from five information levels, namely, base information, function information, state information, quality of service (QoS) information, and manufacturing capability information.

2.1. The classification of CMfg requirement and service resource

In the cloud service platform, the CMfg requirement classification is divided into three categories according to the type of user service demands, which include cloud design demand, CMfg demand, and cloud purchase demand (Wu et al. 2013). Specific information description is shown in Table 2.

| Category            | Demand information                                      | Feedback                                              |
|---------------------|---------------------------------------------------------|-------------------------------------------------------|
| Cloud design demand | Technical indicators (working pressure, size, and service life) | Design specification and 3D model or document package for manufacturing |
| Cloud manufacturing demand | Process parameters, process, and coordination precision | Semi-finished products, finished products, or parts   |
| Cloud purchase demand | Name, material, and price                               | Product                                                |

Product design plays a vital role in the entire production life cycle process. For enabling the cloud users to search the matching service resources accurately and effectively and to meet the design demands, so this paper takes the cloud design demand matching as the research point.

2.2. The semantic matching model of CMfg requirement and service resource

According to the cloud user design requirement, the semantic matching is conducted between the user service demand and service resource based on the cloud service platform in order to provide the most suitable resources to the cloud user. The multilayer semantic matching model of service demand and service resource is established to ensure rapid and effective matching, and the matching framework is shown in Fig. 1.

Semantic matching of CMfg service resource mainly includes three modules, namely, CMfg service resource ontology library, semantic matching, and matching result set. The specific analysis is as follows:

- CMfg service resource ontology library

Unifying the definition and semantic description to build a unified service resource ontology library for product design and development is necessary, because the CMfg service platform provides a wide range of various and multifunctional resource services. As for the ontology library that described by the Ontology Web Language, the unified and standardized definition and semantic description for resource service concept, attribute, correlation can achieve the sharing of knowledge concept in the field of resource service, and ensure the matching on the semantic level. Fig 2 shows the top ontology fragment of CMfg service resource for the design and development of auto parts (only a part of the concept is shown because of the space limitations).

The OWL description and CMfg resource model used in this paper is derived from (Yuan, Deng, and Chaovalitwongse 2017), and the resources are classified and stored according to the relevant semantic description method according to reference (Ye, L., and B. Zhang. 2007). According to the CMfg service resource ontology library, the user will discover the resource service based on the functional semantics, and then filter the service resource by matching method.
• Semantic matching between user product design demand and CMfg service resource

In the semantic matching module, the matching between user service demands and the corresponding service description in cloud service platform is conducted according to the semantic information provided by the service resource ontology library. Then, semantic similarity between information is identified, thus determining the degree of service matching. The entire function module mainly involves resource service and demand description, semantic matching method, and matching result sort. The semantic matching method is the core of the entire resource service matching, and specific algorithm will be detailed in the next section.

The matching result is listed in descending order by semantic similarity, and the user selects service according to the matching result. When selected service does not work well, its reputation will be reduced, and the recommended value in the subsequent selection process will be reduced.

If the publisher of the task chooses a service that does not conform to the actual description or one party needs to terminate the service, the terminating party sends the service change request to the cloud management platform. According to the agreement uploaded to the cloud service management platform by both parties, the transaction will be terminated after the compensation or payment is completed, and the task requester will be republish the relevant information for service matching.
• Matching result set
  From the corresponding matching algorithm, the matching degree between cloud user service demand and service resource is calculated and the service resources that meet the demand of the matching result set are added to facilitate the user reselection and reuse.

2.3. The process of multilayer matching model

This study proposes a resource service multilayer matching method based on semantic similarity, which includes the matching layers of base information, function information, state information, QoS information, manufacturing capability information, and comprehensive matching, to ensure the fast and accurate matching of the user service demand to CMfg service resource and to meet the high-level design requirements of the users. Fig. 3 shows the multilayer matching process.

As to the filtering sequence for information matching, base information is the most basic information, and the difficulty in calculating the similarity is relatively low, at the same time, it’s also the basis for the other information matching, so it is the first layer. Next, because IO and PE attributes display the main function information of the services, and they play the core role in the whole matching process, thereby the function information is regarded as the second layer. Owing to the state information is the dynamic performance of the current service status. It can quickly reflect whether the service meets the demand or not, so it is the third layer. Then, the fourth layer is the QoS information matching. Because it is studied from the numerical and interval type, the numerical represents the simplest attribute, and the other shows the interval. Besides, the QoS values are dynamic and not fixed. With the calculation and service matching, it can effectively grasp the satisfaction of the user, so it is put in the black of the state information. Finally, the manufacturing capability information indicates the ability of the enterprises to accomplish the service demands. To some extent, the same requirement can be met by many ones, and the demanders may not care not too much, so it is put in the end.

The specific process (shown in Fig. 3) is detailed as follows: First, the appropriate threshold is set according to the importance of the resource service demand performance. Then, the semantic similarity of each layer is calculated, and compared with the threshold. If the similarity is less than the threshold, then the service resource is filtered out, thus reducing the calculation of the next layer. Otherwise, the service resource is retained to enter the next matching layer. Through the calculation of each matching layer, there still exist a large number of service resources that meet the demands. Thus, conducting comprehensive matching is necessary. From the matching results of the aforementioned layers, the threshold is once again set accordingly and the comprehensive matching degree of the resource service is counted and
further filtered to obtain the best matching service resources for the user.

3. The design and realization of the multilayer matching algorithm for resource for resource service matching

3.1. Base information matching

The base information of CMfg service resource and service demand is described by resource ontology in the domain. Their names, synonyms, and other information sources are all from the concept of the ontology. Thus, the base information of resource service matching is measured by the similarity between ontology concepts. The matching algorithm is as follows:

Step 1: The base information matching model between CMfg service resource and service demand based on semantic similarity is established. The semantic description of the base information for CMfg service resource RP and user service demand RR is presented in this step. The notation of CMfg service resources offered by the provider is expressed as $R_{RP_{BaseInf}} = \langle r_{p_1}, r_{p_2}, r_{p_3}, \ldots, r_{p_n} \rangle$, which represents the base information concept collection of service resources. The notation of user service demands offered by the demander is expressed as $R_{RR_{BaseInf}} = \langle r_{r_1}, r_{r_2}, r_{r_3}, \ldots, r_{r_m} \rangle$, which represents the base information concept set offered by the service demander. Notably, $n \geq m$ are quantities of the base information concept word of resource service. From these quantities, a semantic matching model for the base information is expressed as (Yuan at al., 2017):

$$\text{Match}(RP, RR)_{BaseInf} = \sum_{i=1}^{n} \lambda_i \cdot \text{Sim}(r_{p_i}, r_{r_i})$$  \hspace{1cm} (1)

where $\text{Sim}(r_{p_i}, r_{r_i})$ represents the semantic similarity degree between $r_{p_i}$ and $r_{r_i}$, $\lambda_i (0 \leq \lambda_i \leq 1)$ is the weight of $\text{Sim}(r_{p_i}, r_{r_i})$, and $\sum_{i=1}^{m} \lambda_i = 1$.

