QoS Correlation-based Service Composition Algorithm for Multi-constraint Optimal Path

Jian Yu (181380559@qq.com)
Sanming University https://orcid.org/0000-0003-3865-2664
Zhixing Lin
Sanming University
Qiong Yu
Sanming No 2 High School
Xiangmei Xiao
Sanming University

Research Article

Keywords: Multi-constraint, Optimal path,QoS,Correlation, Service composition

Posted Date: December 1st, 2021

DOI: https://doi.org/10.21203/rs.3.rs-1010772/v1

QoS Correlation-based Service Composition Algorithm for Multi-constraint Optimal Path

Jian Yu[1,2*], Zhixing Lin[1,2], Qiong Yu[3] and Xiangmei Xiao[1,2]

[1*]Network Center of Sanming University, Sanming Fujian, 365004, China.

[2]School of Information Engineering, Sanming University, Sanming 365004, China.

[3]Sanming No.2 High School, Sanming Fujian, 365000, China.

*Corresponding author(s). E-mail(s): 20020127@fjsmu.edu.cn;

Abstract

With the development of network service integration, in order to obtain a better quality of service (QoS) guarantee, In this Paper,We consider the characteristics of integrated network service composition and correlation, this paper proposes a approximate algorithm based on multi-constraint optimal path selection (MCOPS). We analyses the QoS correlation criteria, correlation ratios, and Skyline algorithms to calculate the optimal path by dynamic programming, record the path nodes, and obtain the optimal service composition path that meets the user's demand. Simulation results demonstrate the good performance of the proposed algorithm in both the average calculation time and the solution path quality.

Keywords: Multi-constraint, Optimal path,QoS,Correlation, Service composition

1 Introduction

With the rapid development of cloud services [1], heterogeneous networks [2], 5G networks [3], mobile payments [4], and other network services, a certain correlation can be observed between various services [5]. Cloud computing [6] employs the Internet as a carrier to provide users fast, reliable, and quality of service (QoS) guaranteed data processing services. In this complex network environment, everything is for service. Users with a mobile phone connected to the Internet can choose different services and forms anytime and anywhere. Different network services have a certain correlation in real-life scenarios, making a service correlation performance between network QoS assurance to obtain a superior service composition solution. How to deal with this correlation has become a hot research topic.

In real life, people maximize their interests based on the relevance of things. For example, when a user wants to travel, he should formulate the most suitable itinerary by correlating between various services in a certain APP on the mobile phone. For each service class (also called task or abstract service, such as ticket reservation, car-hailing service, hotel accommodation, and catering reservation), there are many candidate services (also called service instances or specific services) with different qualities to perform the functions of each service class. Therefore, people usually choose the best service composition plan based on the candidate service quality to improve the quality of experience.

Different service providers cooperate and launch a series of preferential services to attract more customers and improve service quality. For example, an APP service provider establishes a cooperative relationship with an offline entity, and passengers can enjoy the accommodation discounts while staying in their partner hotel. At this time, customers can select relevant candidate services according to their needs to obtain a superior service composition plan and improve the quality of experience. When a user selects a certain network service, its service quality often changes depending on the network service quality; that is, the choice of one candidate service can affect the service quality of the other one. This means a correlation, also known as dependency [7], between different services.

In terms of network service composition correlation, it has also become the focus of many scholars' research. Zhen and Xiang [8] proposed a composition method of association rules to overcome the overload problem of the high-speed rail information service function. The contextual information was added to the service composition, and a set of services that meet the current application scenario was recommended to the user according to the correlation rules to improve the service's accuracy and hit rate. Barakat et al. [9] considered the QoS correlation between Web services. They adopted the technology of eliminating correlation to propose a correlation-aware service selection method. Wagner et al. [10] modeled the dependency relationship between service quality to design a multi-objective service composition algorithm to meet user requirements. In the service social network environment, according to the service coordination network characteristics, Lei and Ming-lun [11] proposed an improved Skyline manufacturing service composition optimization method based on QoS collaboration-association and investigated the influence degree of related resources on QoS. The QoS collaboration association improved the search efficiency and results of the Skyline algorithm. Yuanlei et al. [12] proposed a genetic clustering-based reliable Web service composition optimization method. The initial combined service set was screened once using a confidence table to improve the reliability of the composition. In order to improve the degree of combinational optimization, the genetic algorithm was then utilized to optimize its service set for secondary clustering.

Ergun et al. [13] proposed an approximate combination algorithm (ADAPT) for the dual-measurement QoS routing problem. By selecting the optimal upper and lower bounds, a polynomial-time approximation test process was introduced to continuously reduce the distance between these two bounds, while the path search problem was transformed into an interval search problem. The range shrinking was employed to find the most optimal solution quickly. Xue et al. [14] proposed the PeseudoMCPP algorithm, in which a path search space was constructed using the number of path hops. When the first dimension path weight constraint D was satisfied, the other weights were satisfied constraint C. The research results mentioned above were mainly for Web service composition, only focusing on obtaining the maximum effective value of the service composition, and did not consider the correlation between cloud services and network service quality values in the cloud computing environment. However, both cloud service quality and network service performance can affect user experience and affect the service quality level of the cloud computing platform. Therefore, by comprehensively considering the network service characteristics and combining the ideas of the above two algorithms, this paper provided the MCOPS algorithm to solve the problem and help users obtain a superior service composition plan.

2 Related work

2.1 QoS correlation-based composition model

Based on [15], this paper employs auxiliary graphs to establish the additional search space (extra space brought by correlation) to study the correlation. Therefore, the key point is to construct an effective auxiliary graph to reflect the correlation between services.

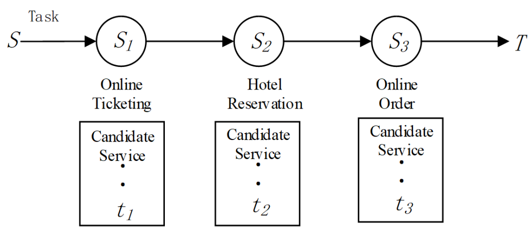


Fig. 1: An example of a travel network service correlation plan

Definition 1 QoS correlation is a certain connection between the QoS of two candidate services. For example, the QoS of a candidate service can change according to the changes in the QoS of other candidate services. Assuming that the three tasks of air ticket reservation, hotel reservation, and online meal ordering in the travel plan shown in Fig. 1 are network services, the corresponding network delay can be adjusted as its correlation service, as shown in Table 1.

