Selection of optimal cloud services based on quality of service ontology

Zhenglan Xie*, Hankun Yin

Chongqing Institute of Engineering, Chongqing 400056, China

zlxie@cqie.edu.cn

ABSTRACT. *The provision of personalized services according to users' actual needs and preferences is a research hotspot in the era of cloud services. In light of the problem, this paper explores the construction of quality of service (QoS) ontology and the optimization of cloud services in the cloud manufacturing environment. Specifically, the QoS attribute features of cloud services were analyzed before setting up the QoS ontology of cloud services. On this basis, an optimal cloud service selection model was established based on QoS ontology, and solved through analytic hierarchy process (AHP). Finally, the validity and applicability of the method were verified by an example. The research results shed new light on the selection of optimal cloud services based on QoS ontology.*

RÉSUMÉ. *La fourniture de services personnalisés en fonction des besoins et des préférences actuelles des utilisateurs est un point chaud de la recherche à l'ère des services Cloud. En fonction de ce problème, cet article explore la construction d'une ontologie de qualité de service (QDS) et l'optimisation des services Cloud dans l'environnement de fabrication Cloud. Plus précisément, les caractéristiques d'attributs de QDS des services Cloud ont été analysées avant la configuration de l'ontologie de QDS des services Cloud. Sur cette base, un modèle de sélection de services Cloud optimaux a été établi en fonction de l'ontologie de QDS et résolu grâce au processus d'analyse hiérarchique (AHP, le sigle de «analytic hierarchy process » en anglais). Enfin, la validité et l'applicabilité de la méthode ont été vérifiées à l'aide d'un exemple. Les résultats de la recherche ont jeté un nouvel éclairage sur la sélection de services Cloud optimaux basés sur l'ontologie de QDS.*

KEYWORDS: *analytic hierarchy process (AHP), cloud services, optimization model, QoS ontology.*

MOTS-CLÉS: *processus d'analyse hiérarchique (AHP), services cloud, modèle d'optimisation, ontologie de QDS.*

DOI:10.3166/ISI.23.6.127-141

1. Introduction

This the development of cloud computing, big data and the Internet of Things (IoT) (Reddy *et al*., 2017), cloud manufacturing (Li *et al*., 2012) has emerged as a web-based service-oriented way of networked manufacturing. The new manufacturing pattern turns wide-area, heterogeneous and distributed resources into physical and virtual services, provides various manufacturing services according to user demand, and promotes the development of manufacturing towards agility, greening, service and intelligence (Yin *et al*., 2012).

The key to effective use and sharing of cloud manufacturing resources lies in the optimal selection of cloud service. Over the years, quite a number of scholars have explored the optimal-selection model for cloud service. For example, scholars (Liu *et al*., 2013; Fang *et al*., 2016; Kurdi *et al*., 20125; Yin *et al*., 2012) establish some models and propose some algorithms for the multi-tasking in cloud manufacturing system. Liu *et al*. (2013) analyses the hierarchal features of manufacturing tasks, and constructs a hierarchal cloud manufacturing service model. Considering user demand, Cai *et al*. (2014) presents a user-specific interactive method for feature weight analysis of service quality. Chen *et al*. (2015) creates virtual service that reflects the horizontal synergy between enterprises, aiming to realize collaboration in cluster supply chain. Liu *et al*. (2016) explore the optimal-selection of cloud services related to the perception of quality of service (QoS).

Most of the above studies are concentrated in the modelling and algorithm of cloud service optimization. Only a few have tackled the issues from the angle of QoS. Even if QoS is taken into account, the attribute analysis is too simple to unlock the description ability of QoS attribute. Hence, the existing approaches are in lack of intelligent optimization of cloud services and unable to satisfy the personalize preferences of cloud service users.

In light of the above, this paper attempts to investigate the construction of QoS ontology and optimal-selection of cloud service in cloud manufacturing. First, the QoS attribute features of cloud services were analysed to build a cloud service QoS ontology; then, a cloud service optimization model was created based on QoS ontology, and the problem was solved with analytic hierarchy process (AHP); finally, the proposed method was proved feasible and applicable through a case study.

Analysis of Cloud Service QoS Attribute Features and Construction of QoS Ontology

1.1. QoS attribute

The QoS attribute describes the non-functional attributes in the cloud service and evaluates the degree of satisfaction to the quality of the specified cloud service. A QoS attribute can include a number of child attributes, each of which corresponds to a specific evaluation criterion (Figure 1).

The following are some of the prominent features of QoS attributes. (1) QoS attributes fall into different classes, and form a complex hierarchy; (2) Different QoS attributes have different tendencies, i.e. a QoS parameter has either positive or negative impact on service quality; (3) QoS weight value reveals the personalized preference of service users in the service selection process, and the weight is directly proportional to the preference; (4) QoS represents the degree of satisfaction with the service, and the satisfaction degree is strong or selective; (5) QoS attributes have different terms of validity.