Step 2: The semantic similarity degree is calculated. The reverse relationship between semantic similarity and semantic distance (Zhang at al., 2013) is obtained as:

$$\text{Sim}(r_{p_i}, r_{r_i}) = 1 - \text{Dis}(r_{p_i}, r_{r_i})$$  \hspace{1cm} (2)

Step 3: The semantic distance is calculated. The domain ontology of CMfg service resource is regarded as a hierarchy concept tree in a form, which has the parent/child inheritance relationship. The semantic distance between $r_{p_i}$ and $r_{r_i}$ is the sum of the concept weights, $\text{Weight}_j$, on the adjacent edge of the $r$ lines with the shortest path; that is,

$$\text{Dis}(r_{p_i}, r_{r_i}) = \sum_{j=1}^{c} \text{Weight}_j$$  \hspace{1cm} (3)

Step 4: The concept weight is calculated. From the information perspective, the deeper the level of the ontology concept, the more the information obtained. In view of the ontology in each level, the two ends of each side are regarded as a parent/child relationship. Based on Lin (1998), the quantity of the information of the father concept is defined as $I(parent(c_j))$ in the concept hierarchy tree and the child concept is taken as $I(c_j)$. This quantity is obtained as:

$$\text{Weight}_j = \frac{1}{n} \frac{I(parent(c_j)) - I(c_j)}{n'} I(parent(c_j)) + I(c_j)$$  \hspace{1cm} (4)

where $n'$ represents the number of direct child concept to parent concept. The formula used to calculate $I(c_j)$ is expressed as:

$$I(c_j) = -\log p(c_j) = -\log p(parent(c_j))$$  \hspace{1cm} (5)
where \( p(c_j) \) represents the probability of the ontology concept word \( c_j \) in any service resource ontology instance. Based on Bai (2007), the probability of each concept word node is calculated according to the ontology hierarchy using the domain ontology as the statistical source. This probability is expressed as:

\[
p(\text{root}) = 1
\]  

The concept root is the root node of the ontology. With Equations 4, 5, and 6, Equation 3 can be rewritten as:

\[
\text{Dis}(rp_i, re_i) = \frac{1}{n'} \sum_{j=1}^{n'} \frac{\log \frac{1}{n'}}{\log p^3(\text{parent}(c_j))/n'}
\]  

Step 5: The semantic similarity calculating model of base information matching is defined as:

\[
\text{Match}(\text{RP, RR})_{base} = \frac{1}{n'} \sum_{j=1}^{n'} \left( \delta_0 - \frac{1}{n'} \sum_{j=1}^{n'} \frac{\log \frac{1}{n'}}{\log p^3(\text{parent}(c_j))/n'} \right)
\]

3.2. Function information matching

In the CMfg service platform, function information matching is mainly conducted from four attributes, namely, input, output, precondition, and effect. Input refers to the required information for service execution, such as equipment manual and processing object. Output indicates the output after the completion of the specified service. Precondition represents the logical condition to be satisfied when the service is executed. Effect is the result of the successful execution of the service. These attributes are mainly measured and described by the similarity of the ontology concept in the service resource ontology domain. Therefore, the algorithm model of function information matching is expressed as:

\[
\text{Match}(\text{RP, RR})_{\text{function}} = \delta_1 \text{Sim}(\text{RP, RR})_{\text{input}} + \delta_2 \text{Sim}(\text{RP, RR})_{\text{output}} + \delta_3 \text{Sim}(\text{RP, RR})_{\text{precondition}} + \delta_4 \text{Sim}(\text{RP, RR})_{\text{effect}}
\]  

where \( \text{Sim}(\text{RP, RR})_{\text{input}}, \text{Sim}(\text{RP, RR})_{\text{output}}, \text{Sim}(\text{RP, RR})_{\text{precondition}}, \text{and} \text{Sim}(\text{RP, RR})_{\text{effect}} \) represent the semantic similarity of the four attributes in function information matching. \( \delta_1, \delta_2, \delta_3, \) and \( \delta_4 \) (\( \delta_1, \delta_2, \delta_3, \delta_4 \in [0, 1] \)) represent the weight of each function attribute, and \( \delta_1 + \delta_2 + \delta_3 + \delta_4 = 1 \).

- Input/output (I/O) matching

In the CMfg environment, the I/O parameter is the formal description for the CMfg service resource. The parameter mainly describes the concept and function attribute of service resources, and the corresponding semantic similarity calculation is conducted according to the ontology concept. Thus, the I/O parameter is calculated by the principle of the concept similarity of the base information matching. Take the input parameter set as an example.

The input parameter set of the resource service description released from the cloud service platform is defined as \( \text{InPS} \). The input parameter set of the user product design demand description is defined as \( \text{InPD} \).

The similarity calculating principle of input parameter matching is described as follows: The similarity calculation starts from the element \( \text{InPS}_i \) to \( \text{InPS}_j \) in \( \text{InPS} \) with \( \text{InPD}_j \) in \( \text{InPD} \). Through the calculation, the best \( \text{InPS}_i \) with the maximum semantic similarity can be obtained.

From the similarity calculating principle, the semantic similarity matching algorithm of the input parameter set is expressed as:

\[
\text{Sim}(\text{RP, RR})_{\text{input}} = \text{Match}(\text{InPS}, \text{InPD}) = \sum_{j=1}^{n_i} \frac{1}{n_i} \text{Sim}(\text{InPS}_i, \text{InPD}_j)
\]  

where \( \text{InPS}_{ij} \) represents the \( j \)th input parameter released by the CMfg service resource. \( \text{InPD}_{ij} \) denotes the \( j \)th input parameter released by the resource service demand. \( n_i \) is the number of all the parameter components of the input.
Similarly, the semantic similarity matching algorithm of the output parameter set is obtained as:

\[
\text{Sim}(\text{RP}, \text{RR})_{\text{output}} = \text{Match}(\text{OutPS}, \text{OutPD}) = \sum_{j=1}^{n_o} \frac{1}{n_o} \text{Sim}(\text{OutPS}_{\text{outj}}, \text{OutPD}_{\text{outj}})
\]  

(11)

where \(\text{OutPS}_{\text{outj}}\) represents the \(j\)th output parameter released by the CMfg service resource. \(\text{OutPD}_{\text{outj}}\) denotes the \(j\)th output parameter released by the resource service demand. \(n_o\) is the number of all the parameter components of the output. Specifically, the semantic similarity calculation of I/O matching can be achieved according to the principle of the base information matching.

- Precondition and effect (P/E) matching

The attribute parameter of P/E is not a concept in the service resource ontology domain, only a kind of constraint expression that composed of the object, predicate, and parameter. For example, the maximum power speed is 6,000 rpm to 6,500 rpm. The object is “the maximum power speed” the predicate is “is,” and the parameter is “6,000 rpm to 6,500 rpm.” Therefore, the constraint expression must be decomposed in this kind of matching calculation. The specific dismantling is defined as:

\[
(\text{CMSRorRD})_{\text{P/E}} = \langle \text{object}, \text{predicate}, \text{parameter} >
\]

where \((\text{CMSRorRD})_{\text{P/E}}\) is the CMfg service resource or resource demand; that is, the P/E attribute constraints of the CMfg service resource (or service demand).

