Computing Uncertain Skyline of Web Services via Interval Number

Guobing Zou,Mei Zhao
School of Computer Engineering and Science
Shanghai University,China
Email: gbzou,zhaomei@shu.edu.cn

Sen Niu
School of Computer Engineering and Science
Shanghai University,China
Email: sniu@shu.edu.cn

Yanglan Gan*
School of Computer Science and Technology
Donghua University,China
Email: ylgan@dhu.edu.cn

Bofeng Zhang
School of Computer Engineering and Science
Shanghai University,China
Email: bfzhang@shu.edu.cn

***Abstract*—QoS values may significantly vary due to the invocation of Web services under dynamical network environment. Although some of approaches apply uncertain QoS to computing the skyline for the reduction of the number of candidate services, they have no uncertain QoS model that needs to be aligned to realistic invocation and execution of Web services. To solve the issue, this paper presents a novel approach to uncertain service skyline via Chebyshev's inequality and interval number. We model uncertain QoS of a Web service by a QoS matrix and then each dimension is shrank to an uncertain QoS scope by interval number. Finally, based on the QoS model, we propose an uncertain service skyline algorithm to compare the QoS of two Web services with domination relationship strategy. Extensive experiments have been conducted on 1,558,224 Web service invocation records. The experimental results demonstrate the effectiveness of our proposed approach.**

***Keywords*—Web service, Uncertain QoS, Skyline, Region partitioning, Interval number**

I. INTRODUCTION

Web services are self-describing software components that can be advertised, located and invoked across the Internet[1].The development of technologies for Web services is expected to change the way of conducting business on the Web[2].Therefore, there may be multiple service providers competing to offer the same functionality, but with different QoS. Thus, QoS is a crucial criterion for selecting among functionally similar Web services.

Finding the perfect Web service which is the best in all QoS attributes is ideal for user. Unfortunately, such a Web service is seldom found[3].Traditionally, the non-functional evaluation of a service is determined by computing an overall score that aggregates individual QoS values. Users are required to assign weights to QoS attributes. However, it is a difficult task to assign a set of weights for their corresponding QoS attributes. To solve the problem, computing the service skyline based on QoS has been a popular solution for service optimization paradigm among services with the same functionality[4][2][5].

The network environment is dynamically changing for the invocation of Web services. QoS values may significantly vary due to the update of server or workload changes. Moreover, some of the selected services may suddenly become unavailable at run-time, while new service candidates with the same functionality may be launched[6][7].In such a case, some efforts[3][8][2] have been done to take the uncertainty into account. K. Benouaret, et al. presented each QoS attribute using a possibility distribution[3].However, it is difficult to acquire the accurate possibility distribution in real-world applications. S. Wang, et al. proposed an uncertain QoS-aware skyline service selection approach. By applying cloud model[8]to compute the uncertainty of each QoS attribute, it adopted skyline computing to prune redundant services. A novel approach is investigated for computing the service skyline from uncertain QoWS[2].However, the time complexity is too high to find service skyline from available candidate services. As a consequence, how to accurately assess the model of uncertain QoS for a Web service and further efficiently filter out those services with low QoS has become a research challenge to be solved.

In this paper, we proposed a novel approach to computing the skyline of Web services with uncertain QoS. First, we formulate the QoS model with a matrix that consists of a set of uncertain QoS transactions. Then, we get the interval number of every deminsion of a service by Chebyshev Inequality. Based on the ranking method of interval numbers, we presented a novel strategy called interval-dominant of service for uncertain QoS comparison. Finally, we present a algorithm to compute the internal-dominant skyline. Extensive experiments have been conducted on real-world QoS dataset. The experimental results validate the effectiveness and efficiency of the proposed uncertain service skyline approach.

The rest of this paper is organized as follows. Section 2 provides the preliminaries. Section 3 formulates the problem with uncertain QoS. Section 4 presents our approach for computing uncertain skyline. Section 5 shows the experimental results. Section 6 reviews the related work.Finally, Section 7 concludes the paper.

978-1-5090-2239-7/16/$31.00 copyright 2016 IEEE
SNPD 2016, May 30-June 1, 2016, Shanghai, China

II. PRELIMINARIES

In this section, we first present a reminder about skyline on certain QoS. Then, we provide the basic notions of Chebyshev's inequality and interval number.

