A SLA VIOLATION DEGREE-AWARE CLOUD SERVICE EVALUATION APPROACH BASED ON HISTORICAL RECORDS

Lianyong Qi¹,²,³, Wanchun Dou¹,², Xuyun Zhang⁴, Yuming Zhou¹,²
1 Department of Computer Science and Technology, Nanjing University, China
2 State Key Laboratory for Novel Software Technology, Nanjing University, China
3 School of Information Science and Engineering, Qufu Normal University, China
4 Machine Learning Research Group, NICTA, Melbourne, Australia
{lianyongqi@gmail.com}

Abstract
Cloud computing has provided a flexible and cost-effective resource provision manner to the public. From the perspective of cloud users, it is necessary to evaluate the cloud service’s quality based on the service’s historical execution records. However, the traditional evaluation approaches often assume that cloud service’s historical quality is a fixed quality value (i.e., quality point), while neglects the long-term running characteristic of cloud service and the unpredictability of network environment, as well as the resulted fluctuant service quality (i.e., quality curve); this may decrease the completeness of cloud service evaluation. Besides, the traditional SLA (Service Level Agreement) contract manner only considers whether the agreed SLA is violated, while neglects the SLA violation degree, which may be unfair to cloud service providers. In views of the above two challenges, a SLA violation degree-aware cloud service evaluation approach \( Vio\_degree\_eva \) is put forward in this paper. \( Vio\_degree\_eva \) not only considers two kinds of historical records (i.e., quality point and quality curve), but also includes SLA violation degree in cloud service evaluation, so as to make the evaluation more reasonable. Finally, through a set of experiments, we validate the feasibility of our proposal.

Keywords: cloud service, evaluation, historical record, SLA violation degree, quality curve

1. INTRODUCTION

Different from the traditional computing patterns, cloud computing technology has provided the public a completely new computing resources provision manner. With the increasing number of cloud services published by cloud providers (e.g., IBM, Amazon, Microsoft), users can easily deploy their various business applications, in a flexible and cost-effective way [1].

Today, many cloud providers have published their respective cloud services with same or similar functionality. For example, Amazon S3, Google Drive and Windows Azure all provide flexible data storage service. In this situation, from the perspective of cloud users, it is necessary to evaluate the quality of functional-equivalent cloud services so that a quality-optimal candidate could be selected. However, due to the fake quality propagation and unstable network environment, the service quality advertised by cloud service providers is not always trustable [2]. Therefore, it becomes a necessity to evaluate the quality of a cloud service based on the service’s historical execution records. Many researchers have studied this hot academic problem and made their contributions. However, there are still some challenges in the present research work.

(1) The traditional cloud service evaluation approaches often assume that cloud service’s historical quality is a fixed quality value (i.e., quality point). This assumption does not always hold in actual cloud service execution, because a cloud service may continuously serve a cloud user for a long period (e.g., one year) [3]. While during this long period, cloud service’s running performance may fluctuate with time, because the network environment cannot always stay stable or be predicted accurately. In this situation, cloud service’s historical quality is not a fixed value (i.e., quality point), but fluctuates with time (i.e., quality curve). Unfortunately, present research work seldom considers the different forms (i.e., quality point and quality curve) of cloud service quality as well as their integration problem.

(2) Due to the unstable cloud service quality, the SLA (Service Level Agreement) contracted between cloud user and cloud provider is vulnerable to network fluctuation. For example, the response-time SLA of Chinese train-ticket-order service (www.12306.cn, whose some computing resources are rented from cloud platform Aliyun) [4] can be easily violated by a sharp increment of ticket order quantity, especially when the Chinese festival is approaching. In this situation, cloud service (e.g., Aliyun) may violate the contracted SLA and should compensate the cloud user. However, present SLA contract manner often considers the SLA violation result (i.e., whether SLA is violated) only, while neglects the SLA violation degree, which may be unfair to the cloud providers.
In view of the above two challenges, a cloud service evaluation approach named \( \text{Vio}_\text{degree}_\text{eva} \) (SLA violation degree-aware cloud service evaluation) is put forward in this paper. \( \text{Vio}_\text{degree}_\text{eva} \) not only considers two kinds of historical records (i.e., quality point and quality curve) of cloud services, but also introduces SLA violation degree into SLA contract so as to make the contract fairer to cloud providers.