Table 1: Two examples of correlation criteria

Task	Candidate service	Service cost (yuan)	Network delay(ms)
T_A	a_1	350	3
	a_2	420	2.8
	a_3	480	2.6
T_B	b_1	280	2.3
	b_2	260	2.4
	b_3	240	2.6
T_C	c_1	60	2.8
	c_2	70	2.5
	c_3	80	2.2

This paper employs the method proposed by Yu et al. [16] to transform the travel plan flow chart into a candidate service network topology diagram according to the following rules to model the additional search space caused by the QoS correlation:

(1) In the abstract flow chart, if there is a directed edge from the abstract service S_A (service class) to another abstract service S_B, then each specific service (candidate service or service instance) in the abstract service S_A has an edge pointing to each specific service in the abstract service S_B;

(2) The cost of each specific service is set on the corresponding service node;

(3) A source node s and a target node t are added, where the node s connects all specific service nodes in the first abstract service, while the network delay value of these edges is set to 0, and the node t connects all specific service nodes in the last abstract service.

(4) The network delay value of each specific service is moved to its outgoing edge.

After the above four composition steps, a new directed acyclic service network graph can be obtained, in which each node has a network service cost value and each edge has a network delay value. As shown in Fig. 2, the QCSC problem can be modeled as an MCOPS problem. Finding a path from s to t in the directed graph that meets user requirements is a feasible service composition plan.

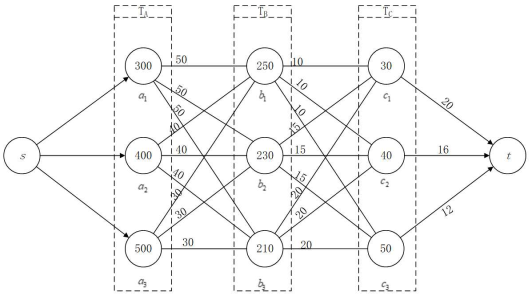


Fig. 2: Service correlation composition model

2.2 QoS correlation criteria

Definition 2 (Correlation Criterion) When the QoS of a candidate service in a certain service class depends on the corresponding one in another service class, it is said that there is a correlation criterion between the two candidate services. Obviously, in a correlation criterion, if the QoS of one of the candidate services changes, the QoS of the related candidate service must also change. When multiple candidate services are involved in a correlation criterion, the change of the QoS of these candidate services will inevitably change the QoS of some other candidate services. From the mathematical definition, in the correlation criterion $z_i = \left\{(\alpha, \beta) \mid \alpha \in V, \beta \in E'\right\}$, E' represents the new weight set of the correlation node, $\alpha \in \left\{v'_1, v'_2, \cdots, v'_I\right\}$ represents I correlation nodes, and $\beta \in \left\{\omega'_1(e), \omega'_2(e), \cdots, \omega'_I(e)\right\}$ represents I new weights.

The mathematical definitions for the process of establishing an association matcher and their corresponding symbols required in this process [17 19] are given as follows:

SoS: Set of Service;

FoS: Function of Service;

PI PO: Property Input Property Output;

$\overset{t}{\rightarrow}$: The order relationship between services within the effective time t of the execution path-transitivity, irreversibility;

$\updownarrow_t$: The branch relationship between services on the execution path within the service's effective time;

$\overset{t}{\leftrightarrow}$: The parallel relationship of the service on the execution path (considering that the services are independent) within the service's effective time t;

$\overset{t}{\leftarrow}$: The service can be repeated within the effective time t, while the service itself has reflexive characteristics;

$\overset{t}{\perp}$: The interrupt call relationship between services within the execution path's effective time t;

$\prod t$: The semantic fuzzy matching relationship between services within the execution path's effective time t.

CR1: The sequential structure correlation; if there is a sequential order relationship between two atomic services (or component services) in the execution path of the composition business, this relationship belongs to the correlation relationship of the structure of the mapping service.

$$\left\{(s_i, s_j) \subset \mathrm{CR}\,1 \mid \left(\forall s_i \in [\mathrm{SoS}_i, \mathrm{FoS_i}], \forall s_j \in [\mathrm{SoS}_j, \mathrm{FoS}_j]\right) \wedge \exists \left(s_i \xrightarrow{t} s_j\right)\right\} \quad (1)$$

CR2: The branch structure correlation; if there is a judgment relationship on the same node between two atomic services (or component services) in the execution path of the composition service, this relationship belongs to the branch structure correlation.

$$\begin{aligned} &\{(s_i, s_j) \subset \mathrm{CR1} \mid (\forall s_i \in [\mathrm{SoS}_i, \mathrm{FoS}_i] \\ &\forall s_j \in [\mathrm{SoS}_j, \mathrm{FoS}_j]) \wedge \exists\left(\updownarrow_t (s_i, s_j)\right)\} \end{aligned} \quad (2)$$

CR3: Parallel structure correlation; if two atomic services (or component services) have independent parallel relationships on a certain path in the execution path of the composition service, this structure is called a parallel structure correlation.

$$\left\{(s_i, s_j) \subset \mathrm{CR}\,1 \mid \left(\forall s_i \in [\mathrm{SoS}_i, \mathrm{FoS}_i], \forall s_j \in [\mathrm{sos}_j, \mathrm{FoS}_j]\right) \wedge \exists \left(s_i \overset{t}{\leftrightarrow} s_j\right)\right\} \quad (3)$$

CR4: Cyclic structure correlation; if two atomic services (or component services) are executed multiple times according to functional requirements at a certain time in the execution path of the composition service, this operation is called a cyclic structure correlation.

$$\left\{s_i \subset \ \mathrm{C\ R\ 4} \mid \left(\forall s_i \in [\mathrm{S} \circ \mathrm{S}_i,\ \mathrm{F} \circ \mathrm{S_i}]\right) \wedge \exists \left(s_i \overset{t}{\leftarrow}\right)\right\} \quad (4)$$

CR5: Interrupt call structure correlation; if the atomic service (or component service) interrupts and calls other services according to functional requirements at a certain time in the execution path of the composition service, this operation is called an interrupt call structure correlation.

$$\left\{(s_i, s_j) \subset \mathrm{CR5} \mid \left(\forall s_i \in [\mathrm{SoS}_i, \mathrm{FoS}_i], \forall s_j \in [\mathrm{SoS}_j, \mathrm{FoS_j}]\right) \wedge \exists \left(s_i \overset{t}{\perp} s_j\right)\right\} \quad (5)$$