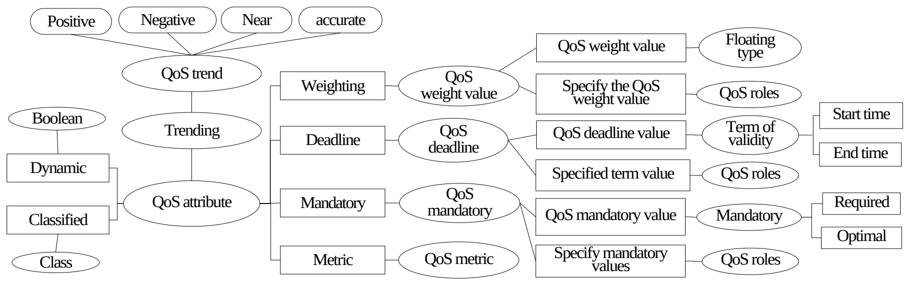

Figure 1. QoS attribute characteristics of cloud services

1.2. QoS association

QoS association refers to the correlation between different QoS parameters in the same service. The parameter is often measured by two dimensions: the direction and the size. Figure 2 illustrates the correlation between different QoS attributes. There is also a special interactive relationship between different QoS attributes. The attributes may be opposite or alike, and the interactive influence has multiple layers. Thus, the QoS association must be considered before releasing service information.

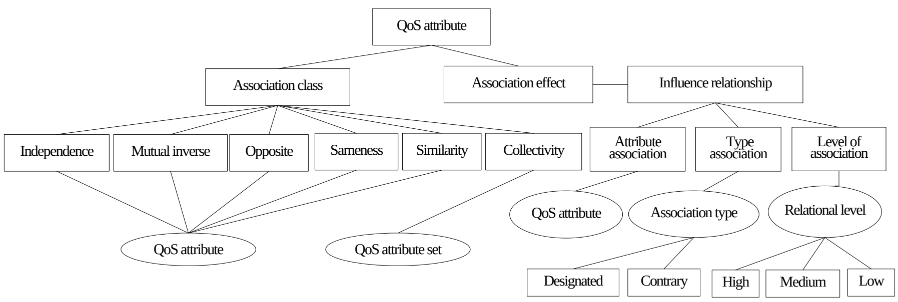

Figure 2. Correlation and description of QoS

1.3. QoS weight

Under the influence of service selection and individual preference, service users are not often consistent in-service selection behaviours. The selection process is based on QoS attribute and attribute group. To rationalize service selection, some service domains and platforms allow users to define personalized weight values for each QoS attribute and attribute group. In addition, the weight value distribution of QoS attribute not only follows the user's personalized preference, but also the role differentiation of the service and the group of QoS attribute.

1.4. QoS measurement and type of parameter value

As shown in Figure 3, the QoS measurement covers the name, value, type and unit of QoS parameters. The following points must be considered during the measurement: (1) both constraint and no-constraint measures should be utilized (Liu *et al*., 2009) to disclose the semantic correlation between parameter names; (2) the parameter value, with a certain periodicity, stands for the quantified value of QoS attribute; (3) the parameter type may be depicted as a single value, multiple values and a fuzzy value; (4) there is a certain semantic correlation between parameter units.

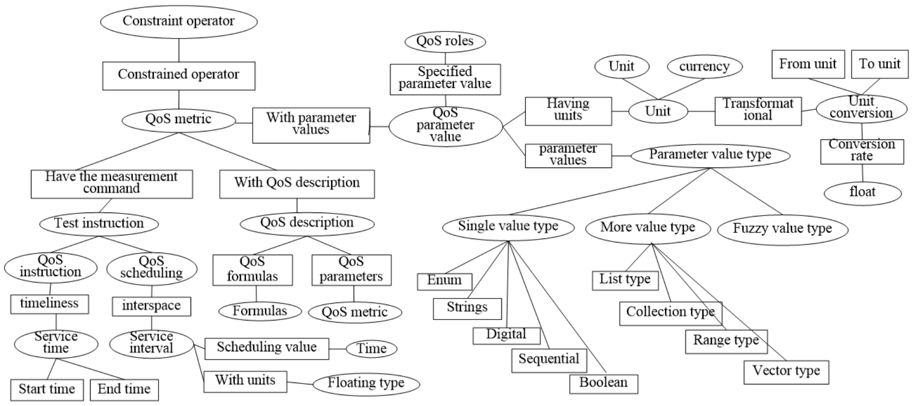

Figure 3. QoS measurement and description of parameter value type

1.5. Construction QoS ontology for cloud service

In view of the above features of QoS features for cloud services, a QoS ontology system (Figure 4) was constructed on core attributes, including service time T, service cost C, service quality Q, reliability R, application A, reputation Re, combination Co and security level Se (Ismail *et al*., 2018; Chakraborty, 2017). The indices can be added or removed by the user according to the actual needs (Mao *et al*., 2015; Paul *et al*., 2017).