In the entire constraint expression, the object and predicate are from the class, attribute, and instance of the CMfg service resource ontology domain for user product design. Thus, Equation 8 can be used to calculate the semantic matching degree. And the calculation of the attribute “parameter” can be obtained by the numerical interval matching algorithm shown in Equation 12:

\[
\text{Sim}(pa_{\text{rk}}, pa_{\text{rp}}) = \begin{cases} 
1 & \text{if } (pa_{\text{rk}} \cap pa_{\text{rp}} = pa_{\text{rk}}) \\
\frac{|pa_{\text{rk}} \cap pa_{\text{rp}}|}{|pa_{\text{rk}}|} & \text{if } (pa_{\text{rk}} \cap pa_{\text{rp}} \neq \emptyset \text{ and } pa_{\text{rk}} \cap pa_{\text{rp}} \neq pa_{\text{rk}}) \\
0 & \text{if } (pa_{\text{rk}} \cap pa_{\text{rp}} = \emptyset) 
\end{cases}
\]  

(12)

where \(pa\) is the abbreviation of \textit{parameter}, \(pa_{\text{rk}}\) is the attribute numerical interval constraint of user design demand. \(pa_{\text{rp}}\) is the attribute numerical interval constraint of the CMfg service resource. “||” represents the length of the parameter interval domain, for example, \(|60, 70| = 10\).

For the matching calculation of the entire constraint expression, if and only if the object matching degree is greater than the threshold and the predicate has a correct match, then the similarity matching calculation for the parameter would be conducted. The sum of the matching degree of the three values is the matching degree of the entire sentence. Accordingly, the similarity matching algorithm of the P/E attribute for the CMfg service resource and user product design demand is obtained by using Equation 13:

\[
\text{Sim}(\text{RP}, \text{RR})_{\text{P/E}} = \alpha_1 \text{Sim}(\text{RP}, \text{RR})_{\text{object}} + \alpha_2 \text{Sim}(\text{RP}, \text{RR})_{\text{predicate}} + \alpha_3 \text{Sim}(\text{RP}, \text{RR})_{\text{parameter}}
\]  

(13)

where \(\alpha_1, \alpha_2, \alpha_3 (\alpha_1, \alpha_2, \alpha_3 \in [0, 1])\) represent the weight of each attribute constraint in the matching process and \(\alpha_1 + \alpha_2 + \alpha_3 = 1\).

3.3. State information matching

State information refers to the current status of the CMfg service resource, whether it exists or is available for the immediate use in place, which means the possibility of the CMfg service resource available for selection (Yin, Zhang, and Zhong 2012). State information matching is described in the form of the available \(A(S_k)\) to express resource service
matching directly. From the point of using time, the definition is described as:

\[ A(S_k) = \frac{T(S_k)}{t} = \begin{cases} \geq 1 & \text{available} \\ < 1 & \text{unavailable} \end{cases} \]  \quad (14)

where \( T(S_k) \) represents the total time required for using the service resource \( S_k \) within the recent \( t \) times. The “available” and “unavailable” forms that represent the “idle” and “working” status are defined. Then, the state information similarity matching algorithm is expressed as:

\[ \text{Match}(RR, RP)_{\text{StateInf}} = \text{Sim}(T(S_k)_{RR}, T(S_k)_{RP}) = \begin{cases} 1 & A(S_k) \geq 1 \\ 0 & A(S_k) < 1 \end{cases} \]  \quad (15)

where \( T(S_k)_{RR} \) represents the total time required for using the service resource within the recent \( t \) period. \( T(S_k)_{RP} \) denotes the total time provided by the service resource within the recent \( t \) times.

### 3.4. QoS information matching

QoS information refers to the degree of the resource services provided by the cloud service platform to meet the user product design demands, and it’s also an important reference for selecting the cloud service. When a service resource is registered in the cloud service platform, the corresponding evaluating criteria for services will be defined, including time, cost, and credibility, to help each party measure the quality of cloud services.

- **Time and cost matching**
  The criterion of time and cost for resource service are mainly numerical parameters, and can be referred to the numerical matching algorithm expressed in Equation 16:

\[ \text{Match}(RR, RP)_{T/C} = \text{Sim}(T_1/C_1, T_2/C_2) = \begin{cases} 1 & \text{when meet the condition} \\ 0 & \text{not meeting} \end{cases} \]  \quad (16)

The condition means, for example, the time/cost of the service demand is expressed as \( T_1/C_1 = \{"10 \text{ days"},"3,000 \text{ Yuan"}\} \) and the finished time/cost provided by the resource service is described as \( T_2/C_2 = \{"9.5 \text{ days"},"2,850 \text{ Yuan"}\} \). That is, when the time/cost is less than or equal to the service demand, the matching degree is 1; otherwise, 0.

- **Credibility matching**
  For credibility evaluation, different cloud services and various evaluation indicators are involved, which are related to software, knowledge, and human resource. The evaluation indices of software resource service include compatibility, operability and interfacefriendliness, etc. Knowledge resource service possesses the evaluating criteria of processing capability, craft, and failure rate. Human resource service evaluation indices contain innovativeness, team cooperation ability, and service level. The credibility of the service is defined in Table 3.

| Credibility | Excellent | Very good | Good | Fair | Not good |
|-------------|-----------|-----------|------|------|----------|
| Value \( Q \) | \geq 4.5 | 4 \leq Q < 4.5 | 3.5 \leq Q < 4 | 3 \leq Q < 3.5 | <3 |

The function of evaluating the resource service credibility is defined as:

\[ Q(S_i) = \frac{\sum_{i=1}^{3} \sum_{j=1}^{N} Q_i^j}{N_Q} \]  \quad (17)

where \( i = 1, 2, \) and 3 represent software, knowledge, and human resource, respectively. \( N \) denotes the evaluation index number of the resource service \( S_i \), \( (j=1, 2, 3, \ldots, N_i) \) is a numerical parameter. \( Q_i^j \) indicates the evaluation of the \( j \)th index to the required resource service after being obtained by the user. \( N_Q \) shows the total number of service evaluation.
Through the calculation and conversion of the service evaluation, the matching of the resource service credibility can be achieved. The numerical matching algorithm is shown in Equation 18.

\[
\text{Match}(RR, RP)_R = \text{Sim}(C_{RR}, C_{RP}) = \begin{cases} 
1 & \text{when meet the condition} \\
0 & \text{not meeting}
\end{cases}
\] (18)

The condition means, for example, that the credibility of the user product design demand \(C_{RR}\) is “4.8” or “excellent” and the value provided by the CMfg service resource is 4.9. That is, if \(4.9 > 4.8\), then the condition is met and the matching degree is 1; otherwise, 0.

### 3.5. Manufacturing capability information matching

The manufacturing capability of the CMfg resource service means the ability of completing the user service demand. This study mainly concentrates on the working precision of the product and service maintenance.

- **Working precision matching**

  Working precision means the accordance between the actual geometric parameters of the processed product and the desired geometric parameters provided by the user service demand. The parameters include size, shape, and location. Different geometric parameters correspond to different measures. Thus, the working process matching between the user required precision \(Cap_{RR}\) and the \(Cap_{RP}\) provided by CMfg resource service is defined as:

\[
\text{Sim}(Cap_{RR}, Cap_{RP}) = \sum_{i=1}^{3} \sum_{j=1}^{N_c} \text{Sim}(Cap_{RR}^i, Cap_{RP}^j)
\] (19)

where \(i = 1, 2, 3\) means the size, shape, and location of three geometric parameters, respectively. \(Cap_{RP}^j\) indicates the precision demand of the \(i\)th geometric parameter for user product. \(Cap_{RP}^j\) represents the \(j\)th measure index of the \(i\)th geometric parameter provided by the CMfg service resource. \(N_c\) represents the total number of measures of three geometric parameters.