The remainder of this paper is organized as follows. In Section 2, we discuss the different forms of cloud service quality and introduce a novel concept of flexible double-range SLA, based on which the motivation of this paper is demonstrated. A SLA violation degree-aware cloud service evaluation approach, i.e., \( \text{Vio}_\text{degree}_\text{eva} \) is put forward in Section 3, which considers different forms of service quality and SLA violation degree simultaneously. In Section 4, a set of experiments are deployed to validate the feasibility of our proposal, in terms of effectiveness and efficiency. Our \( \text{Vio}_\text{degree}_\text{eva} \) approach is evaluated in Section 5; and finally, in Section 6, we summarize the paper and point out our future research directions.

2. **FORMAL DEFINITIONS AND MOVITATION**

In this section, we first classify the quality forms of cloud services into two categories. Afterwards, a novel concept of double-range SLA is introduced. Finally, with cloud service’s different quality forms and different SLA forms, we demonstrate the motivation of our paper.

2.1 **QUALITY FORMS OF CLOUD SERVICES.**

Concretely, the quality forms of cloud services could be classified into the following two categories: quality point (Def.1) and quality curve (Def.2). To ease the following discussions, \( \text{CS} \) denotes a cloud service; \( \text{HR} \) denotes a historical record of \( \text{CS} \); \([0, T]\) denotes the agreed service-period of \( \text{CS} \). Besides, for simplicity, we only consider a negative quality dimension \( q \) here (positive quality dimension can be converted into negative one by multiplying \(-1\)).

**Definition 1. Quality point.** For historical record \( \text{HR} \) of cloud service \( \text{CS} \), its quality over dimension \( q \) (denoted by \( \text{HR}(q) \)) is a quality point, iff equation in (1) holds.

\[
\text{HR}(q) = a
\]  

Here, \( a \) is a fixed value calculated after comprehensive consideration of \( \text{CS} \)'s running quality during \([0, T]\). For example, after statistics, \( \text{HR}(\text{failure\_rate}) = 5\% \) during the service-period \([0 \text{ hour}, 100 \text{ hours}]\). Here, we utilize \( \text{Gen}_{\text{SLA}} \) to denote the traditional single-range SLA (e.g., \( \text{failure\_rate} \in [0, 10\%] \)) over quality dimension \( q \). Then a conclusion could be drawn that SLA is satisfied if \( \text{HR}(q) \in \text{Gen}_{\text{SLA}} \) holds; otherwise, SLA is violated.

**Definition 2. Quality curve.** For historical record \( \text{HR} \) of cloud service \( \text{CS} \), its quality over dimension \( q \) (denoted by \( \text{HR}(q) \)) is a quality curve, iff equation in (2) holds.

\[
\text{HR}(q) = f(t) \quad (0 \leq t \leq T)
\]  

Here, \( \text{HR}(q) \) is not a fixed value, but changes with service time \( t \). Next, an example is provided in Figure 1 to illustrate the quality curve of \( \text{latency} \) of cloud service \( \text{CS} \), during its service-period \([0 \text{ hour}, 100 \text{ hours}]\). As Figure 1 shows, the \( \text{latency} \) of cloud service \( \text{CS} \) is not a fixed value, but fluctuant with running time \( t \).

In this situation, we cannot determine the concrete \( \text{latency} \) value of \( \text{CS} \). Besides, the fluctuant service quality brings another challenge, i.e., the traditional SLA contract specified by a single range (e.g., \( \text{latency} \in [0\text{ms}, 200\text{ms}] \)) is not suitable to constrain the fluctuant cloud service quality very well, because the single-range SLA could be easily violated by a sudden fluctuation of cloud service quality. For example, considering the example in Figure 1, the single-range \( \text{latency} \) SLA, i.e., \([0\text{ms}, 200\text{ms}] \) is violated by a sudden but normal fluctuation at peak point \( P \).