A. Skyline on certain QoS

Skyline computation has received significant consideration in database query optimization[9][10].For a d-dimensional dataset, the skyline consists of a set of points, each of which is not dominated by any other points. Specifically, a point $\vec{p}$ $(p_1,\cdots,p_d)$ dominates another point$\vec{r}(r_1,\cdots,r_d)$,if $\forall i \in [1,d], p_i \geq r_i$ and $\exists j \in [1,d], p_j > r_j$.We use the notation ($\geq$) to represent better than or equal to and the notation ($>$) to represent better than.

In the context of Web services, we exploit dominance relationship to compare the QoS values between two Web services. A service skyline can be regarded as a set of services, where each of them is not dominated by others in terms of all the QoS attributes. Those services that are not involved in a service skyline can be pruned to reduce the number of candidate services for further service-oriented applications.

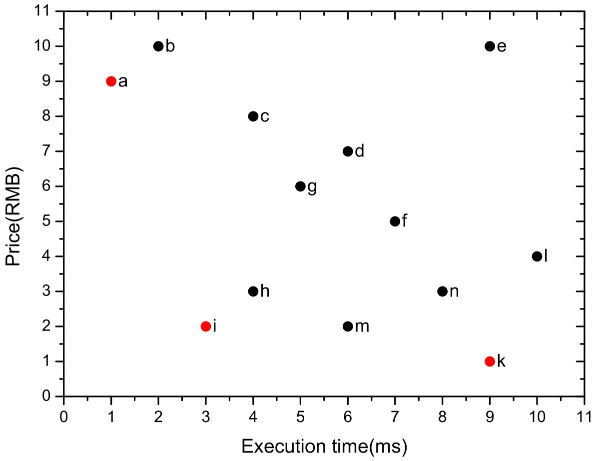


Figure 1. Example of skyline services

Figure 1 shows an example of service skyline of a certain service class. Each service is described by two QoS attributes, namely execution time and price. Hence, QoS values of each service can be represented as a point in the 2–dimensional space. We can observe that service a belongs to the skyline, because it is not dominated by any other services, i.e. there is no other service that simultaneously offers both shorter *execution time* and lower *price* than a.It holds for the services i and k, which are also included in the skyline. On the other hand, the other services are not contained in the skyline, because they are dominated by the skyline services. For example, services b is dominated by service a and service h is dominated by service i.

B. Chebyshevs inequality

Chebyshevs inequality holds that at least $1 - 1/k^2$ of data from a sample must fall within k standard deviations from the mean, where k is any positive real number greater than one. Chebyshevs inequality provides a way to know what fraction of data falls within k standard deviations from the mean for a given data set. Therefore, if a random variable x has a finite mean μ and finite variance σ^2, then for all $k > 1$,

$$P(|x - u| \geq k\sigma) \leq \frac{1}{k^2} \tag{1}$$

C. Interval number

Definition 1 (Interval number**).** Interval number[11]is a set of real numbers on a closed interval. Let R be the set of real numbers, the interval number X,$X = [x^L, x^U]$,where$x^L$is the lower bound of interval and x^U is the upper bound of interval.If $x^L = x^U$,the interval number X is a real number. The general rules of operation are similar to the operation rules of the set.

Definition 2 (Possibility**).** Given two interval numbers $a = [a^L, a^U]$ and $b = [b^L, b^U]$, let $L(a) = a^U - a^L$ and $L(b) = b^U - b^L$, then the possibility $p(a \geq b)$ is defined as[12]:

$$p(a \geq b) = \begin{cases} 1 & b^L \leq b^U \leq a^L \leq a^U \\ 1 - \frac{1}{2}\frac{(b^U - a^L)^2}{L(a)L(b)} & b^L \leq a^L \leq b^U \leq a^U \\ \frac{1}{2}\frac{(a^L + a^U - 2b^L)}{L(b)} & b^L \leq a^L \leq a^U \leq b^U \\ \frac{1}{2}\frac{(2a^U - b^U - b^L)}{L(a)} & a^L \leq b^L \leq b^U \leq a^U \\ \frac{1}{2}\frac{(a^U - b^L)^2}{L(a)L(b)} & a^L \leq b^L \leq a^U \leq b^U \\ 0 & a^L \leq a^U \leq b^L \leq b^U \end{cases} \tag{2}$$