This property is mainly the numerical interval parameter; its matching degree of calculation is shown in Equation 12. If the required precision level exists in the service level interval, then the matching degree is 1; otherwise, 0.

- **Service maintenance matching**

  Service maintenance means the probability that the services provided by the cloud service platform are properly maintained in the event of failure. It is an important index to reflect the manufacturing capability of the CMfg resource service, and its value is derived as:

\[
P_{Ma} = \frac{S_{Ma}}{N_{Ma}}
\] (20)

Where \(S_{Ma}\) means the total number of successful maintenance in the event of failure according to a certain product design demand for the CMfg resource service. \(N_{Ma}\) represents the total number of the failures.

The calculation of the matching degree of service maintenance refers to the numerical matching algorithm shown in Equation 21:

\[
\text{Sim}(Ma_{RR}, Ma_{RP}) = \begin{cases} 
1 & P_{Ma} \geq \eta_{Ma} \\
0 & P_{Ma} < \eta_{Ma}
\end{cases}
\] (21)

where \(\eta_{Ma}\) represents the threshold set by the user service demand.

In summary, the similarity matching algorithm for manufacturing capability is described as:

\[
\text{Match}(RR, RP)_{\text{CapabilityInf}} = \beta_1 \text{Sim}(Cap_{RR}, Cap_{RP}) + \beta_2 \text{Sim}(Ma_{RR}, Ma_{RP})
\] (22)

where \(\beta_1\) and \(\beta_2\) (\(\beta_1, \beta_2 \in [0,1]\)) represent the weight of the working precision and service maintenance respectively, and \(\beta_1 + \beta_2 = 1\).
3.6. Comprehensive matching

Through information matching, the service resources that meet the conditions still exist in a large number. Conducting comprehensive matching and further filtering of the matching result is necessary to achieve the best matching resource service. Then the service resources that are most consistent with the user demands will be obtained.

The weights of the five information levels matching $\xi_i$ ($i = 1, 2, \ldots, 5$) are defined. The comprehensive matching degree calculation is defined as:

$$
\text{Match}(RR, RP)_{\text{Comprehensive}} = \xi_1 \text{Match}(RR, RP)_{\text{BaseInf}} + \xi_2 \text{Match}(RR, RP)_{\text{FunctionInf}} + \xi_3 \text{Match}(RR, RP)_{\text{StateInf}} + \xi_4 \text{Match}(RR, RP)_{\text{QoSInf}} + \xi_5 \text{Match}(RR, RP)_{\text{CapabilityInf}}
$$

(23)

where $\xi_i \in [0, 1]$ and $\sum_{i=1}^{5} \xi_i = 1$. The determination of the weight will be elaborated in the next section.

4. The determination of the weight of the resource service information matching module

Weight aimed at a specific index is a relative concept and it reflects the importance of the index in the overall evaluation system (Cheng 2006). In the CMfg environment, each resource service corresponds to a number of service information indicators. The user focuses varied attentions on the function of service resource, and the demand is not the same. Therefore, it’s necessary to determine the proportion of each information indicator according to the importance of different service requirements.

In the determination of the weight in similarity calculation (Yang, and Lin 2003), the analytic hierarchy process is usually used to solve the process of multi-objective and multi-information factors. However, in the traditional analytic hierarchy process (Wei, Li, and Li 2005), the traditional hierarchical quantitative method is used to evaluate the weight judgment matrix. It’s easy to cause one factor to be five, seven, and even nine times as much as the other one in the same evaluation system, and it affects the rationality of weight. Meanwhile, the factor should be adjusted if the judgment matrix is unsatisfied in the consistency check, thus affecting the processing efficiency. Therefore, this study proposes an I-AHP method, which adopts the three-scale method instead of the nine-scale method, and doesn’t require the consistency checking with the optimal transmit matrix, and avoids the blindness of the judgment matrix. In addition, the scale arithmetic (d-scale) is adopted in the judgment matrix, which can effectively adjust and weaken the influence of each judgment factor. Thus, the specific steps are as follows:

1. Establish the hierarchical structure
2. Construct the comparison matrix (or precedence relation matrix)

### Table 4 Three-scale method

| Scale | Definition of importance | Explanation |
|-------|--------------------------|-------------|
| 2     | $C(i) > C(j)$            | Factor vi is more important than factor vj |
| 1     | $C(i) = C(j)$            | Factor vi and factor vj are equally important |
| 0     | $C(i) < C(j)$            | Factor vi is less important than factor vj |

$C(i)$ and $C(j)$ represent the relative importance of factor vi and vj, respectively.

From the existing researches, this study adopts the three-scale method (Li at al., 2004) in constructing the comparison matrix. Details are shown in Table 4. This method not only reduces the difficulty of expert judgment but also lowers the influence of human subjectivity on the evaluation results. Meanwhile, when the comparison matrix is transformed into the judgment matrix, it also meets the demand of consistency checking and calculation accuracy.

- Each information factor is compared with the others and the comparison matrix $V = (v_{ij})_{m \times m}$ is obtained as:

$$
V = (v_{ij})_{\text{com}} = \begin{bmatrix}
\begin{array}{cccc}
v_{11} & v_{12} & \cdots & v_{1m} \\
v_{21} & v_{22} & \cdots & v_{2m} \\
\vdots & \vdots & \ddots & \vdots \\
v_{m1} & v_{m2} & \cdots & v_{mm}
\end{array}
\end{bmatrix}
$$

where m is the order of the comparison matrix, which is equal to the number of the service information description types.

- The sums of each row in the comparison matrix are obtained in Table 5.

[DOI: 10.1299/jamdsm.2019jamdsm0085] © 2019 The Japan Society of Mechanical Engineers
In Table 5,

\[ s_i = \sum_{j=1}^{m} v_{ij} \quad (i = 1, 2, \ldots, m) \]  

(24)

(3) Build the judgment matrix

Definition: \( s_{\text{max}} = \max \{s_i\} \) is the maximum sort index of the corresponding factor, \( s_{\text{min}} = \min \{s_i\} \) is the minimum sort index of the corresponding factor, and \( c_m = s_{\text{max}}/s_{\text{min}} \) represents the importance between \( s_{\text{max}} \) and \( s_{\text{min}} \). These parameters are used to build the judgment matrix \( D = (d_{ij})_{m \times m} \):

\[
d_{ij} = \begin{cases} 
\frac{s_i - s_j}{s_{\text{max}} - s_{\text{min}}} \left[ c_m - d(\text{scale}) \right] + 1 = \frac{s_i - s_j + s_{\text{min}}}{s_{\text{min}}} \quad (s_i \geq s_j) \\
\frac{1}{s_{\text{min}}} \left( s_i - s_j \right) \left[ c_m - d(\text{scale}) \right] + 1 = \frac{s_{\text{min}}}{s_i - s_j + s_{\text{min}}} \quad (s_i < s_j)
\end{cases}
\]  

(25)

where \( d(\text{scale}) \) is the scale arithmetic value of the three-scale method; \( d(\text{scale}) = 1 \) in this method. Furthermore, \( i, j = 1, 2, \ldots, m \).