CR6: Equivalent or exact matching correlation; if the PI of the latter service is equivalent to or a subset of the PO of the previous service in the service composition process, it is called an equivalent or exact matching correlation.

$$\{(s_i, s_j) \subset \mathrm{CR6} \mid \left(\forall s_i \in [\mathrm{SoS}_i, \mathrm{PO}_i], \forall s_j \in [\mathrm{SoS}_j, \mathrm{PI}_j]\right) \wedge \exists \left(\mathrm{PI}_j \approx \mathrm{PO}_i \| \mathrm{PI}_j \subseteq \mathrm{PO}_i\right)\} \quad (6)$$

CR7: Partial matching correlation; if the PO of the former part is a subset of the latter service parameters in the service composition process, it is called a partial matching correlation.

$$\left\{(s_i, s_j) \subset \mathrm{CR7} \mid \left(\forall s_i \in \left[\mathrm{SoS}_i, \mathrm{PO}_i\right], \forall s_j \in \left[\mathrm{S} \circ \mathrm{S}_j, \mathrm{PI}_j\right]\right) \wedge \exists \left(\mathrm{PO}_i \subseteq \mathrm{PI}_j\right)\right\} \tag{7}$$

CR8: Cross-matching correlation; if the intersection of the PIs of the latter service and the POs of the previous service is non-empty in the service composition process, it is called a cross-matching correlation.

$$\left\{(s_i, s_j) \subset \mathrm{CR}\,8 \mid \left(\forall s_i \in \left[\mathrm{SoS}_i, \mathrm{PO}_i\right], \forall s_j \in \left[\mathrm{SoS}_j, \mathrm{PI}_j\right]\right) \wedge \exists \left(\mathrm{PO}_i \cap \mathrm{PI}_j \neq \phi\right) \cap \left(\mathrm{PO}_i \notin \mathrm{PI}_j\right) \tag{8}$$

CR9: Fuzzy matching correlation; the service interface parameters are described according to description languages, such as BPEL and UDDI, in the service composition process . A fuzzy relationship is formed through the fuzzy matching strategy of service interface parameters, called the fuzzy matching correlation.

$$\left\{(s_i, s_j) \subset \mathrm{CR9} \mid \left(\forall s_i \in \left[\mathrm{SoS}_i, \mathrm{PI_i}\right], \forall s_j \in \left[\mathrm{S} \circ \mathrm{S}_j, \mathrm{PO}_j\right]\right) \wedge \exists \left(\mathrm{OP}_j \Pi_t \mathrm{PI}_i\right)\right\} \tag{9}$$

CR10: Competitive relationship correlation; if a competitive relationship is formed between services in the same business field or with the same function in the business implementation process of composition services, this relationship is called a competitive relationship correlation.

$$\left\{(s_i, s_j) \subset \mathrm{CR10} \mid \left(\exists s_i, s_j \in \left[\mathrm{soS}_i, \mathrm{Fos}_i\right]\right) \wedge \forall \left(\left(s^\wedge s_j\right) \subset \mathrm{CS}\right)\right\} \tag{10}$$

CR11: Collaboration relationship correlation; in the business implementation process of composition services, in the same business process, one service has a guiding and assisting role for another one to produce a correlation relationship, called the collaboration relationship correlation.

$$\left\{(s_i, s_j) \subset \mathrm{CR11} \mid \left(\forall s_i \in \left[\mathrm{SoS}_i, \mathrm{FoS}_i\right], \forall s_j \in \left[\mathrm{SoS}_j, \mathrm{FoS}_j\right]\right) \wedge \left(\left(s_i \&\& s_j\right) \subset \mathrm{CS}\right)\right\} \tag{11}$$

CR12: Time window correlation; business application requirements generate an interactive relationship between services in a specific period in the process of service implementation and operation, called a time window correlation.

$$\left\{(s_i, s_j) \subset \mathrm{CR12} \mid \left(\forall s_i \in \left[\mathrm{SoS}_i, \mathrm{FoS}_i\right], \forall s_j \in \left[\mathrm{SoS}_j, \mathrm{FoS}_j\right]\right) \wedge \exists \left(T_t \left(s_i \&\& s_j\right) \subset \mathrm{CS}\right)\right\} \tag{12}$$

CR13: Alliance relationship correlation; an alliance relationship is formed in formulating the execution path of the composition service due to the business operation, environmental requirements, and commercial interests, called alliance relationship correlation. For example, if the services of Internet companies A and B are optional services within the scope of a business alliance, they tend to recommend each other to users, which will eventually affect the selection result of the composition service. Service bundling and service recommendation are the external manifestations of the alliance relationship. Since the two may not participate in the service composition process simultaneously, their specific equations (set relations) are not listed here. This rule will analyze them from the fields of semantics and fuzzy matching.

2.3 QoS correlation ratio

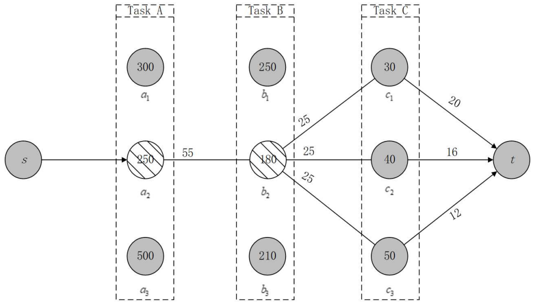
Definition 3 (Correlation Ratio) In a service network with a known scale, the correlation service ratio is defined as the ratio of the number of correlation candidate services to the number of all candidate services.

When there is a correlation between candidate services in different tasks (or service classes), users may obtain a superior quality service composition plan by utilizing their correlation. According to the data presented in Table 1, Table 2 gives an example of two correlation criteria for the four candidate services belonging to the four tasks. As shown in Fig. 2, the least costly service composition path (SCP) is $p_1 = s \rightarrow a_1 \rightarrow b_3 \rightarrow c_1 \rightarrow t$ when the correlation between services is not considered. However, when considering the correlation criterion 1 in Table 2, a better SCP p_2 can be obtained, i.e., $p_2 = s \rightarrow a_2 \rightarrow b_2 \rightarrow c_1 \rightarrow t$. This is because the cost of p_2 is less than that of p_1; that is, 460 $<$ 540. Furthermore, while considering the correlation criterion 2 in Table 2, an SCP with a lower cost than p_2 can be found: $p_3 = s \rightarrow a_3 \rightarrow b_3 \rightarrow c_3 \rightarrow t$, where its total cost is 450. Therefore, it is crucial to consider the correlation between the QoS of candidates to find a better SCP in the service composition process.