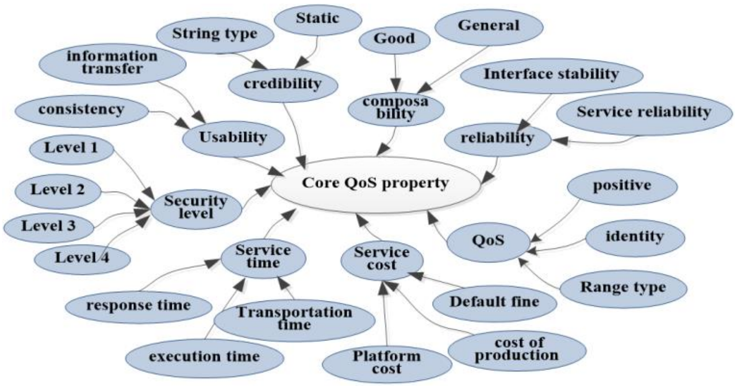

Figure 4. Core QoS attribute ontology

2. Optimal selection of cloud services based on QoS ontology

2.1. QoS attribute group

In cloud services, QoS attribute sets can be divided into different attribute groups by the priority order, so as to facilitate the ranking of services. For example, the QoS attribute group specified by the service user should be prioritized over that specified by the service platform; this means the former attribute group has a higher weight than the latter.

Here, four QoS attributes are defined for the optimization algorithm of cloud service: NCS, NCO, NPS and NPO. Among them, NCS and NPS represent the attributes specified by the service user and service system platform, respectively, while NCO and NPO stand for selectable attributes. Therefore, $NS = NCS \cup NPS$ forms an intrinsic QoS attribute group, and $NO = NCO \cup NPO$ forms an optional attribute group.

The QoS attribute sets were divided into different groups according to the constraints, definitions and optimization algorithms of QoS attribute. First, some specific attribute sets were defined by experts, and the attributes in the intrinsic attribute group were defined or constrained by the general users. Second, some of the QoS attributes satisfied the service quality criterion in the optional attribute group. Third, the weights of QoS attribute groups were determined and distributed, in addition to some QoS attributes in a certain attribute group.

2.2. Comparison rules of QoS values

This section introduces the comparison rules and methods of QoS values to evaluate the relative ranking of two candidate cloud services (Mao *et al*., 2015). The specific method should be determined by the type and tendency of parameters in QoS attribute.

Assume that C1 and C2 are two candidate services in a cloud service Q, TC1 and TC2 be the parameter values of QoS attribute R in C1 and C2, respectively, and MR is the normalized (Xiao and Cai, 2009) weight value of R in the corresponding attribute group.

According to the attribute R, let b_m=*CR 1/CR 2*be the relative rank between C1 and C2, then the relationship is *CR 1/CR 2=1/CR 2/CR 1*. Among the four sets of comparison rules, the first three fit in with any parameter type and tendency of the QoS attribute set, while the fourth only applies to specific situations.

First, if TC1 is uncertain and TC2 is determined, then *CR 1/CR 2*=M_R; in this case, C1 and C2 cannot be compared directly, unless the MR is introduced to the rank comparison. Second, if TC1 is determined but TC2 is uncertain, then *CR 1/CR 2*=1/M_R ; in this case, the comparison should be carried out as it is in case 1. Third, if TC1 and TC2 are uncertain, then *CR 1=CR 2* and *CR 1/CR 2*=1. Fourth, if TC1 and TC2 are determined, then *CR 1≠CR 2*; in this case, the relative rank of C1 and C2 should be compared by different methods based on the parameter type and tendency of R.

The parameter type and trend were analysed as follows:

When the parameter type is Boolean:

$$C_1^R / C_2^R = \begin{cases} M_R & \text{if } T_{C2} \equiv T_R \\ 1/M_R & \text{if } T_{C1} \equiv T_R \end{cases} \quad (1)$$

If the candidate service sets have the same parameter value in the ranking calculation, the QoS attribute Q of the Boolean type will have a relative high ranking in the candidate calculation of cloud service R.

When the parameter type is digital (Goswami and Paul, 2017):

If the QoS parameter values of the C1 and C2 are different from cloud service R, the candidate service nearer to the parameter value has an edge over the other candidate service. Likewise, the candidate service, whose QoS parameter value is consistent with the specified value, ranks higher than its counterpart.

$$C_1^R / C_2^R = \begin{cases} \left|T_{C1} - T_R\right| / \left|T_{C2} - T_R\right| & \text{if } T_{C1} \neq T_R \wedge T_{C2} \neq T_R \\ M_R & \text{if } T_{C2} \equiv T_R \\ 1/M_R & \text{if } T_{C1} \equiv T_R \end{cases} \quad (2)$$

For a positive type QoS attribute: *CR 1/CR 2*=T_{C1}/T_{C2}; in this case, the parameter value of the candidate service is positively correlated with its ranking; For a negative type QoS attribute: *CR 1/CR 2*=T_{C2}/T_{C1}; in this case, the parameter value of the candidate service is negatively correlated with its ranking.