(4) Establish the quasi-optimal consistency matrix

- The transmit matrix \( H \) for judgment matrix \( D \) is solved.
  
  From the theorem of the transmit matrix (Niu, Chen, and Cheng 2011), the matrix is defined as:

\[
h_{ij} = \begin{cases} 
\log \left( \frac{s_i - s_j + s_{\text{max}}}{s_{\text{min}}} \right) & s_i \geq s_j \\
\log s_{\text{min}} & s_i < s_j
\end{cases}
\]  

(26)

If the transmit matrix \( H' \) exists, then \( \sum_{i=1}^{m} \sum_{j=1}^{m} (h_{ij} - h_{ij})^2 \) can be regarded as the minimum. \( H' \) is called the optimum transmit matrix for \( H \). From the theorem of optimum transmit matrix (Niu, Chen, and Cheng 2011), the following equation is obtained:

\[
h_{ij} = \frac{1}{m} \sum_{k=1}^{m} (h_{ik} - h_{jk}) = \frac{1}{m} \times [\text{Sum}(h_{ij}) - \text{Sum}(h_{ik})] \quad (i, j = 1, 2, \ldots, m)
\]  

(27)

- The quasi-optimal consistency matrix \( D' = (d'_{ij})_{m \times m} \) for judgment matrix \( D \) is obtained, and \( d'_{ij} = 10^{h_{ij}} \).

(5) Calculate the weight

- The product of each element in a row of matrix \( D' \) is counted and the \( m \)th root is calculated. The equation is expressed as:

\[
w_i = \sqrt[m]{\prod_{j=1}^{m} d'_{ij}} \quad (i = 1, 2, \ldots, m)
\]  

(28)

- The normalized vector \( w' = (w'_1, w'_2, \ldots, w'_m)^T \) is obtained by the following equation:

\[
w_i = w_i / \sum_{i=1}^{m} w_i
\]  

(29)
The vector $\mathbf{w} = (w_1, w_2, \ldots, w_m)^T$ is the required weight vector.

### 5. Example verification

An instance of auto part design is used to illustrate the service matching process specifically to verify the feasibility and effectiveness of the proposed method.

First, the demand of a user product design service A is determined as the basis for matching of the CMfg service resource. Then, the matching degree with the candidate service resources B and C searched from the cloud service pool is calculated. The threshold for each information layer is set as 0.75, 0.70, 0.70, 0.65, and 0.65. With the matching calculation of each information layer, the optimum matching service resource will be obtained. The specific information of A, B, and C is described in Table 6.

#### Table 6 The information description of product design demand A and service resources B and C

| Matching period | Content       | Design demand A | Service resource B | Service resource C |
|-----------------|---------------|-----------------|--------------------|--------------------|
| Base information matching | Name          | Automatic transmission (AT) | Electric drive transmission | Fixed shaft manual transmission |
|                  | Synonym       | None            | None               | None               |
| Input            | Product design parameter | Design drawings and instructions | Design drawings and instructions | Design drawings |
| Output           | 1. Maximum power speed is 5,500 rpm to 6,500 rpm 1. Maximum gradeability is 55° to 65° | 1. Maximum power speed is 5,000 rpm to 6,000 rpm 1. Maximum gradeability is 50° to 60° | 1. Maximum power speed is 4,000 rpm to 4,500 rpm 1. Maximum gradeability is 35° to 40° |
| Function information matching | Precondition | 1. Gear number is 5 to 6 2. Transmission ratio is 4 to 5 3. Centerline spacing is 65 mm to 80 mm | 1. Gear number is 5 to 6 2. Transmission ratio is 4 to 5 3. Centerline spacing is 70 mm to 80 mm | 1. Gear number is 6 to 8 2. Transmission ratio is 5 to 7 3. Centerline spacing is 80 mm to 150 mm |
| State information matching | Idle/working | The next 3 days | The next 5 days (idle) | The next 10 days (idle) |
| Quality of service information matching | Time | 2015.6.1–2015.6.3 | 2015.6.1–2015.6.2 | 2015.6.1–2015.6.3 |
| | Credit | 900 Yuan | 850 Yuan | 800 Yuan |
| Manufacturing capability information matching | Working precision | Dimensional tolerance (+0.065 mm) | Precision class IT5-8 | Precision class IT6-8 |
| | Maintenance | 0.95 | 0.98 | 0.97 |

From the matching method proposed in this study, research on the matching between product design demand and CMfg resource service is conducted. Particularly, the matching between A and B is taken as an example.

In view of the demand of the transmission design, the ontology fragment of cloud service resource ontology domain is obtained combined with the auto parts ontology domain in the CMfg resource service (Yuan at al., 2017) (as shown in Fig. 4).
The occurrence probability, quantity of information, and adjacent weight for each node are calculated. Then, the transmission ontology fragment hierarchy tree can be obtained. The specific process is as follows:

- First, the occurrence probability and quantity of information for root node—power train system (PTS) is calculated:
  \[
  p(PTS) = 1, \quad I(PTS) = -\log(p(PTS)) = 0
  \]
- Then, the occurrence probability, quantity of information, and node adjacent weight for the transmission node are calculated:
  \[
  \begin{align*}
  p(\text{transmission}) &= \frac{1}{5} \times p(PTS) = \frac{1}{5} \\
  I(\text{transmission}) &= -\log(p(\text{transmission})) = \log 5 \\
  \text{Weight}(\text{transmission}, PTS) &= \frac{1}{5} \times (I(\text{transmission}) - I(PTS)) = 0.2
  \end{align*}
  \]
- The node MT is calculated, with transmission as its father node. The formula is expressed as:
  \[
  \begin{align*}
  p(MT) &= \frac{1}{3} \times p(\text{transmission}) = \frac{1}{15} \\
  I(MT) &= -\log(p(MT)) = \log 15 \\
  \text{Weight}(MT, \text{transmission}) &= \frac{1}{3} \times \frac{I(MT) - I(\text{transmission})}{I(MT) + I(\text{transmission})} = 0.085
  \end{align*}
  \]

Given the space limitation, each step of the calculation is not listed. The final results are shown in Fig. 5. The deeper the node, the smaller the adjacent weight will be; the greater the quantity of information, the higher the corresponding similarity.

Then, the matching algorithm in this paper is described in detail as follows:

1. Base information matching

The definition of synonym is “none”; thus, it only needs to conduct name matching. Therefore, \(Sim(A, B)_{\text{BaseInf}} = 1 - \text{Dis}(A, B) = 1 - 0.046 = 0.954\).
(2) Function information matching

- I/O matching

The essence of I/O matching is measured by the matching of the ontology concept word derived from the concept of the CMfg service resource ontology. A and B are the same. Thus, $Sim(A, B)_{input} = 1$ and $Sim(A, B)_{output} = 1$.