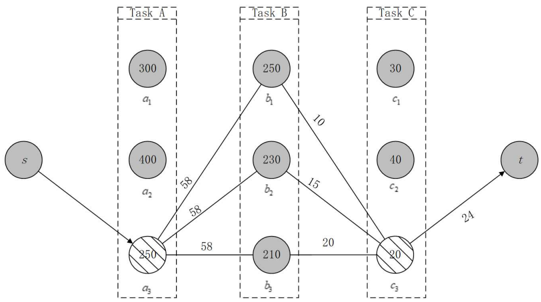
Table 2: Examples of correlation criteria

Correlation Criteria	Task	Candidate service	Service cost (yuan)	Network delay(ms)
Criterion 1	A	a_2	250	2.9
	B	b_2	180	2.5
Criterion 2	A	a_3	220	2.8
	C	c_3	20	2.4

It creates a solid foundation for the further design of service composition algorithms with QoS correlation by constructing the correlation between services using the auxiliary graph. Fig. 3 shows an example of constructing the correlation between services using the auxiliary graphs. A correlation between candidate services means a larger search space. Thus, the correlation can be considered to find a superior SCP. Constructing the auxiliary graph based on the correlation can flexibly express the search space increased due to the correlation, which is convenient for further designing service composition algorithms with QoS correlation.



(a) Search space constructed according to correlation criterion 1



(b) Search space constructed according to correlation criterion 2

Fig. 3: An example of constructing correlations between services based on the auxiliary graph

Fig. 3 (a) and Fig. 3 (b) show the search spaces constructed according to correlation criteria 1 and 2 in Table 2, respectively. In these figures, only the nodes with connection relationships represent a part of the search space increased due to the correlation criterion. In the process of searching for an SCP that meets the user requirements, the nodes without connection relationships can be deleted in advance. The search space caused by the correlation should be constructed according to the correlation between services to obtain a larger search space. It will most likely lead to a superior SCP.

The above application examples only consider cloud costs while selecting the optimal SCP. When there is a correlation between the QoS of network services (such as the network delay), the service composition becomes more complicated. According to Table 1 and Table 2, Table 3 presents the SCP obtained after considering the QoS correlation between network services. It can be seen from the data in this table that when the service cost drops, the network delay increases. Thus, considering the QoS correlation between cloud and network services is a multi-constraint service composition problem. Therefore, it is vital to provide an algorithm that can deal with the multi-constraint service composition problem to effectively improve users' QoS.

Table 3: SCP costs considering QoS correlation

Service path	Service cost(yuan)	Network delay(ms)
p1	540	3.3
p2	460	2.9
p3	450	2.8

3 QCSC algorithm

3.1 Composition service based on Skyline algorithm

Skyline algorithm [20] aims to filter out the representative data set, which is not dominated by other data from a large data set. This algorithm is applied to reduce the number of candidate service sets and reduce the time complexity of the service composition algorithm. When the other composition services do not dominate a single one, it belongs to Skyline. Since the Skyline composition service contains all possible "optimal" services, using Skyline provides the composition service agent to respond to user requests quickly. Each Skyline composition service corresponds to a certain combination of user preferences (the set of weights that users assign to each QoS attribute). If the user determines this combination of preferences, the composition service will be returned to the user. For non-Skyline composition services, no matter what weight the user determines, they will not be selected. Therefore, the composition service agents do not have to perform global search calculations. However, they should search for a composition service that satisfies the user's constraints and has the optimal QoS in the Skyline composition service, thereby improving efficiency.

Definition 4 Service Skyline is a set of Web services. Other services do not dominate each service S in the set of candidate services where S is located. Simultaneously, other composition services do not dominate each composition service set in the Skyline composition service.

The score of the service s with d-dimensional QoS attributes is $s = \sum_{i=1}^{d} q_i(s)$. If $s_1 \succ s_2 (cs_1 \succ cs_2)$, then $s_1 < s_2 (cs_1 < cs_2)$; that is, if the score of s_1 is lower than that of s_2, s_2 must not be able to dominate s_1, where cs represents a certain service composition.

A partial search strategy is performed, given an abstract composition service $ACS = \{S_1S_2 \cdots S_m\}$, including m abstract services, and given the Skyline service $\{SSKY_1SSKY_2 \cdots SSKY_m\}$ corresponding to each abstract service, the abstract service $S_1S_2 \cdots S_m$ in ACS can be obtained by solving the corresponding $SSKY_1SSKY_2 \cdots SSKY_m$ in the Skyline service.

3.2 QCSC algorithm based on the MCOP

The QCSC aims to eliminate the QoS correlation between network services and users. Considering the correlation between QoS leads to a larger search space and indicates better quality service composition solutions. Constructing an auxiliary graph to express the QoS correlation can effectively simplify the problem into a multi-constraint QoS optimal path selection problem. Therefore, this paper proposes the MCOPS algorithm to obtain a network service composition plan that meets the users' QoS requirements. The algorithm flow chart is shown in Fig. 4.

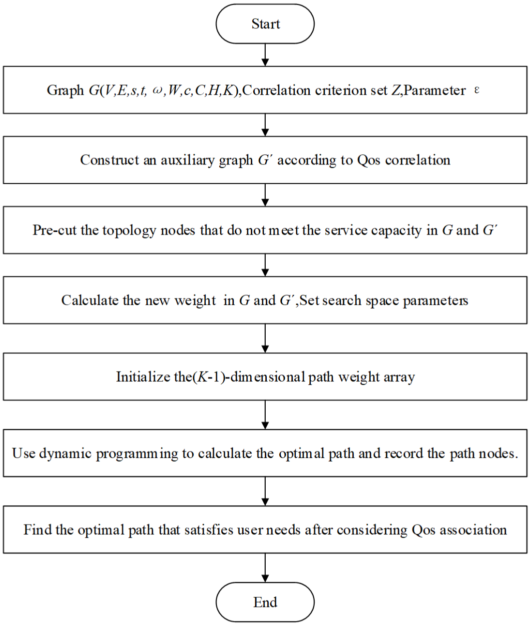


Fig. 4: MCOPS algorithm design process

The pseudo-code of the MCOPS algorithm of the network-aware service composition is shown in Algorithm 1.