It has been proved that interval number has the following properties:
(1)$0 \leq p(a \geq b) \leq 1; p(a \geq b) = 1$,iff $b^U \leq a^L$; $p(a \geq b) = 0$,iff$a^U \leq b^L$;
(2)$p(a \geq b) + p(b \geq a) = 1$, where the special case is $p(a \geq a) = \frac{1}{2}$;
(3)if $p(a \geq b) \geq \frac{1}{2}$ and $p(b \geq c) = \frac{1}{2}$, then $p(a \geq c) \geq \frac{1}{2}$;

At present, the ranking method based on possibility is usually used to sort the interval numbers.

III. PROBLEM FORMULATION

In this section, we focus on the understanding of computing skyline services from uncertain QoS of Web service by a set of formal definitions.

Definition 3 (Uncertainty of service**).** A Web service s is 3-tuple $s =< I, O, Q >$, where $< I, O >$ are the inputs and outputs for service functionality. Q represents the non-functional performance with uncertainty. $s.Q$ is denoted as the QoS of s.

For the representation of non-functional performance of a Web service, we define the QoS criteria as below.

Definition 4 (QoS criteria**).** Given a Web service $s =< I, O, Q > \in S$, its uncertain QoS $s.Q$ is aligned by a set of QoS attributes, $QS = \{q_1, q_2, \cdots, q_n\}$, where each q_i is used to represent one facet of non-functional uncertain QoS values of s.

Definition 5 (Transaction log)**.** Given a service $s \in S$ and $QS = \{q_1, q_2, \cdots, q_n\}$, a transaction log t is a history record by the invocation and execution of s for one time, denoted as $t = \{q_1(s), q_2(s), \cdots, q_n(s)\}$.

Under the real condition, different quality criteria have different computation ways. Some attributes must be computed from not only one transaction log, such as reliability. The reliability of a service s is the probability that a request is correctly responded within the maximum expected period of time. The value of the reliability is computed from historical data during the past invocation and execution using the formula $q_{rel}(s) = N_c(s)/K$, where$N_c(s)$is the number of times that the service s has been successfully delivered within the period of time, and k is the total number of invocations. Every transaction log in our experiment dataset has a value of response HTTP code about its running states, thus we can compute the reliability of a service s.

Example 1. Consider a Web service P that offers weather forecast service as shown in Table I. The uncertain QoS performance of P is recorded by a set of transaction logs, which capture QoS values in practice. Transaction logs can be obtained from monitoring mechanisms of Web services. Here, we only consider eight number of transaction logs $\{t_{1,} \cdots, t_8\}$, although the actual number is much larger. The dynamic environment causes the QoS uncertainty of its performance. Thus, given a service s and a set of QoS criteria $QS = \{q_1, q_2, \cdots, q_n\}$, transaction logs of s consist of a finite set of Web services transactions, denoted as $T(s) = \{t_1, t_2, \cdots, t_m\}$, which indicates the uncertainty of service QoS.

Table I
A SET OF TRANSACTION LOGS OF A WEB SERVICE P

Web Service: P (WSID16432)	
Response Time (ms)	*HTTP Code*
58	200
60	200
50	200
90	200
125	200
180	200
75	200
43	200

Definition 6 (Uncertain QoS of Web service)**.** Given a Web service $s \in S$, a set of QoS criteria with n attributes $QS = \{q_1, q_2, \cdots, q_n\}$, and its transaction logs $T(s) = \{t_1, t_2, \cdots, t_m\}$ with a period of time, the QoS uncertainty of s can be formalized as a matrix M_{m*n}:

$$M(s) = \begin{pmatrix} t_1 \\ \vdots \\ t_m \end{pmatrix} = \begin{pmatrix} q_{11} & q_{12} & \cdots & \cdots & q_{1n} \\ \vdots & \ddots & & & \vdots \\ \vdots & & \ddots & & \vdots \\ \vdots & & & \ddots & \vdots \\ q_{m1} & q_{m2} & \cdots & \cdots & q_{mn} \end{pmatrix} \tag{3}$$

where m and n are the number of transaction logs of service and QoS criteria for each transaction log, respectively. That is, each row t_i is a transaction log and each column row q_j represents a set of values on the j^{th} QoS criterion across m transaction logs.