- P/E matching

First, the decomposition of the constraint statement in accordance with the rule $< object, predicate, parameter >$ is conducted. The object and predicate have an exact match; thus, the matching degree is 1. The parameter is calculated by using Equation 12. Condition 1 is taken as an example (in general, the weights are defined as equal):

$$Sim(A, B)_{p1} = \frac{1}{3} \times Sim(A, B)_{object} + \frac{1}{3} \times Sim(A, B)_{predicate} + \frac{1}{3} \times Sim(A, B)_{parameter}$$

$$= \frac{1}{3} \times 1 + \frac{1}{3} \times 1 + \frac{1}{3} \times Sim([5,500, 6,500], [5,000, 6,000])$$

$$= \frac{5}{6}$$

Similarly, $Sim(A, B)_{p2} = \frac{5}{6}$.

Thus, the matching degree of the precondition is derived as:

$$Sim(A, B)_{precondition} = \frac{1}{2} \times Sim(A, B)_{p1} + \frac{1}{2} \times Sim(A, B)_{p2} = \frac{5}{6} = 0.833$$

Similarly,

$$Sim(A, B)_{effect} = \frac{1}{3} \times Sim(A, B)_{e1} + \frac{1}{3} \times Sim(A, B)_{e2} + \frac{1}{3} \times Sim(A, B)_{e3}$$

$$= \frac{1}{3} \times 1 + \frac{1}{3} \times 1 + \frac{1}{3} \times \left[ \frac{1}{3} + \frac{1}{3} + \frac{1}{3} \times \frac{70 - 80}{80 - 65} \right]$$

$$= 0.963$$

In summary, the function information matching degree is obtained as:

$$Sim(A, B)_{FunctionInf} = \frac{1}{4} \times Sim(A, B)_{input} + \frac{1}{4} \times Sim(A, B)_{output}$$

$$+ \frac{1}{4} \times Sim(A, B)_{precondition} + \frac{1}{4} \times Sim(A, B)_{effect}$$

$$= 0.949$$
(1) State information matching

Table 9 shows the demand for the design service in the next three days. However, service resource B can provide the design service in the next five days. \( T(\text{Sim}) = \frac{5}{3} = 1.67 > 1 \); thus, the matching degree is 1. That is, \( \text{Sim}(A,B)_{\text{StateInf}} = 1 \).

(2) QoS information matching

- Time and cost matching

In Table 9, from the time to meet the requirement of the service perspective, the constraint information description of the time and cost for the service demand A and the service resource B is simplified as TA/CA = \{"3 days","900 Yuan"\}, TB/CB = \{"2 days","850 Yuan"\}. The service resource B meets the constraints of time and cost; thus, \( \text{Sim}(A,B)_T = 1 \) and \( \text{Sim}(A,B)_C = 1 \).

- Credibility matching

Credibility evaluation involves a variety of resources corresponding to multiple indicators. With the established model \( Q(S_i) = \frac{\sum Q(I)_{i}}{N_Q} \), the cloud service platform directly provides credibility for the corresponding service resource. As shown in Table 9, \( Q(S_A) = 4.8 \), \( Q(S_B) = 4.9 \), and \( Q(S_B) > Q(S_A) \). Therefore, the matching degree is 1.

In summary, the matching degree of QoS information can be obtained as:

\[
\text{Sim}(A,B)_{QoS} = \frac{1}{3} \times \text{Sim}(A,B)_T + \frac{1}{3} \times \text{Sim}(A,B)_C + \frac{1}{3} \times \text{Sim}(A,B)_Q = 1
\]

(3) Manufacturing capability information matching

- Working precision matching

Table 9 only provides the size tolerance of service demand A, which is ±0.065 mm, to simplify the equation. The processing capacity IT5-8 is provided by service resource B, which can satisfy the demands. Thus, the matching degree is 1.

- Service maintenance matching

From the user feedback on the usage of service resources, with the calculation of the \( \rho_{Ma} = \frac{S_{Ma}}{N_{Ma}}_{\text{model}} \), the cloud service platform provides the maintenance. As shown in the Table 9, \( \rho_{Ma} = 0.98 > (\eta_{Ma})_{A} \); therefore, the matching degree is 1.

In conclusion, the matching degree of manufacturing capability information is derived as:

\[
\text{Sim}(A,B)_{\text{CapabilityInf}} = \frac{1}{2} \times \text{Sim}(A,B)_{\text{precision}} + \frac{1}{2} \times \text{Sim}(A,B)_{Ma} = 1
\]

(4) Comprehensive matching

Considering all the previous presented information matching, the following expression can be obtained:

\[
\text{Sim}(A,B)_{\text{Comprehensive}} = \xi_1 \times \text{Sim}(A,B)_{\text{BaseInf}} + \xi_2 \times \text{Sim}(A,B)_{\text{FunctionInf}} + \xi_3 \times \text{Sim}(A,B)_{\text{StateInf}} + \xi_4 \times \text{Sim}(A,B)_{QoS} + \xi_5 \times \text{Sim}(A,B)_{\text{CapabilityInf}}
\]

\[
= 0.954\xi_1 + 0.949\xi_2 + \xi_3 + \xi_4 + \xi_5
\]

Determining the weight for each of the information matching module in comprehensive matching based on I-AHP is specified as follows:

Through the consultation with the expert on CMfg service matching, the importance of the five information types is compared. And the results are shown in Table 7.

| Table 7 The correlation of each information factor in the comparison matrix |
| --- |
| | Base information | Function information | State information | QoS information | Manufacturing capability information | S |
| --- |
| Base information | 1 | 2 | 2 | 2 | 2 | 9 |
| Function information | 0 | 1 | 2 | 2 | 2 | 7 |
| State information | 0 | 0 | 1 | 2 | 2 | 5 |
| QoS information | 0 | 0 | 0 | 1 | 2 | 3 |
| Manufacturing capability information | 0 | 0 | 0 | 0 | 1 | 1 |

From Table 7, the comparison matrix \( V \) can be obtained:
Through the comparison matrix $V$, $s_{max} = 9$ and $s_{min} = 1$ are obtained. From Equations 25, 26, and 27, the judgment matrix $D$, transmit matrix $H$, optimum transmit matrix $H'$, and quasi-optimal consistency matrix $D'$ are established and specified as follows:

$$D = (d_{ij})_{5 \times 5} = \begin{bmatrix}
1 & 3 & 5 & 7 & 9 \\
1/3 & 1 & 3 & 5 & 7 \\
1/7 & 1/5 & 1/3 & 1 & 3 \\
1/9 & 1/7 & 1/5 & 1/3 & 1 \\
0 & 0.477 & 0.699 & 0.845 & 0.954
\end{bmatrix}$$

$$H = (k_{ij})_{5 \times 5} = \begin{bmatrix}
0 & -0.477 & 0 & 0.477 & 0.699 & 0.845 & 0.954 \\
-0.699 & -0.477 & 0 & 0.477 & 0.699 \\
-0.845 & -0.699 & -0.477 & 0 & 0.477 \\
-0.954 & -0.845 & -0.699 & -0.477 & 0 \\
0 & 0.286 & 0.595 & 0.904 & 1.190
\end{bmatrix}$$

$$H' = (k'_{ij})_{5 \times 5} = \begin{bmatrix}
0 & -0.286 & 0.309 & 0.618 & 0.904 \\
-0.595 & -0.309 & 0 & 0.309 & 0.595 \\
-0.904 & -0.618 & -0.309 & 0 & 0.286 \\
-1.190 & -0.904 & -0.595 & -0.286 & 0 \\
0 & 0.491 & 1 & 2.037 & 3.936
\end{bmatrix}$$

$$D' = (d'_{ij})_{5 \times 5} = \begin{bmatrix}
1 & 1.932 & 3.936 & 8.017 & 15.488 \\
0.518 & 1 & 2.037 & 4.150 & 8.017 \\
0.254 & 0.491 & 1 & 2.037 & 3.936 \\
0.125 & 0.241 & 0.491 & 1 & 1.932 \\
0.065 & 0.125 & 0.254 & 0.518 & 1
\end{bmatrix}$$

Calculate the element product of each row of matrix $D'$, then compute the root of five. According to the Equation 28, $w'_1 = \sqrt[5]{\prod_{j=1}^{5} d'_{1j}} = 3.936$. Similarly, $w'_2 = 2.037$, $w'_3 = 1$, $w'_4 = 0.491$, and $w'_5 = 0.255$. Then, the vector $w'$ can be regarded as $w' = (w'_1, w'_2, w'_3, w'_4, w'_5)^T = (3.936, 2.037, 1, 0.491, 0.255)^T$.