Algorithm 1

Input: Graph $G\left(V,E,s,t,\delta,W,c,C,H,K\right)$, Parameter ε, correlative criterion $Z=\left\{z_i\left(\alpha,\beta\right)\mid 0\leq i\leq M\right\}$;

Output: Path p^{opt}

First, construct an auxiliary graph $G'_i\left(V',E'\right)$ similar to the graph $G\left(V,E\right)$for each correlation criterion $z_i\left(\alpha,\beta\right)\in Z$. Then, re-weight all the outgoing edges of the correlation node in G' according to the correlation criterion. For example, if three nodes are involved in a correlation criterion in the auxiliary graph G', three new weights will be given to their outgoing edges (the outgoing edges of the same node have the same weight). In other words, a new weight w'_i (e)will be given to a correlation node $v'_i\in a$ in a correlation criterion of the graph G'.

Delete the outgoing edges of other nodes in the same task (service class) as the correlation node $G'_i\left(V',E'\right)$; that is, delete all outgoing edges of the node $v\in\left\{S_h\backslash v'_i\right\}$

If $c_v<C_h, 1\leq h\leq H$, delete the incoming edges of other nodes in the same task (service class) as the correlation node v'; that is, delete all incoming edges of the node $v\in\left\{S_h\backslash v'\right\}$

Compute $\delta_k^N(e)=\left\lfloor\frac{\delta_k(e)}{W_k}\cdot\frac{H-1}{\varepsilon}\right)$, and set $\Lambda=\left|\frac{H+1}{\varepsilon}\right\rfloor$, $\quad 2\leq k\leq K$;

for $i=0$ to M **do**
    for $\lambda_k=0$ to $\Lambda, 2\leq k\leq K$ **do**
        $g_s^i\left(\lambda_2,\cdots,\lambda_K\right)=0,\quad g_v^i\left(\lambda_2,\cdots,\lambda_K\right)=\infty,\forall v\in V$
    end for
    for $\lambda_k=1$ to Λ **do**
        for $\forall(u,v)\in E\qquad\lambda_k\geq\delta_k^N(u,v),\quad v\neq s$ **do**
            $g_v^i\left(\lambda_2,\cdots,\lambda_K\right)=\min\left\{g_u^i\left(\lambda_2-\delta_2^N(u,v),\cdots,\lambda_K-\delta_K^N(u,v)\right)+\delta_1(u,v)\right\}$
            if $g_v^i\left(\lambda_2,\cdots,\lambda_K\right)>g_v^i\left(\lambda_2,\cdots,\lambda_j-1\cdots,\lambda_K\right)$ **then**
                $g_v^i\left(\lambda_2,\cdots,\lambda_K\right)=g_v^i\left(\lambda_2,\cdots,\lambda_j-1\cdots,\lambda_K\right),\quad 2\leq j\leq K$
            end if
        end for
    end for
end for
if $g_t^i\left(\lambda_2,\cdots,\lambda_K\right)\leq W_1$ **then**
    Find a $s-t$ path p^i s.t. $\delta_1\left(p^i\right)\leqslant W_1,\quad\delta_k\left(p^i\right)\leq\lambda_k$;
end if
if $\delta_k\left(p^i\right)\leq W_k$ **then**
    p^i is a feasible solution, path set $PS\leftarrow p^i$;
end if
if PS $\neq\varnothing$ **then**
    Select the optimal path p^{opt} based on $QCPWD$ in PS;
    return p^{opt};
end if
return No feasible path, EXIT.

The MCOPS algorithm constructs the auxiliary graph to express the QoS correlation. The search space is constructed based on the number of path hops, which is mainly composed of the following specific steps:

(1) Step 1 : for each correlation criterion $z(\alpha, \beta)$, an auxiliary graph G' is constructed similar to graph G. Then, all the outgoing edges of the correlation node in G' are re-weighted according to the correlation criterion. For example, in the auxiliary graph G', if three nodes are involved in a correlation criterion, three new weights will be given to their outgoing edges (the outgoing edges of the same node have the same weight). In other words, in a correlation criterion of the graph G', a new weight $\omega'(e) \in \beta$ will be given to a correlation node $v' \in \alpha$.

(2) Step 2: delete the outgoing edges of other nodes in the same task (service class) as the correlation node; that is, delete all outgoing edges of the node $v \in \{S_h/v'\}$.

(3) Step 3: similar to Step 2, delete the incoming edges of other nodes in the same task (service class) as the correlation node; that is, delete all the incoming edges of the node $v \in \{S_h/v'\}$.

After the above three steps, the minimum search space for the correlation criterion is retained in graph G'.

(4) Step 4: transfer the weights of all the correlation nodes involved in a correlation criterion to their outgoing edges. As shown in Figs. 1 and 2, the cloud cost in the weights of the correlation nodes is included in the service node, while another weight, such as the network delay, is initialized on the connection edge between services. In Algorithm 1, since this step is not necessary for actual applications, the pseudo-code of the QCSCA algorithm does not reflect this step.

(5) Step 5 (line 3): delete the topology. If the node cannot provide sufficient service capabilities, the node is deleted.

(6) Step 6 (line 4): calculate the new weight and the parameter Λ according to the number of path hops H.

(7) Step 7 (lines $6-8$) : initialize the array.

(8) Step 8 (lines $9-16$) : calculate the path, and inherit the known results continuously to establish the foundation for finding a superior SCP in the future.

(9) Step 9 (lines 18-23): find the feasible SCP. It should be noted that the program here can find multiple SCPs that meet the users' requirements.

(10) Step 10 (lines $24-28$) : select the optimal service composition and return it to the user.

3.3 Algorithm analysis

Theorem 1 *The time complexity of the MCOPS algorithm in the worst case [19] is* $O\left(mM\left(\frac{H}{\varepsilon}\right)^{K-1}\right)$.