Example 2. Lets also take the Web service P as an example, the uncertain QoS matrix of P is modeled as:

$$M(P) = \begin{pmatrix} 58 & 60 & 50 & 90 & 125 & 180 & 75 & 43 \\ 200 & 200 & 200 & 200 & 200 & 200 & 200 & 200 \end{pmatrix}^T \tag{4}$$

Given an uncertain Web service repository which including uncertain Web services $S = \{s_1, s_2, \cdots, s_N\}$ and its uncertain QoS matrix $M(s_i)$ of every uncertain services_i, we aim at how to efficiently select skyline services from S.

IV. APPROACH

To model the QoS uncertainty and compute the service skyline from an uncertain Web service repository S, we proposed a novel approach that integrates uncertain QoS modeling with service dominance relationship strategy. As a result, we propose a algorithm for computing uncertain service skyline.

A. Framework of Computing Uncertain Service Skyline

To illustrate the process of computing service skyline with QoS uncertainty, an overall framework is shown in Figure 2.

Its input consists of N number of uncertain services in a service repository, each of which has a set of associated transaction logs. The outputs of our approach include those skyline services extracted from the uncertain Web service repository. More specifically, the framework goes through three crucial steps. First, we extract an uncertain QoS matrix M for each service from S to represent its QoS uncertainty during its invocations and executions. Then, we apply Chebyshevs inequality to calculate the interval number on each QoS criterion of a Web service. Finally, we compute the skyline services by the skyline algorithm with the strategy of dominance relationship with QoS uncertainty.

B. The evaluation of service uncertain QoS

Given an uncertain service repository S,donated as $S = \{s_1, s_2, \cdots, s_N\}$, a set of QoS attributes $QS = \{q_1, q_2, \cdots, q_n\}$ and m transaction logs $T(s) = \{t_1, t_2, \cdots, t_m\}$ of s_i, we evaluate uncertain QoS of each Web service s_i as n-dimensional vector, where every dimension $q_l (1 \leq l \leq n)$ is represented as an interval number $[\mu_l - 3\sigma_l, \mu_l + 3\sigma_l]$,$\mu_l$ is a finite mean of the lth dimension and σ_l is the finite standard deviation that can be computed from transaction logs $T(s)$ based on the Chebyshevs inequality.If $\mu_l - 3\sigma_l \leq \min(q_{jl})(1 \leq j \leq m)$, we set the minimum as the lower bound of the interval number. If $\mu_l + 3\sigma_l \geq \max(q_{jl})(1 \leq j \leq m)$, we set the maximum as the upper bound of the interval number.

Example 3. Lets take the uncertain QoS matrix of Web service P in example 2 as an example. The uncertain QoS matrix is $M(P)$, from which we can calculate the internal

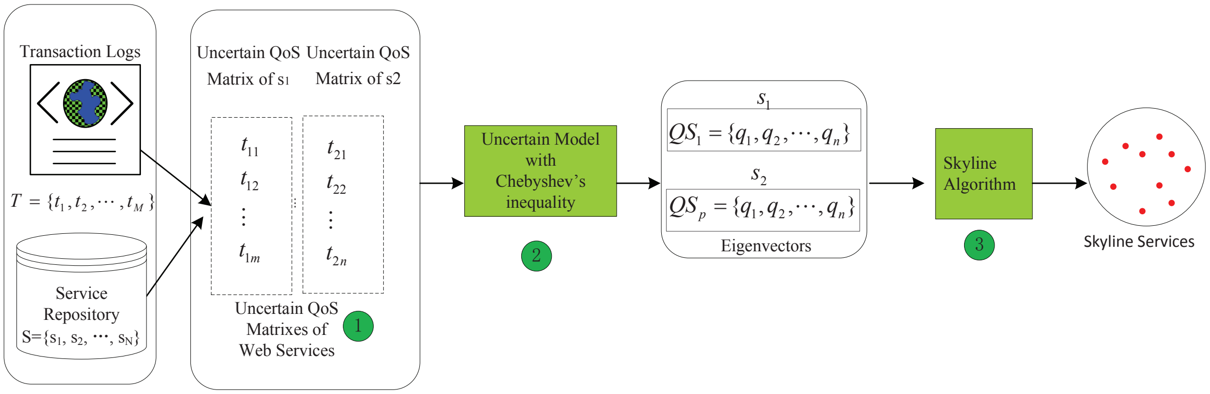


Figure 2. The framework of our approach for computing service skyline with QoS uncertainty