If the parameter value is a string (The function fit (Str1, Str2)), the return value is the fitness $\in(0, 1)$ between Str1 and Str2.

$$C_1^R / C_2^R = \begin{cases} M_R & \text{if } fit(T_{C1}, T_R) \equiv 0 \wedge fit(T_{C2}, T_R) \neq 0 \\ 1/M_R & \text{if } fit(T_{C1}, T_R) \neq 0 \wedge fit(T_{C2}, T_R) \equiv 0 \\ 1 & \text{if } fit(T_{C1}, T_R) \equiv 0 \wedge fit(T_{C2}, T_R) \equiv 0 \\ \dfrac{fit(T_{C1}, T_R)}{fit(T_{C2}, T_R)} & \text{if } fit(T_{C1}, T_R) \neq 0 \wedge fit(T_{C2}, T_R) \neq 0 \end{cases} \tag{3}$$

For string type QoS attributes, the candidate service ranks higher if its parameter value is closer to service R. Suppose the service availability A requested by the service user belongs to level 2 (the level is negatively correlated with the grade and service quality). Then, C1 can provide level 2 usability level, while C2 can provide level 1 usability level. The semantic similarity calculation of ontology domain is calculated by:

$$C_1^R / C_2^R = \frac{fit(\text{"}level\ 2\text{"}, \text{"}level\ 2\text{"})}{fit(\text{"}level\ 1\text{"}, \text{"}level\ 2\text{"})} = \frac{0.2356}{0.1824} = 1.2917 \tag{4}$$

Therefore, it is concluded that C2 ranks higher than C1 in service selection.

If the parameter value is enumerated (the return value of function where (e, S) is the location of element e in the collection or list S):

$$C_1^R / C_2^R = \begin{cases} M_R & \text{if } T_{C2} \equiv T_R \cap T_{C1} \neq T_R \\ 1/M_R & \text{if } T_{C2} \neq T_R \cap T_{C1} \equiv T_R \\ \dfrac{where(T_{C1}, E^R)}{where(T_{C2}, E^R)} & \text{if } E^R \text{ } is\ Positive \\ \dfrac{where(T_{C2}, E^R)}{where(T_{C1}, E^R)} & \text{if } E^R \text{ } is\ Negative \end{cases} \tag{5}$$

For the QoS attribute with enumeration, the ranking is higher if the candidate service contains a positive element. The inverse is also true. For example, suppose security level Se is an enumerated type and there exists (Sen and Sasmita, 2017; Saha and Biswas, 2017)

E^R={Se1.0, Se2.0, Se3.0, Se4.0, Se 5.0}

(the security level is positively correlated with the parameter value). The security level provided by service C1 is Se 4.0, and the security level provided by service C2 is Se 2.0. The values are substituted into (4) to obtain:

$$C_1^R / C_2^R = \frac{where(Se\ 4.0, E^Q)}{where(Se\ 2.0, E^Q)} = \frac{1.95}{1.53} = 1.27 \tag{6}$$

It is also concluded that S1 ranks higher than S2 in service selection.

2.3. Application of the AHP based on QoS

The AHP is a method to decompose elements by objectives, guidelines and plans, laying the basis for qualitative and quantitative analyses. For the optimal-selection of QoS cloud services, the target layer contains the optimization target of cloud services, the criterion layer encompasses the QoS attributes, constraints, and weights, and the plan layer gives the relative ranking calculated by the final parameter values. To identify the optimal cloud service, the author built a hierarchal AHP model, computed the relative weight of QoS attributes, and determined the relative ranking of cloud services by the structure judgement matrix.

2.4. Establishment of the AHP hierarchal model

In cloud services, QoS attributes and candidate cloud service sets form a complex set, which is essential to the multi-layered structure model in AHP. As shown in Figures 5 and 6, the target layer contains the preferred target of candidate cloud services. In the QoS layer, the attribute is divided into the inherent attribute NS and the optional attribute NO.The corresponding weight values are MS and MO (MS+MO=1). The values of MS and MO depend on the ontology and application of certain domain. In general, MS is greater than MO. Meanwhile, the intrinsic attribute NS is split into user-defined attribute set NCS and system-defined attribute set NPS. The corresponding weights are MCS and MPS (MCS+MPS=1). The same is true for selective attributes NO.All in all, there are four sets of attributes in the QoS group, namely NCS, NPS, NCO and NPO. The corresponding weights are MCS, MPS, MCO, and MPO. Unlike the assignment rules for MS and MO, the weight values in the user and system-defined attribute sets can be assigned in a random manner. In the QoS attribute layer, the four attribute sets can be decomposed into a number of child attribute sets. The decomposition process is recursive until non-vector type attributes appear. The relative ranking of final candidate services is recorded in the solution layer.