Proof Step 1 (lines $1-2$), according to the QoS correlation, the required time to construct the auxiliary graph is $O(M(n+m))$; Step 2 (line 3), the required time to delete the topology node and its edges is $O((M+1)(n+m))$; Step 3 (Lines 4-17), the required time to calculate the composition path is $O\left(m(M+1)\left(\frac{H+1}{\varepsilon}\right)^{K-1}\right)$; Step 4 (Lines $18-28$), the required time to select the path is $O(K(M+1))$ Therefore, the time complexity of the MCOPS algorithm in the worst case is $O\left(nM+mM+nM+mM+n+m+mM\left(\frac{H+1}{\varepsilon}\right)^{K-1},+m\left(\frac{H+1}{\varepsilon}\right)^{K-1}+MK+K\right)$, QED □

Theorem 2 *The optimal SCP p^{opt} finally obtained by MCOPS at least satisfies $\omega_k\left(p^{opt}\right) \leq (1+\rho)\cdot\eta^{opt}\cdot W_k, \quad 2 \leq k \leq K$, where $\rho=\frac{\varepsilon}{\eta^{opt}}$ is the approximate rate.*

Proof (1)Since the MCOPS algorithm employs the number of paths hops to construct the search space, the known results are continuously inherited in the path finding process. Since the QoS correlation to construct the auxiliary graph is not related to the number of path hops, it can be known that its optimal search path method is subject to multiple constraints: According to Theorem 1, all SCPs $p \in \mathrm{PS}$ obtained by the MCOPS algorithm should satisfy $w_k(p) \leq (1+\rho)\cdot\eta^{\mathrm{opt}}\cdot W_k, \quad 2 \leq k \leq K$, where $\rho=\frac{\varepsilon}{\eta^{\mathrm{opt}}}$ is the approximate rate. (2)The optimal SCP $p^{\mathrm{opt}} \in PS$ is selected according to the simulation test evaluation index QoS correlation-based Path Weights Distance (QCPWD in this paper. Therefore, the optimal SCP p^{opt} obtained by the MCOPS algorithm finally satisfies at least $w_k\left(p^{\mathrm{opt}}\right) \leq (1+\rho)\cdot\eta^{\mathrm{opt}}\cdot W_k, 2 \leq k \leq K$, where $\rho=\frac{\varepsilon}{\eta^{\mathrm{opt}}}$ is the approximate rate, and W_k is the edge metric parameter, QED. □

4 Simulation test and analysis

4.1 Test preparation

The simulation employed a set of directed acyclic graphs as the test data set to evaluate the algorithm's performance effectively. Assuming that each directed acyclic graph had ten tasks (H=10), there were multiple candidate services in each task, while the number of services ranging from 100 to 1000 and two QoS parameters (K=2) was set on each edge in the directed acyclic graph. The first QoS parameter was randomly selected between [11, 15], while the second was randomly selected between [16, 20]. The correlation node's correlation value (new weight) was randomly selected between [5, 10]. The simulations were performed under the same test environment, the same data set, the same parameter settings, and the same data structure. Moreover, the same service composition plan was employed under the same service network node scale (SNNS) to evaluate the algorithm's performance.

4.2 Test index

In order to evaluate the performance and effectiveness of MCOPS, two evaluation indicators were utilized to reflect the solution time and the solution quality of the algorithm: Average Computation Time (ACT) and QoS Correlation-based Path Weights Distance (QCPWD) were employed to verify the comprehensive performance of the algorithm.

ACT was mainly employed to evaluate the algorithm indicators in terms of the network delay. The shorter the waiting time for the network user composition plan, the better the algorithm performance. QCPWD mainly considered the QoS correlation. The distance between the weights of each dimension of the SCP and the user constraints obtained after running the algorithm independently for 30 times is:

$$\mathrm{QCPWD}(p) = \sum_{k=1}^{k} \left[1 - \frac{\delta k(p)}{wk}\right] \tag{13}$$

QCPWD reflected the end-to-end QoS guarantee level of the SCP obtained by the algorithm for users. Obviously, the higher the QCPWD value, the better the service correlation composition path obtained after calculations.

4.3 Test results

This paper introduced the approximation parameter ε of literature [12] into the MCOPS algorithm based on the previous analysis. In the test, different ε values (ε=0.001, 0.01, and 0.01) were adjusted to analyze the impact of εon the QoS correlation. For the user end-to-end QoS request and the user QoS multiple constraints, the edge metric parameters W were set to W1=90 and W2=80 in all tests to ensure that the algorithm could find a feasible service composition in all service networks. In this test, the average time of the SCP was obtained by the union of the results of the algorithm's independent operation of 30 times for different values of ε.

Fig. 5 shows that the MCOPS algorithm could obtain the SCP that meets the users' requirements as the network service node scale increased in the interval [100-1000]. When the service node increased, the ACT value also increased, demonstrating that it took more time to find the trend of the same path in a larger-scale service network. The MCOPS algorithm mainly constructed the search space based on the number of paths hops in the SCP, which could quickly lock the range of feasible SCPs. In the test, the higher the parameter ε, the lower ACT spent searching for the path. This was because the smaller the algorithm parameter ε, the larger the path search space, and the algorithm should spend more time to find SCPs that meet user needs.

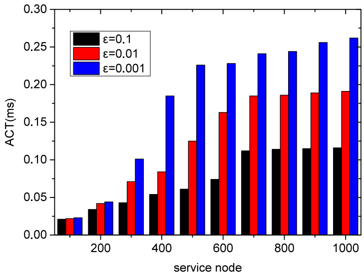


Fig. 5: ACT values of the MCOPS algorithm for different service numbers without considering the correlation scenario

Fig. 6 shows the simulation results of the solution time of the MCOPS algorithm with or without considering the correlation criteria (the number of correlation criteria is 3). As shown in Fig. 4, in the case of different scale service networks (scale [100,500]), when considering the QoS correlation between service nodes (the correlation is 3 at this point), the MCOPS algorithm needed more time to find a feasible and superior SCP, thereby leading to a higher ACT value.

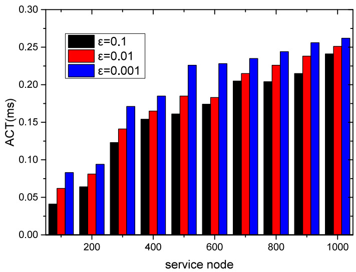


Fig. 6: ACT values of the MCOPS algorithm when ε varies in the interval [0.01, 0.03]

Fig. 7 shows the ACT values under the same service node (assuming 500) and different correlation numbers when the given value of ε was 0.001, 0.01, and 0.1. With the increase in the QoS correlation criteria (from 1 to 10), the MCOPS algorithm needed more time to find a feasible and superior SCP, thereby increasing the ACT. The reason was that when the auxiliary graph indicated that there was a QoS correlation between the candidate service nodes, the increase in the number of correlation criteria resulted in more auxiliary graphs, which was essentially to construct a larger path search space. At this point, it was necessary to consider the situation when there was no correlation (the original search space) and the increased part of the search space caused by the correlation, and the two together constituted a larger search space. Therefore, the MCOPS algorithm took more time to find the feasible service composition with superior quality.