number on every dimension based on Chebyshevs inequality. As a result, the average and standard deviation on response time are $\mu = 85.125$ and $\sigma = 46.43$, respectively. For the reliability of the P service, the results are $\mu = 1$ and $\sigma = 0$. Therefore, the uncertain QoS of P can be evaluated as $QS(P) = ([43, 180], 1)$.

C. The dominance relationship strategy of uncertain QoS

Based on the dominance relationship on certain QoS of Web service, we extend it to adapt for the comparison between Web service with QoS uncertainty.

Definition 7 (Domination among uncertain Web services)**.** Given an uncertain service repository $S = \{s_1, s_2, \cdots, s_N\}$, a set of QoS attributes $QS = \{q_1, q_2, \cdots, q_n\}$.Let$s_i, s_j \in S$ be two uncertain services, the representation of every dimension in a service QoS matrix is an interval number. Under these assumptions,s_i dominates s_j, if $\forall i \in [1, n], s_i.q_k \geq s_j.q_k$ and $\exists k \in [1, n], s_{i.}q_k > s_j.q_k$.We use $\geq$ to represent better than or equal to and $>$ to represent better than.

For interval number we define the following rules:

Lemma 1 (better than or equal to)**.** *For two interval numbers* $a = [a^L, a^U]$ *and* $b = [b^L, b^U]$*, we have a betterthanorequalto b if* $p(a \geq b) \geq \frac{1}{2}$.

Lemma 2 (better than)**.** *For two interval numbers* $a = [a^L, a^U]$ *and* $b = [b^L, b^U]$*, we have a betterthanorequalto b if* $p(a \geq b) > \frac{1}{2}$.

We set "$\frac{1}{2}$ " as the threshold for identifying the dominance relationship of interval numbers between two uncertain Web services. The threshold λ can be adjusted to adapt for different application conditions.

D. Uncertain Skyline Algorithm

In this section, a algorithm of computing service skyline is presented based on the above results. The pseudo code of our proposed service skyline algorithm is as follows.

Uncertain skyline algorithm take services with transaction logs and possibility threshold λ of dominance relationship as input. And take the skyline of Web services as result. At first, we get the uncertain model by computing the interval number on every dimension of QoS. Next, we conduct experiment on these services model to compute the skylines depending on the dominance relationship. The main idea of the algorithm is that at the beginning the list of skyline contains the first service of the service list, while for each subsequent service p, there are three cases: (i) If p is dominated by any service in the skyline list, it is discarded as it is not part of the skyline, (ii) if p is dominated any service in the skyline list, it is inserted, and all services in the skyline list dominated by p are dropped and (iii) if p is neither dominated by ,nor dominates, any point in the skyline list is simply inserted without dropping any point.

Algorithm 1: Uncertain Skyline Algorithm

```
Input:a set of functionally similar Web service$S = \{s_1, s_2, \cdots s_N\}$, every service with Transaction logs $T(s_i)$, possibility threshold$\lambda$of dominance relationship;
Output:the service skyline $Skys$
Steps:
(1) Given transaction logs $T(s_i)$, we extract uncertain QoS matrix $M(s_i)$ as
    $M(s_i) = \begin{bmatrix} t_1 \\ \vdots \\ t_{\mathrm{m}} \end{bmatrix} = \left\{ \begin{array}{ccccc} x_{11} & x_{12} & \cdots & \cdots & x_{1n} \\ \vdots & \ddots & & & \vdots \\ \vdots & & \ddots & & \vdots \\ \vdots & & & \ddots & \vdots \\ x_{m1} & x_{m2} & \cdots & \cdots & x_{mn} \end{array} \right\}$
    Where, m and n are the number of the transaction logs and QoS criteria;
(2) We have a service list $S = \{s_1, s_2, \cdots s_N\}$
    $Skys \leftarrow s_j \in S$;$S = S - s_1$; Boolean isSkyline;
    foreach $s_i \in S$ do
        $isSkyline \leftarrow true$;
        foreach $s_k \in Skys$ do
            if $Dominates(s_k, s_i, \lambda)$ then
            $isSkyline \leftarrow flase$;
            if $Dominates(s_i, s_k, \lambda)$ then
            remove $s_k$ from $Skys$;
        if $isSkyline$ then insert $s_i$into Skys;
    return $Skys$;
```