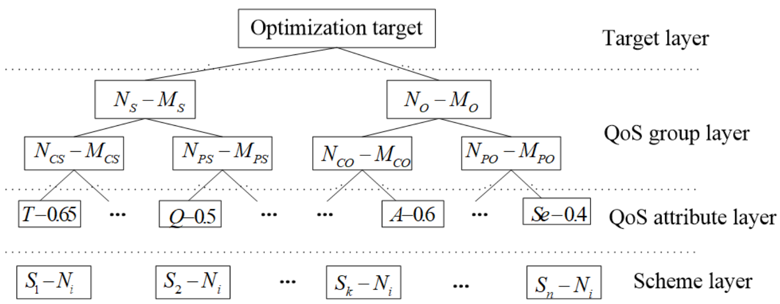

Figure 5. The hierarchical model that distinguishes intrinsic and selectable attributes

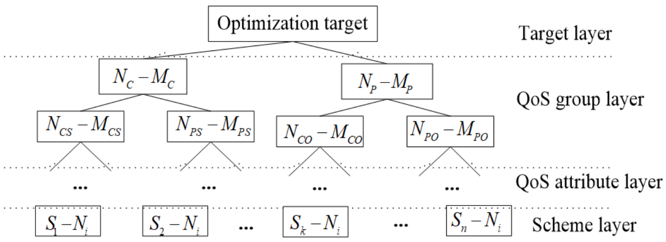

Figure 6. The hierarchical structure model that distinguishes constraint priorities of service users and service system

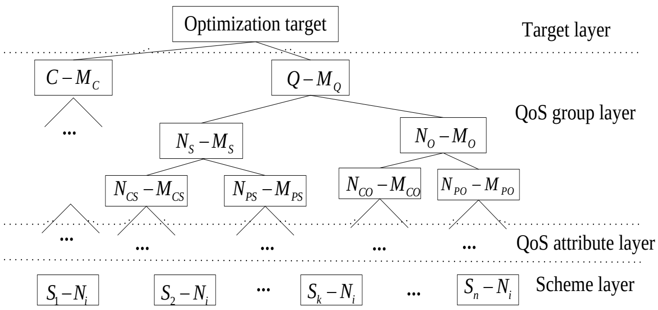

Figure 7. The hierarchical structure model that distinguishes the importance of a particular attribute

To highlight the QoS inherent attributes and optional attributes, the model in Figure 5 should be chosen, and the hierarchal AHP model can be determined based on ontology and specific domain. To highlight the QoS constraint priorities between service users and service system platforms, the model in Figure 7 should be adopted to weigh the importance of service cost C and service quality Q.

2.5. Calculation of the relative weight of QoS attributes

1) Construct the comparison matrix: suppose that there are n candidate service sets in a cloud service, the set of attributes contained in each service is $N_T=\{n_1, n_2, \ldots, n_n\}$, and the corresponding weight set is the row vector $M=\{m_1, m_2, \ldots, m_n\}$; therefore, it is possible to construct an n-order weight matrix: $AM=[m_{ij}]_{n\times n}$.

2) Calculate the row product and evaluate the relative weight: Calculate the row product of every row element in matrix A_M in 1) and open n square: $\overline{m}_1=\sqrt[n]{\prod_{j=1}^{n} m_{ij}}$ For example, a row element in matrix A_M is A=[1 3 5 7] the $\overline{m}_1=\sqrt[4]{1\times3\times5\times7}=3.201$ Therefore, after n times of row operation, all the values of $\overline{m}_1$ can be obtained, putting the normalized relative weight value at $m_i=\overline{m}_i/(\sum \overline{m}_i)$

3) Form a relative weight vector $M_a=[m_i]_{n\times 1}$ where m_i is the normalized weight value in 2).

2.6. Construction of judgment matrix for relative ranking of cloud services

To calculate the relative ranking of cloud services, it is necessary to obtain the eigenvectors corresponding to any of the core QoS attributes in the candidate service. Due to the difference in service demander/provider and the attribute units, the units of different attributes should be normalized before further operations. According to the comparison rule of attribute values in section 3.2, the relative ranking matrix of the service can be constructed for any attribute in NCS, NPS, NCO and NPO attribute sets. For property R, the relative ranking matrix for its service is:

$$AN_R = \begin{bmatrix} 1 & \cdots & v_k/v_1 & \cdots & v_m/v_1 \\ \vdots & \ddots & \vdots & \vdots & \vdots \\ v_1/v_k & \cdots & 1 & \cdots & v_m/v_k \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ v_1/v_m & \cdots & v_k/v_m & \cdots & 1 \end{bmatrix}_{m\times m} \tag{7}$$

Where m is the number of candidate service sets; v_k is the parameter value of attribute R in each attribute set. Based on matrix AN_R, the relative position ranking vector BN_R of attribute R can be obtained by:

$$AN_R = [an_r]_{m*m} \xrightarrow{Normalization} BN_R = [bn_r]_{m\times l} \tag{8}$$

Through the relative ranking matrix AN_R. The detailed procedure is described in the application validation section.