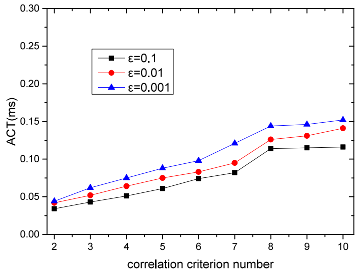


Fig. 7: ACT values under the same SNNS and different correlation criteria

Figs. 8 and 9 show the ACT values of the MCOPS algorithm under the same SNNC but the different correlation criteria. As the correlation ratio between services increased (20% to 100%), the MCOPS algorithm needed more time to calculate the feasible SCP, increasing the ACT. This was because when there was a certain proportion of correlations between candidate services in a service network of a certain scale, the time of the original path search space would increase after considering these correlations. With the increase of ε, the MCOPS algorithm would require less time to calculate the feasible SCP. Increasing the value of ε played an essential role in fine-tuning the search space. When ε was too large, the search space may be so small, which could not contain any feasible SPC. The value of ε completely depends on the dynamic settings from the service provider or user to adapt to the constantly changing network service environment.

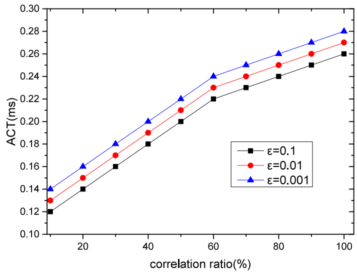


Fig. 8: ACT values of the MCOPS algorithm without considering the correlation ratio

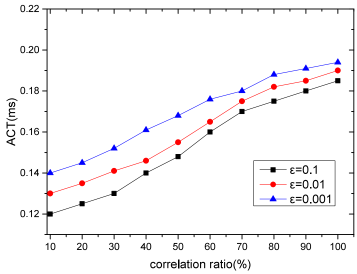


Fig. 9: ACT values of the MCOPS algorithm with considering the correlation ratio

Fig. 10 shows the ACT values of the MCOPS algorithm under different SNNSs but the same correlation ratio. For a given correlation ratio (10%), with the increase of the scale service node scale, the ACT value of the MCOPS algorithm gradually increased, demonstrating that the MCOPS algorithm required more time to obtain a feasible SCP. This was because the search space would increase under a correlation ratio in service scenarios of any scale. This also indicated that the MCOPS algorithm could stably and reliably find an SCP satisfying the end-to-end QoS multi-constraint under different service networks according to user needs.

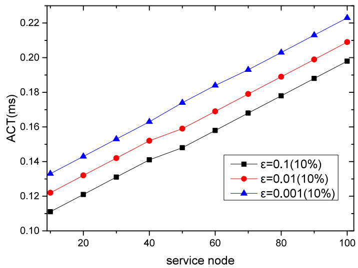
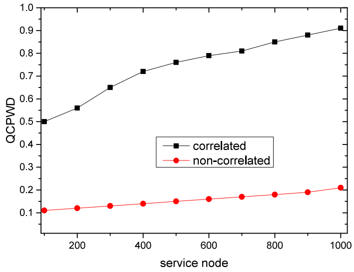
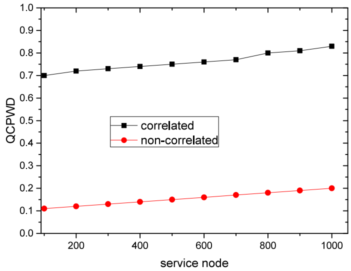


Fig. 10: ACT values under different SNNSs and the same number of the correlation criteria

Fig. 11 (a) and (b) show the QCPWD values of the MCOPS algorithm in solving the path quality under different SNNSs, the same correlation criterion, and the same correlation ratio. For a given number of correlation criteria (3) and correlation ratio (10%), for different scale service networks ranging in the interval [100,1000], when considering the number of correlation criteria between candidate services, Fig. 11 shows that the QCPWD value was above 0.5, which was much higher than the non-correlation MCOPS algorithm, resulting in a superior SCP. In other words, a superior SCP could be obtained considering the correlation between services.



(a) The number of correlation criteria is 3

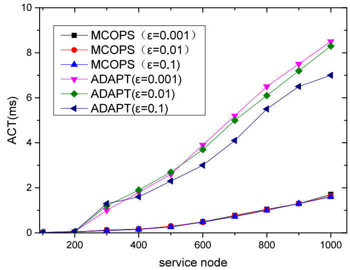


(b) The correlation ratio is 10%

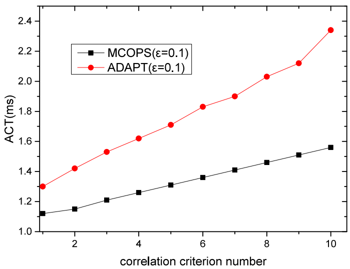
Fig. 11: QCPWD values under different SNNSs and the same number of correlation criteria

4.4 Performance comparison with ADAPT algorithm

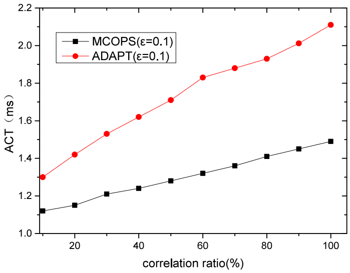
In order to evaluate the performance of the MCOPS algorithm compared with the previous algorithms, the performance of the MCOPS algorithm was compared with that of the ADAPT algorithm. Considering the service mode and service scale and the QoS parameter characteristics of network services, the main advantage of the ADAPT algorithm was that it continuously inherited the known results while finding the optimal solution. Fig. 12(a) compares the ACT values of the two algorithms under different network service nodes and the same ε values. The test results indicated that for a given ε, the MCOPS algorithm could find the same SCP faster than the ADAPT algorithm. This means that the MCOPS algorithm had a smaller ACT value. As the number of services increased, its advantages became more obvious. The ADAPT algorithm employed the "scaling and rounding" technology to construct the search space where the feasible SCP was located by continuously compressing the upper and lower boundaries of the path. This process was complicated and time-consuming. Therefore, the MCOPS algorithm had more obvious advantages in ACT performance compared with the ADAPT algorithm.