V. EXPERIMENTAL EVALUATION

A. *Experimental setup and dataset*

To demonstrate the effectiveness and efficiency of our approach , we implemented a prototype system that integrates the uncertain QoS representation and uncertain dominance relationship strategy in Java. All the experiments are run on a PC in Windows 7 operating system. We validate our approach on real world dataset WS-DREAM. It can be found and downloaded from [13]. WS-DREAM dataset contains the number of 1,558,214 service invocation records (i.e., transaction logs), which are invoked by 150 service requesters from 27 countries all over the world about 100 Web services.

B. *Finding uncertain service skyline*

We analyze the uncertain service skyline concerning the parameters λ(possibility threshold),k(the parma of Chebyshevs inequality),n(the size of service repository) on the dataset.

We conduct experiment on 100 Web services by regulating those three parameters. To compare the effectiveness of calculate the interval number of uncertain QoS in a service matrix; we apply another model Min_Max to compute interval number of every dimension on every service. Min_Max obtains an interval number via the minimum value of the dimension as the lower bound and the maximum value as the upper bound.As the changes of λ and k, the size of uncertain service skyline can be analyzed with the parameter $n = 100$ by Chebyshevs inequality and Max_Min, as shown in Figure 3.

From the above experimental results, we find that the size of service skyline raises with the increasing probability threshold of λ. In addition, the size of service skyline of two calculation ways of interval number are affected by the parameter k.

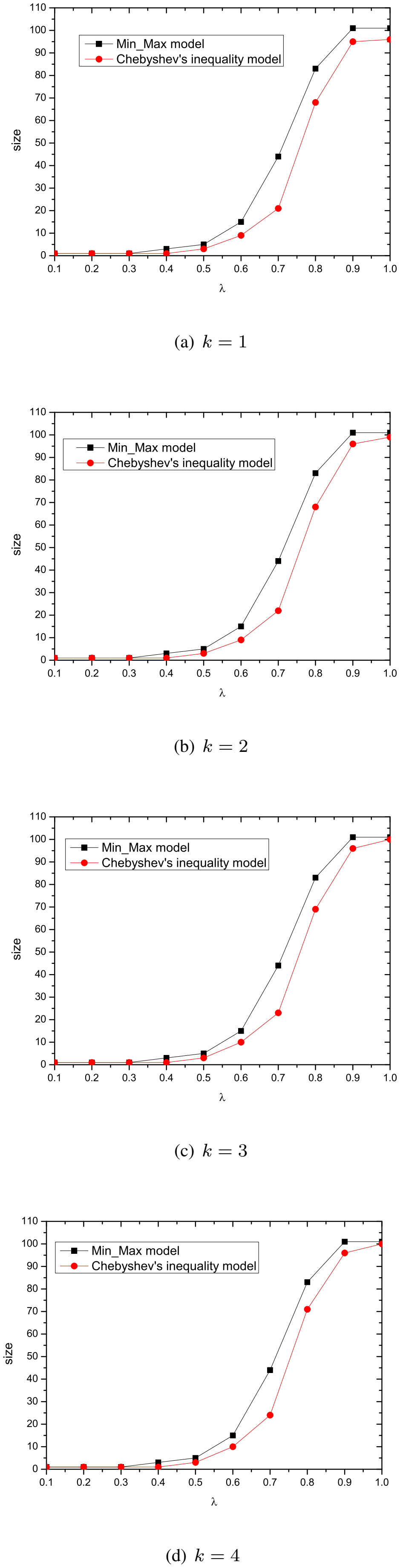
C. *Performance of finding service skyline*

Experiment is done to evaluate the performance of finding an uncertain service skyline. We test the time used for finding service skylines on the 100 uncertain Web services with different probability thresholdλ. Figure 4 illustrates the computational time on the dataset, From the results, we find that time consumption of finding a service skyline becomes larger along with the increase of λ.