According to the formula 29, $w = (w_1, w_2, w_3, w_4, w_5)^T = (0.510, 0.263, 0.130, 0.064, 0.033)^T$ can be obtained by conducting the normalized processing for vector $w'$. That is, the corresponding weight is $\xi_1 = 0.510$, $\xi_2 = 0.263$, $\xi_3 = 0.130$, $\xi_4 = 0.064$, and $\xi_5 = 0.033$.

Therefore, the comprehensive matching degree can be obtained as:

$$\text{Sim}(A, B)_{\text{Comprehensive}} = 0.954\xi_1 + 0.949\xi_2 + \xi_3 + \xi_4 + \xi_5 = 0.963$$

Similarly, the matching degree between A and C can be obtained. The matching results are shown in Table 8.

In the Table 8, the matching degree between A and B is greater than that between A and C. The table also shows the specific content and corresponding degree of each information matching layer, which can achieve the effective ranking of each selected service resource. From the entire process, the matching between the CMfg service resource and the user service demand is transformed into a quantitative calculation. The experimental results also show that the proposed method can effectively distinguish the matching degree between different service demands and CMfg service resources. Thus, it's easy to achieve the rapid matching, and improve the efficiency and accuracy.
Table 8 Matching results between service demand A and service resources B and C

| Matching period          | Matching content | Matching result between A and B | Matching result between A and C |
|--------------------------|------------------|-------------------------------|-------------------------------|
|                          |                  | Matching degree               | Total comparison              | Matching degree               | Total comparison              |
| Base information matching| Name             | 0.954                         | The total degree is 0.954 and is greater than the threshold 0.75 | 0.773                         | The total degree is 0.773 and is greater than the threshold 0.75 |
| Function information matching| Input             | 1                             | The total degree is 0.949 and is greater than the threshold 0.5 | 0.667                         | The total degree is 0.709 and is greater than the threshold 0.70 |
|                          | Output           | 0.833                         |                               | 0.667                         |                               |
|                          | Precondition     | 0.963                         | The total degree is 0.70 and is greater than the threshold 0.70 | 0.667                         | The total degree is 0.709 and is greater than the threshold 0.70 |
|                          | Effect           |                               |                               |                               |                               |
| State information matching| Idle/working     | 1                             | The total degree is 1 and is greater than the threshold 0.70 | 1                             | The total degree is 1 and is greater than the threshold 0.70 |
| Quality of service information matching| Time             | 1                             | The total degree is 1 and is greater than the threshold 0.65 | 1                             | The total degree is 0.667 and is greater than the threshold 0.65 |
|                          | Cost             |                               |                               | 0                             |                               |
|                          | Credit           |                               |                               |                               |                               |
| Manufacturing capability information matching| Working precision | 1                             | The total degree is 1 and is greater than the threshold 0.65 | 1                             | The total degree is 1 and is greater than the threshold 0.65 |
|                          | Maintenance      |                               |                               |                               |                               |
| Comprehensive matching   |                  | 0.963                         |                               | 0.786                         |                               |

At the same time, the proposal is verified by 300 cloud service resources collected by the web crawler tool, which correspond to three design demand: A1, A2 and A3. Use the classification matching method in reference 28, the matching method of this paper based on AHP (Amandeep at al., 2015) and the proposed method based on I-AHP in this paper to match the service resources, and count the number of service resources correspond to the design demand A1, A2 and A3 under three methods whose comprehensive matching greater than 0.5. In order to judge the quality of the match, recall rate and precision is used as the index of service matching. The recall rate refers to the ratio of correct matching resources to actual service resources, and precision refers to the ratio of correct matching resources to matching resources. The corresponding results are shown in Table 9.

Table 9 Matching results under three methods

| Design demand | Number of service resources | Classification matching method | The proposed method based on AHP | The proposed method based on I-AHP |
|---------------|------------------------------|-------------------------------|---------------------------------|-----------------------------------|
|               |                              | Matching resources Correct resources | Matching resources Correct resources | Matching resources Correct resources |
| A1            | 100                          | 102 70                         | 96 65                          | 117 83                           |
| A2            | 100                          | 98 67                         | 108 70                         | 113 79                           |
| A3            | 100                          | 105 74                         | 95 64                          | 106 76                           |
| Average       |                              | 105 70.33                      | 96 69.16                       | 66.33 69.16                      | 70.85 |

As can be seen from Table 9, the precision of the three matching methods is approximate, but the matching method based on I-AHP has a better recall rate than other methods. The main reason is that the demands and the service resources are processed by modularization, and hierarchical matching is carried out according to the characteristics of each demand and service. The recommended value of knowledge which meets the user's requirements will be fed back to the users, and the user can choose the service by himself, which satisfies the user's personalized service demand.
6. Conclusions and future works

For fast and efficient service matching and promoting CMfg implementation, efforts have been concentrated on these issues. Several methods have been discussed and presented. The main innovative contributions and specific works of this study are as follows: From the matching of the user service demands and CMfg resource services, a multilayer matching method for CMfg service is proposed. Based on the analysis of the classification of the CMfg services and the semantic description, the matching model of CMfg service resource and user design service demand is established. In addition, the matching process of the model is divided into six stages. The specific matching content and similarity algorithm for each layer are designed as well. Furthermore, an I-AHP method is proposed to solve the weight problem in comprehensive matching. Finally, a case study is used to illustrate and validate the proposed method.

With these works, service matching was transformed into a quantitative calculation, which exhibits an effective and rapid role in distinguishing different CMfg service resources. Simultaneously, the proposed I-AHP method can better solve the weight problem, scientifically express the influence of each layer on its matching goal, and be more in line with the actual service demand situation.

This study represents a preliminary attempt, and further works should be addressed in future implementation. For example, (a) further study on the asymmetric information for service matching, (b) extension for the method, through the interaction between the services to obtain the resource service reliability, and also make the semantic Web service matching into a trusted system, further improve the precision and enhance the superiority of this method, (c) applying the method to the automatic combination of semantic Web services, and design an efficient service composition, (d) function information matching should be not only for static specification but also dynamic capabilities.