(a) ACT values of the MCOPS and ADAPT algorithms under different service network nodes



(b) ACT values of the MCOPS and ADAPT algorithms under different correlation criteria



(c) ACT values of the MCOPS and ADAPT algorithms under different correlation ratios

Fig. 12: A comparison between the performance of MCOPS and ADAPT algorithms

Figs. 12 (a), 12 (b), and 12 (c) compare the ACT values of the MCOPS and the ADAPT algorithms under different service numbers, different correlation criteria, and different correlation ratios, respectively. For a given value of , as the number of QoS correlation criteria increased, the MCOPS algorithm required less calculation time than the ADAPT algorithm to find a feasible

SCP that meets the user requirements. Therefore, the MCOPS algorithm could find a feasible SCP that meets user needs faster than the ADAPT algorithm while considering the QoS correlation between candidate services. According to Fig. 12 (c), the advantage of the MCOPS algorithm was more evident with the increase of the correlation ratio between candidate services.

5 Conclusion

This paper fully studied the QCSC problem of multi-constraint paths. Firstly, the QCSC was modeled, and the auxiliary graph was then introduced to construct an additional search space to express the correlation between services. Accordingly, the QCSC problem was transformed into a special MCOP problem. Secondly, an MCOPS algorithm was proposed based on the Skyline algorithm of service composition, which could solve the QoS correlation between services. The simulation experiment demonstrated that the proposed algorithm performs well in solution time, solution quality, and QCPWD, indicating that the MCOPS algorithm could quickly and effectively find an SCP with superior quality. The performance of the MCOPS algorithm was superior to that of the ADAPT algorithm. Finally, the study on service composition algorithms also provided a good theoretical foundation for today's Internet era.
Funding Statement: This work was funded by the E-commerce Engineering Center Fund of Fujian Province, China, Grant Number KBX2106; the " Thirteenth Five-Year Plan" Development Fund of Fujian Province, China, Grant Number FJKCG20-058; and the Science and Technology Bureau Fund of Sanming City, Fujian Province, China, Grant Number 2020-S-12.
Conflictsof Interest: The authors declare that they have no conflicts of interest to report regarding the present study.
Data Availability: Data sharing not applicable to this article as no datasets were generated or analyzed during the current study.

References

[1] Y. Z, Zhou, D. Zhang. "Near-End Cloud Computing: Opportunities and Challenges in the Post-Cloud Computing Era," *Chinese Journal of Computers*, vol. 42,no. 4,pp.677-700,2019.

[2] K.Yang, X. L. Ding; J. P. Liu. "Research and application of smart edge computing in IoT," *Telecommunications Science*, vol.35,no.S2,pp.176-184,2019.

[3] N. Anjum, Z. Yang, I. Khan, M. Kiran, F. Wu et al., "Efficient algorithms for cache-throughput analysis in cellular-d2d 5g networks," *Computers, Materials & Continua*, vol.67,no.2,pp.1759–1780,2021.

[4] R. Sankaran,. S. Chakraborty, "Why customers make mobile payments? Applying a means-end chain approach,"*Marketing Intelligence &*

Planning,vol.39,no.2,pp.109-124, 2021.

[5] K. L. Rao ,. "Research on Optimization of Service Composition Based on T-QoS Awareness," Ph.D.dissertation. Nanjing University of Posts and Telecommunications, china,2015.

[6] C. L. Li ; Z. H. Deng., "On the QoS of Cloud Computing," *Library & Information,* vol.4,no.4,1-5. 2012.

[7] X. B. Xiong ;R. Yang. Computing Arbitrary Granularities of Service Process with QoS Correlations [J]. *Journal of Chinese Computer Systems*, vol.39,no.7,pp.1565-1568. 2018.

[8] Z. H. Zhang ;X. T. Zhang, "Research on High-speed Railway Information Service Combination Method Based on Association Rules," *Journal of the China Railway Society*, vol.43,no.4,pp.85-94, 2021.

[9] M. BarakatL,,M. Luck , "Efficient Correlation-aware Service Selection,"P*roceedings of 2012 IEEF International Conference on Web Services (ICWS)*,New Yord,NY,USA, pp.1-8, 2012.

[10] F. Wagner,A. Klein, B. Kloppe *et al*, "Multi-objective Service Composition with Time- and Input-Dependent QoS,"*Proceedigs of 2012 IEEE 19th Inenaional Conference on Web Services(ICWS)*, New Yord,NY,USA,pp. 234-241, 2012.

[11] L. Ren; M. l. Ren, "A Improved Skyline Manufacturing Service Composition Optimization Approach Based on QoS Collaboration Association," *Operations Research and Management Science*, vol.30,no.4,pp.122-127,2021.

[12] Y. L. Zhang; Q. Shao; L. J. Li *et al,* "Reliable Web Service Composition Optimization Method Based on Genetic Clustering,". *Journal of Chinese Computer Systems*, 2020, Vol.41,no.5,1030-1035,2020.

[13] E. Funda, S. Rakesh, Z. Lisa, "An improved FPTAS for Restricted Shorted Path" *Information Processing Letters*,vol.83,no.5,pp.287-291,2002.

[14] G. Xue,W. Zhang, J. Tang *et al*, "Polynomial Time Approximation Algorithms for Mult-cRouting,"·*IEEE/ACM Transactions on Networking*, vol.6,no.3,pp.656-669, 2008.

[15] J. Huang; Q. Duan, "Network-cloud integrated service quality assurance,"in *proc Electronics Industry*, Bei Jing,BJ,China,pp.237-241,2018.

[16] T. Yu; Zhang Y; Lin K J, "Efficient Algorithms for Web Services Selection with End-to-End QoS Constraiints,". *ACM Transactions on the Web*,

vol.1,no.1,pp.1-26,2007.

[17] Q. Y. Lin, "Summary of Research on Association Rules Mining Algorithms,"*Soft Ware Guide*, vol.11,no.6,pp.27-29,2012.

[18] J. Y. Yao, "The principle and application of data warehouse and data mining technology," in *proc Electronics Industry*, Bei Jing,BJ,China,pp.158-160,2009.

[19] L. Xiong. "Research on the Selection Mechanism of Composite Services Supporting Semantic Association,"M.S.disseration, Nanjing University of Posts and Telecommunications, China,2013.

[20] L. G. Dong, G. H. Liu, "Algorithm for Solving and Updating Combinatorial Skyline," *Computer Engineering,* vol.43,no.6,pp.195-201.206,2017