VI. RELATED WORKS

The problem of QoS-based service selection has received considerable attentions in service computing community. QoS computation models have been proposed in some of the previous works [14] [15] [16]. The authors find the optimal selection by linear programming techniques. Moreover, some artificial algorithms have been applied in service selection with multiple dimensional global constraints on QoS[17]. However, these approaches mainly focus on certain QoS and service users need to assign a set of numeric weights on QoS.

Skyline analysis was first introduced into database domain by Bőrzsőnyi[9]. It can also be employed in Web services for QoS evaluation. Given a set of points in a d-dimensional space, the skyline is defined as the subset containing those points that are not dominated by any other points. Recently,



(a) $k = 1$

(b) $k = 2$

(c) $k = 3$

(d) $k = 4$

Figure 3. The size of uncertain service skyline along with the changes of parameters λ and κ

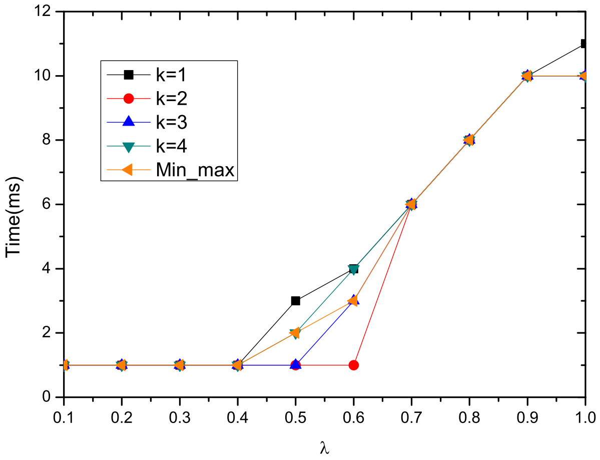


Figure 4. Performance of finding service skyline

skyline computation has been adopted in Web service selection to reduce search space of service selection and shorten the computation time in dynamic composition of Web services .The work in [18] proposes a novel skyline maintain algorithm which is suitable for dynamic service environment. However, they aim at finding service skyline under certain service QoS. Thus, the evaluation of uncertain QoS of Web services was not addressed by these works.

The work presented in [2] addresses the problem of uncertain QoS and computed the skyline from service providers. The authors define a concept called $p - dominant$ skyline, which consists of those service providers S that are not dominated with a probability p by any other service providers. In [3], the authors present each QoS attribute of a Web service using a possibility distribution and introduce two skyline extensions on uncertain QoS of Web service called $pos - dominant$ skyline and $nec - dominant$ skyline. Although possibility theory is used to tackle the problem of computing service skyline for uncertain QoS, the QoS values and their possibility degree distribution are hardly measured in real-world applications. Therefore, we model uncertain QoS of Web services as a matrix with a set of transaction logs. As a result, we apply our strategy of uncertain service dominance relationship to achieve the comparison between two uncertain services.

VII. CONCLUSION

To solve the issue on uncertain QoS of Web services without weight assignment, this paper proposes a novel approach to computing uncertain service skyline. That can be further applied in the optimization of service selection and efficiency improvement of dynamic composition of Web service. We first formulate the uncertain QoS of a Web service as a matrix, where each row represents a transaction log across QoS criteria and each column stands for a set of values on a QoS criterion across multiple transaction logs. Under this uncertain QoS formulation, each demination can be evaluated to gather interval numbers together as a vector by two ways, including Chebyshev inequality and Min_MaX. Finally, we present an uncertain service dominance relationship strategy to achieve the comparison of service QoS. We evaluate our approach experimentally on large-scale real-world dataset with 1,558,214 service invocation transaction logs. The experimental results demonstrate the effectiveness and efficiency of the proposed approach. As for the future work, we plan to investigate on how to maintain and update skyline as the number of transaction logs keep growing and extend the work further service related applications, such as QoS-based Web service composition.

REFERENCES

[1] F. Curbera, M. Duftler, R. Khalaf, W. Nagy, N. Mukhi, and S. Weerawarana, “Unraveling the web services web: an introduction to soap, wsdl, and uddi,” *IEEE Internet computing*, no. 2, pp. 86–93, 2002.