Acknowledgement

This work was supported by the Humanities and Social Sciences Planning Fund of the Ministry of Education under Grant number 17YJA630127, the Fundamental Research Funds for the Central Universities under Grant number 2019B21814, the National Nature Science Foundation of China under Grant number 51875171.

References

Amandeep, Mohammad, F. and Yadav,V., Automatic decision making for multi-criteria load balancing in cloud environment using AHP, Proceedings of International Conference on Computing, Communication and Automation, (2015), pp.569-576. DOI:10.1109/CCAA.2015.7148473.

Bai, D.W., Research on Web Services Semantic Matching and Discovery, Beijing: Beijing University of Posts and Telecommunications, (2007), pp. 43-48. (in Chinese)

Belouadha, F.Z. Omrama, H. and Roudiès, O., Modeling and matching business processes using Web services, International Conference on Multimedia Computing and Systems proceeding (2014), pp. 617-621.

Chen, J., Jiang, Z.H., Concept similarity computation for domain ontology, Computer Engineering and Applications, Vol.42, No.33(2006), pp.163-166.

Feng, W.J., Yin, C., Li, X.B. and Li, L., A classification matching method for manufacturing resource in cloud manufacturing environment, International Journal of Modeling, Simulation, and Scientific Computing, Vol.8, No.2(2017), pp.1750057. DOI: 10.1142/S1793962317920015.

Fei, T., Ying, Z., Li, D.X. and Lin, Z., IoT-Based Intelligent Perception and Access of Manufacturing Resource Toward Cloud Manufacturing, IEEE Transactions on Industrial Informatics, Vol.10, No.2(2014), pp.1547-1557.

Huang, G., Zhou, Z.R. and Zhou, T., Research on the Domain-Ontology-Based Semantic Similarity Computing, Computer Engineering and Science, Vol.29, No.1(2007), pp.112-117.

Lartigau, J., Xu, X.F., Nie, L.S. and Zhan, D.C., Cloud manufacturing service composition based on QoS with geoperspective transportation using an improved Artificial Bee Colony optimization algorithm. International Journal of Production Research, Vol.53, No.14(2015), pp.4380-4404.

Li, B.H., Chai, X.D., Zhang, L. and Lin, Y.Y., Wisdom cloud manufacturing: the wisdom model and means of industrial cloud[J]. China Industry Review, Vol.2, No.2(2016), pp.58-66. (in Chinese)

Lin, D., An information-theoretic definition of similarity, Proc.international Conf.on Machine Learning, Vol.98, No.1998,
Niu, L., Chen, K. and Cheng, Y., Application of improved AHP in Employment Comprehensive Evaluation, Computer Simulation, Vol.28, No.5 (2011), pp.376-379.

Rese, M., Karger, M. and Strotmann, W.C., The dynamics of industrial product service systems (IPS2)-using the net present value approach and real options approach to improve life cycle management, CIRP Journal of Manufacturing Science and Technology, Vol.1, No.4 (2009), pp.279-286. DOI: 10.1016/j.cirpj.2009.05.001

Sheng, B.Y., Zhang, C.L., Yin, X.Y., Lu, Q.B., Cheng, Y., Xiao, T. and Liu, H.M., Common intelligent semantic matching engines of cloud manufacturing service based on OWL-S[J]. International Journal of Advanced Manufacturing Technology, Vol.84, No.1 (2016), pp.103-118.

Wang, Y.X., Qiu, S.H. and Wang, Z.L., A new manufacturing model for servitization-cloud manufacturing overview. Modern Manufacturing Engineering, Vol.36, No.3 (2013), pp.124-128. (in Chinese) DOI: 10.16731/j.cnki.1671-3133.2013.03.023

Wei, S.K., Li, S.J. and Li, Y., Confirming weight of voice of the customer in QFD using the method of AHP. Machinery Design & Manufacture, Vol.42, No.6 (2005), pp.170-172. (in Chinese)

Wu, D.Z., Greer, M.J., Rosen, D.W. and Schaefer, D., Cloud Manufacturing: Strategic vision and state of the art. Journal of Manufacturing Systems, Vol.32, No.4 (2013), pp.564-579 DOI: 10.1016/j.jmsy.2013.04.008.

Wu, Z.J., Gao, Z.B., Cao, Y.L., Ye, X.P. and Yang, J.X., Tolerance design and adjustment of complex customized product based on Cloud Manufacturing. Proceedings of 13th CIRP conference on Computer Aided Tolerancing, Vol.27 (2015), pp.169-175. DOI: org/10.1016/j.procir.2015.04.061

Xie, C., Cai, H.M., Xu, L.D. and Bu, F.L., Linked Semantic Model for Information Resource Service Toward Cloud Manufacturing, IEEE Transactions on Industrial Informatics, Vol.13, No.6 (2017), pp.3338-3349.

Xu, W., Tian, S., Liu, Q., Xie, Y., Zhou, Z. and Pham DT., An improved discrete bees algorithm for correlation-aware service aggregation optimization in cloud manufacturing. International Journal of Advanced Manufacturing Technology, Vol.84, No.1 (2016), pp.17-28.

Yang, S.W., Impact of the upgraded version of German "industrial 4 platform" on China's Manufacturing Industry, Telecommunications Science, Vol.32, No.1 (2016), pp.108-111.

Yang, M.S. and Lin, Z.H., Method to determine importance of customers’ requirements in QFD, Journal of management sciences in China, Vol.6, No.5 (2003), pp.65-71. 6(5):65-71.(in Chinese)

Yang, M.S., Hung, W.L. and Chang-Chien, S.J., On a similarity measure between LR- type fuzzy numbers and its application to database acquisition, International Journal of Intelligent Systems, Vol.20, No.10 (2005), pp.1001-1016.

Ye, L. and Zhang, B., A Method of Web Service Discovery Based on Functional Semantics, Journal of Computer Research and Development, Vol.44, No.8 (2007), pp.1357-1364.

Yin, C., Zhang, Y. and Zhong, T., Optimization model of cloud manufacturing services resource combination for new product development, Computer Integrated Manufacturing System, Vol.18, No.7 (2012), pp.1368-1378.

Yuan, M.H., Deng, K. and Chaovalitwongse, W.A., Manufacturing resource modeling for cloud manufacturing, International Journal of Intelligent System, Vol.32, No.4 (2017), pp.414-436. DOI: 10.1002/int.2.1867.

Zhang, L., Guo, H., Tao, F., Luo, Y.L. and Si, N., Flexible management of resource service composition in cloud manufacturing. Proceedings of International Conference on Industrial Engineering and Engineering, (2010), pp.2278-2282. DOI: 10.1109/IEEM.2010.5674175

Zhang, L., Luo, Y. and Tao, F., Cloud manufacturing: a new manufacturing paradigm, Enterprise Information Systems, Vol.8, No.2 (2014), pp.167-187.

Zhang, X.X., Research on semantic Web service matching algorithm based on similarity of ontology[D]. Chengdu: University of Electronic Science and Technology of China, (2013), pp.38-48. (in Chinese)

Zhou, J., Yao, X., A hybrid artificial bee colony algorithm for optimal selection of QoS-based cloud manufacturing service composition, International Journal of Advanced Manufacturing Technology, Vol.88, No.9 (2016), pp.3371-3387.

[DOI: 10.1299/jamdsm.2019jamdsm0085] © 2019 The Japan Society of Mechanical Engineers