In NCS, NPS, NCO and NPO, any attribute set may contain relative ranking vectors of several attributes, accompanied by different weights. The relative ranking vectors in each attribute set should be further processed to yield the final ranking of cloud services. Taking the user-defined inherent attribute set for example, if there are n attributes, the relative ranking matrix of the attribute set NCS is

$$AN_{CS} = [an_{CS}]_{m\times n} \tag{9}$$

where $[an_{cs}]_{m\times n}$ is composed of n relative rank vectors $[an_{cs}]_{m\times n}$ $[an_{cs}]_{m\times n}$. Then, the relative ranking vector of the attribute set NCS can be obtained by:

$$BN_{CS} = AN_{CS} \times M_q = [an_{cs}]_{m\times n} \times [m_q]_{n\times 1} = [bn_{cs}]_{m\times 1} \tag{10}$$

Where M_q is the relative weight vector. Similarly, the relative ranking vectors BN_{PS}, BN_{co} and BN_{PO} can be acquired for the remaining three attribute sets NPS, NCO and NPO. In the cloud service, the attribute is generally divided into the inherent attribute NS and the optional attribute NO, and NS = NCS∪NPS, NO = NCO∪NPO. Thus, for the inherent attribute set NS, the relative ranking vector is:

$$BN_s = BN_{CS} \times m_{cs} + BN_{PS} \times m_{ps} = [bn_{cs}]_{m\times 1} \times m_{cs} + [bn_{ps}]_{m\times 1} \times m_{ps} = [bn_s]_{m\times 1} \tag{11}$$

For the optional attribute set, the relative ranking vector is:

$$BN_O = BN_{CO} \times m_{co} + BN_{PO} \times m_{po} = [bn_{co}]_{m\times 1} \times m_{co} + [bn_{po}]_{m\times 1} \times m_{po} = [bn_o]_{m\times 1} \tag{12}$$

For the full QoS attribute set NA, there is NA = NS∪NO.Hence, the relative ranking vectors of the intrinsic attribute and the optional attribute should be multiplied by the corresponding weights. The final ranking vector can be calculated as follows.

$$BN_A = BN_S \times m_s + BN_O \times m_o = [bn_s]_{m\times 1} \times m_s + [bn_s]_{m\times 1} \times m_o = [bn_a]_{m\times 1} \tag{13}$$

3. Application verification

This chapter is a case study designed to validate the feasibility of the research results above. The case contains five candidate cloud service sets, and the system involves eight core indies, namely, service cost C, service time T, quality of service Q, reliability R, availability A, reputation Re, combination Co and security level Se. As shown in Table 1, the QoS attribute group and its weight is described, together with QoS tendency and QoS parameter type. which are shown in Table 1.

Table 1. Cloud services based on QoS

Property groups and weights	NS, MS=0.55 NO, MO=0.45		NCS, MCS=0.6 NPS, MPS=0.4	NCO, MCO=0.6	NPO, MPO=0.4			
QoS attributes	T	C	Q	R	A	Re	Co	Se
Attribute weights	0.65	0.35	0.5	0.5	0.6	0.4	0.4	0.6
unit	Month	Ten thousand	%	nothing	nothing	nothing	nothing	nothing
Parameter value type	Numerical	Numerical	Range	Strings	Strings	Strings	Strings	enum
trend	Positive	Negative	Positive	Accurate	Positive	Positive	Positive	Positive
Specified parameter value	1	2	>95	high	Level 2	high	high	Se 3.0

S1	2	2	>96	high	Level 2	high	higher	Se 3.0
S2	1.2	2.4	>90	general	Level 3	higher	higher	Se 2.0
S3	1.1	1.9	>91	higher	Level 3	higher	high	Se 3.0
S4	1.0	2.5	>95	high	Level 2	high	high	Se 3.0
S5	0.8	2.2	>98	high	Level 1	high	high	Se 4.0

In general, T and C are the specific attributes NCS of the service user, Q and R are the selectable attributes NCO of the service user, A and Re are the specific attributes NPS of the service system platform, Co and Se are the optional attributes NPO of the service system platform.

$$AN_C = \begin{array}{c|ccccc} & S_1 & S_2 & S_3 & S_4 & S_5 \\ \hline S_1 & 1 & 2.4/2 & 1.9/2 & 2.5/2 & 2.2/2 \\ S_2 & 2/2.4 & 1 & 1.9/2.4 & 2.5/2.4 & 2.2/2.4 \\ S_3 & 2/1.9 & 2.4/1.9 & 1 & 2.5/1.9 & 2.2/1.9 \\ S_4 & 2/2.5 & 2.4/2.5 & 1.9/2.5 & 1 & 2.2/2.5 \\ S_5 & 2/2.2 & 2.4/2.2 & 1.9/2.2 & 2.5/2.2 & 1 \end{array} \tag{14}$$

First, the relative ranking of the service was calculated as follows. For attribute C, the relative ranking matrix of its service is:

The column vector is normalized to obtain the eigenvector. The relative ranking vector of attribute C is:

$$BN_C = \begin{bmatrix} 0.2013 & 0.1874 & 0.2205 & 0.1860 & 0.1937 \end{bmatrix}^T \tag{15}$$

Similarly, the relative ranking vector of attribute P is:

$$BN_P = \begin{bmatrix} 0.1903 & 0.2068 & 0.1717 & 0.2008 & 0.2213 \end{bmatrix}^T \tag{16}$$