[2] Q. Yu and A. Bouguettaya, “Computing service skyline from uncertain qows,” *IEEE Transactions on Services Computing (SCC)*, vol. 3, no. 1, pp. 16–29, 2010.

[3] K. Benouaret, D. Benslimane, and A. Hadjali, “Selecting skyline web services from uncertain qos,” in *2012 IEEE Ninth International Conference on Services Computing (SCC)*. IEEE, 2012, pp. 523–530.

[4] M. Alrifai, D. Skoutas, and T. Risse, “Selecting skyline services for qos-based web service composition,” in *Proceedings of the 19th international conference on World wide web*. ACM, 2010, pp. 11–20.

[5] K. Benouaret, D. Benslimane, and A. Hadjali, “On the use of fuzzy dominance for computing service skyline based on qos,” in *2011 IEEE International Conference on Web Services (ICWS)*. IEEE, 2011, pp. 540–547.

[6] Z. Zheng, Y. Zhang, and M. R. Lyu, “Cloudrank: A qos-driven component ranking framework for cloud computing,” in *2010 29th IEEE Symposium on Reliable Distributed Systems (SRDS)*. IEEE, 2010, pp. 184–193.

[7] Z. Zheng, T. C. Zhou, M. R. Lyu, and I. King, “Ftcloud: A component ranking framework for fault-tolerant cloud applications,” in *2010 IEEE 21st International Symposium on Software Reliability Engineering (ISSRE)*. IEEE, 2010, pp. 398–407.

[8] S. Wang, Z. Zheng, Q. Sun, H. Zou, and F. Yang, “Cloud model for service selection,” in *2011 IEEE Conference on Computer Communications Workshops (INFOCOM WKSHPS)*. IEEE, 2011, pp. 666–671.

[9] S. Borzsony, D. Kossmann, and K. Stocker, “The skyline operator,” in *Proceedings of 17th International Conference on Data Engineering*. IEEE, 2001, pp. 421–430.

[10] K.-L. Tan, P.-K. Eng, B. C. Ooi *et al.*, “Efficient progressive skyline computation,” in *International Conference on Very Large Data Bases (VLDB)*, vol. 1, 2001, pp. 301–310.

[11] R. E. Moore, R. B. Kearfott, and M. J. Cloud, *Introduction to interval analysis*. Siam, 2009.

[12] Q. Zhang, Z. Fan, and D. Pan, “A ranking approach for interval numbers in uncertain multiple attribute decision making problems,” *Systems Engineering-Theory & Practice*, vol. 5, 1999.

[13] Z. Zheng and M. R. Lyu, “Collaborative reliability prediction of service-oriented systems,” in *Proceedings of the 32nd ACM/IEEE International Conference on Software Engineering-Volume 1*. ACM, 2010, pp. 35–44.

[14] L. Zeng, B. Benatallah, M. Dumas, J. Kalagnanam, and Q. Z. Sheng, “Quality driven web services composition,” in *Proceedings of the 12th international conference on World Wide Web (WWW)*. ACM, 2003, pp. 411–421.

[15] Y. Liu, A. H. Ngu, and L. Z. Zeng, “Qos computation and policing in dynamic web service selection,” in *Proceedings of the 13th international World Wide Web conference on Alternate track papers & posters*. ACM, 2004, pp. 66–73.

[16] L. Zeng, B. Benatallah, A. H. Ngu, M. Dumas, J. Kalagnanam, and H. Chang, “Qos-aware middleware for web services composition,” *IEEE Transactions on Software Engineering*, vol. 30, no. 5, pp. 311–327, 2004.

[17] S.-L. Liu, Y.-X. Liu, F. Zhang, G.-F. Tang, and N. Jing, “Dynamic web services selection algorithm with qos global optimal in web services composition.” *Ruan Jian Xue Bao(Journal of Software)*, vol. 18, no. 3, pp. 646–656, 2007.

[18] J. Wu, L. Chen, S.-G. Deng, Y. Li, and L. Kuang, “Qos-skyline based dynamic service selection,” *Jisuanji Xuebao(Chinese Journal of Computers)*, vol. 33, no. 11, pp. 2136–2146, 2010.