Combining the first-order matrices in (19) and (20), the relative ranking matrix for the attribute group NCS can be obtained as:

$$AN_{CS} = \begin{bmatrix} 0.2013 & 0.1903 \\ 0.1874 & 0.2068 \\ 0.2205 & 0.1717 \\ 0.1860 & 0.2008 \\ 0.1937 & 0.2213 \end{bmatrix} \tag{17}$$

According to the relative weights in Table 1, the relative ranking vector of attribute group NCS can be obtained as:

$$BN_{CS} = AN_{CS} \times M_{CS} = \begin{bmatrix} 0.2013 & 0.1903 \\ 0.1874 & 0.2068 \\ 0.2205 & 0.1717 \\ 0.1860 & 0.2008 \\ 0.1937 & 0.2213 \end{bmatrix} \times \begin{bmatrix} 0.65 \\ 0.35 \end{bmatrix} = \begin{bmatrix} 0.1974 \\ 0.1942 \\ 0.2034 \\ 0.1912 \\ 0.2034 \end{bmatrix} \tag{18}$$

Similarly, the relative ranking vectors of attribute groups NCO, NPS and NPO are respectively calculated by:

$$BN_{CO} = AN_{CO} \times M_{CO} = \begin{bmatrix} 0.1629 & 0.2085 \\ 0.1989 & 0.2009 \\ 0.2035 & 0.1614 \\ 0.2062 & 0.2004 \\ 0.2154 & 0.2196 \end{bmatrix} \times \begin{bmatrix} 0.4 \\ 0.6 \end{bmatrix} = \begin{bmatrix} 0.1903 \\ 0.2001 \\ 0.1782 \\ 0.2072 \\ 0.2179 \end{bmatrix} \tag{19}$$

Second, the NCS and NPS were multiplied by their corresponding weight coefficients MCS and MPS, respectively. Then, the results were added to obtain the relative ranking vector of the service.

$$BN_S = BN_{CS} * m_{cs} + BN_{PS} * m_{ps} = \begin{bmatrix} 0.1974 \\ 0.1942 \\ 0.2034 \\ 0.1912 \\ 0.2034 \end{bmatrix} * 0.6 + \begin{bmatrix} 0.2019 \\ 0.1976 \\ 0.1897 \\ 0.2023 \\ 0.2024 \end{bmatrix} * 0.4 = \begin{bmatrix} 0.1992 \\ 0.1956 \\ 0.1979 \\ 0.1956 \\ 0.2030 \end{bmatrix} \tag{20}$$

Similarly, the relative ranking vector of NO is obtained as:

$$BN_O = BN_{CO} * m_{co} + BN_{PO} * m_{po} = \begin{bmatrix} 0.1903 \\ 0.2001 \\ 0.1782 \\ 0.2072 \\ 0.2179 \end{bmatrix} * 0.6 + \begin{bmatrix} 0.1941 \\ 0.2155 \\ 0.1975 \\ 0.1946 \\ 0.2068 \end{bmatrix} * 0.4 = \begin{bmatrix} 0.1918 \\ 0.2063 \\ 0.1859 \\ 0.2022 \\ 0.2135 \end{bmatrix} \tag{21}$$

Third, the final ranking vector of candidate cloud service is obtained as:

$$BN_A = BN_S * m_s + BN_O * m_O = \begin{bmatrix} 0.1992 \\ 0.1956 \\ 0.1979 \\ 0.1956 \\ 0.2030 \end{bmatrix} * 0.55 + \begin{bmatrix} 0.1918 \\ 0.2063 \\ 0.1859 \\ 0.2022 \\ 0.2135 \end{bmatrix} * 0.45 = \begin{bmatrix} 0.1959 \\ 0.1912 \\ 0.1925 \\ 0.1986 \\ 0.2073 \end{bmatrix} \tag{22}$$

According to the final ranking vector FSRRV of the candidate cloud service, the final optimal-selection ranking of the cloud service is: $S_5>S_4> S_1> S_3>S_2$.

4. Conclusions

This paper investigates the construction of QoS ontology and optimal-selection of cloud service in cloud manufacturing. Specifically, the QoS attribute features of cloud services were analysed to build a cloud service QoS ontology; then, a cloud service optimization model was created based on QoS ontology, and the problem was solved with analytic hierarchy process (AHP); finally, the proposed method was proved feasible and applicable through a case study. The results shed new light on the optimal-selection of cloud service based on QoS ontology.

Acknowledgement

This paper is supported by Fund project: humanities and social sciences research project of chongqing education committee (16SKGH221); Chongqing "three special action plans" characteristic specialty construction project ([2016] 50).

References

Cai T., Liu W. N., Liu B. (2014). A new method of cloud manufacturing service optimal-selection based on intuitionistic fuzzy set. *China Mechanical Engineering*, Vol. 3, pp. 352-356, 421. https://doi.org/10.3969/j.issn.1004-132X.2014.03.014

Chakraborty S. (2017). Computer cyber security analysis as well as results. *Review of Computer Engineering Studies*, Vol. 4, No. 1, pp. 36-40. https://doi.org/10.18280/rces.040107

Chen F. Z., Dou R. L., Li M. Q. (2015). A flexible QoS-aware eweb service composition method by multi-objective optimization in cloud manufacturing Original. *Computers & Industrial Engineering* Vol. 12, pp. 11-18.

Fang D. R., Liu X. D., Romdhani I., Jamshidi P., Pahl C. (2016). An agility-oriented and fuzziness-embedded semantic model for collaborative cloud service search, retrieval and recommendation. *Future Generation Computer Systems*, Vol. 3, pp. 11-26. https://doi.org/10.1016/j.future.2015.09.025

Goswami J., Paul M. (2017). Symmetric key cryptography using digital circuit based on one right shift. *Review of Computer Engineering Studies*, Vol. 4, No. 2, pp. 57-61. https://doi.org/10.18280/rces.040203

Gu J. N. (2011). Research for personalized Web service selection mechanism based on multi-layer QoS ontology. *Chongqing University*.

Ismail A., Saad M., Abbas R. (2018). Cyber security in internet of things. *Review of Computer Engineering Studies*, Vol. 5, No. 1, pp. 17-22. https://doi.org/10.18280/rces.050104

Kurdi H., Al-Anazi A., Campbell C., Faries A. A. (2015). A combinatorial optimization algorithm for multiple cloud service composition. *Computers & Electrical Engineering*, Vol. 2, pp. 107-113. https://doi.org/10.1016/j.compeleceng.2014.11.002

Li B. H., Zhang L., Ren L., Chai X., Tao F., Wang Y., Yin C., Huang P., Zhao X., Zhou Z. (2012). Typical characteristics, technologies and applications of cloud manufacturing. *Computer Integrated Manufacturing Systems*, Vol. 18, No.7, pp. 1345-1356.

Liu G. Q., Zhu Z. L., Liu Y. (2009). A method for QoS measurement of web service based on service using information. *Journal of Northeastern University*, Vol. 10, pp. 1398-1401. https://doi.org/10.1360/972009-1549

Liu W. N., Liu B., Sun L. H. (2013). Multi-Task oriented service composition in cloud manufacturing. *Computer Integrated Manufacturing Systems*, Vol. 1, pp. 199-209. https://doi.org/10.13196/j.cims.2013.01.201.liuwn.021

Liu W. N., Ma G., Liu B. (2013). Study on hierarchical service composition in cloud manufacturing. *China Mechanical Engineering* Vol. 10, pp. 1349-1356, 2013. https://doi.org/10.3969/j.issn.1004-132X.2013.10.015

Liu Z. Z., Chu D. H., Song C. (2016). Social learning optimization (SLO) algorithm paradigm and its application in QoS-aware cloud service composition. *Information Sciences*, Vol. 1, pp. 315-333. https://doi.org/10.1016/j.ins.2015.08.004

Mao C. Y., Chen J. F., Towey D., Chen J., Xie X. (2015). Search-based QoS ranking prediction for web services in cloud environments. *Future Generation Computer Systems*, Vol. 9, pp. 111-126. https://doi.org/10.13196/j.cims201410027

Paul S., Dasgupta P., Kr N. P., Chaudhuri A. (2017). Secured image encryption scheme based on DNA encoding and chaotic map. *Review of Computer Engineering Studies*, Vol. 4, No. 2, pp. 70-75. https://doi.org/10.18280/rces.040206

Reddy V. S., Rao T. V., Govardhan A. (2017). Data mining techniques for data streams mining. *Review of Computer Engineering Studies*, Vol. 4, No. 1, pp. 31-35. https://doi.org/10.18280/rces.040106

Saha S., Biswas K. (2017). A comparative study of Fiber Bragg Grating based tilt sensors, *Review of Computer Engineering Studies*, Vol. 4, No. 1, pp. 41-46. https://doi.org/10.18280/rces.040108

Sen M., Sasmita S. C. (2017). Security and privacy issues for cloud computing and its challenges. *Review of Computer Engineering Studies*, Vol. 4, No. 2, pp. 62-66. https://doi.org/10.18280/rces.040204

Xiao H. G., Cai C. Z. (2009). Comparison study of normalization of feature vector. *Computer Engineering and Applications*, Vol. 22, pp. 117-119. https://doi.org/10.3778/j.issn.1002-8331.2009.22.038

Xue X., Liu Z. Z., Huang B. Q. (2014). Enterprise service composition method for cluster supply chain. *Computer Integrated Manufacturing Systems* Vol. 10, pp. 2599-2608.

Yin C., Xia Z., Li W. (2012). Semantic matching technique of cloud mabufacturing service based on OWL-S. *Computer Integrated Manufacturing Systems*, Vol. 7, pp. 1494-1502.

Yin C., Zhang Y., Zhong T. (2012). Optimization model of cloud manufacturing services resource combination for new product. *Computer Integrated Manufacturing Systems*, Vol. 7, pp. 1368-1378.