In this situation, we cannot simply conclude that the \( \text{latency} \) SLA of cloud service \( \text{CS} \) is violated, due to the following two reasons. First, as Figure 1 shows, in most of the service-period \([0 \text{ hour}, 100 \text{ hours}]\), cloud service \( \text{CS} \) performs very well in \( \text{latency} \) and does not violate the SLA constraint (e.g., \( \text{latency} \in [0\text{ms}, 200\text{ms}] \)). Second, cloud service’s running quality is fluctuant in nature and cannot be predicted accurately before its execution. Therefore, the traditional single-range SLA cannot accommodate the fluctuant quality of cloud service very well. In view of the above considerations, in this paper, we try to introduce the traditional single-range SLA and introduce a novel concept of double-range SLA to accommodate the fluctuant service quality curve.

2.2 **DOUBLE-RANGE SLA**

**Definition 3. Double-range SLA.** A double-range SLA of cloud service, i.e., Double\_SLA could be formalized as a two-tuple in (3). Here, \( \text{Gen}_{\text{SLA}} \) denotes the traditional single-range SLA that depicts cloud user’s general quality expectation, while \( \text{Peak}_{\text{SLA}} \) is another quality constraint that limits the fluctuant quality of cloud services at the peak point (e.g., point \( P \) in Figure 1).

\[
\text{Double\_SLA} = (\text{Gen}_{\text{SLA}}, \text{Peak}_{\text{SLA}})
\]  

![Figure 1. A Quality Curve Instance of CS's Latency](image-url)
The example in Figure 2 illustrates the meanings of double-range SLA. In Figure 2, \( \text{latency} \in [0\text{ms}, 200\text{ms}] \) is the \( \text{General}_{\text{SLA}} \) promised by cloud service \( CS \) (To ease the subsequent discussions, we denote \( \text{General}_{\text{SLA}} \) with its upper bound only, i.e., 200ms), while \( \text{latency} \in [0\text{ms}, 300\text{ms}] \) is the \( \text{Peak}_{\text{SLA}} \), that cannot be violated by the peak point of \( CS \)’s running quality (Likewise, we denote \( \text{Peak}_{\text{SLA}} \) with its upper bound only, i.e., 300ms).

Next, by comparing cloud service’s fluctuant service quality (i.e., \( HR(q) = f(t) \) in (2)) and double-range SLA (i.e., \( \text{Double}_{\text{SLA}} = (\text{General}_{\text{SLA}}, \text{Peak}_{\text{SLA}}) \) in (3)), we can measure a cloud user’s satisfaction degree with cloud service \( CS \)’s historical quality over dimension \( q \). Concretely, as Figure 2 shows, when \( t \in [0, t_1] \cup [t_2, T] \), the farther \( f(t) \) is away from \( \text{General}_{\text{SLA}} \), the more satisfaction a cloud user gets; while when \( t \in [t_1, t_2] \), the farther \( f(t) \) is away from \( \text{General}_{\text{SLA}} \), the more dissatisfaction a user gets. Therefore, we can calculate user satisfaction degree with dimension \( q \)’s service quality in historical record \( HR \) of cloud service \( CS \), i.e., \( \text{Sat}_{\text{degree}} (CS, HR, q) \), with the area covered by \( HR(q) = f(t) \), \( \text{General}_{\text{SLA}} \), \( t = 0 \) and \( t = T \). More formally, \( \text{Sat}_{\text{degree}} (CS, HR, q) \) could be calculated by (4), where \( HR(q) = f(t) \) \((t \in [0, T]) \) holds.

\[
\text{Sat}_{\text{degree}} (CS, HR, q) = \int_0^T (\text{General}_{\text{SLA}} - f(t)) \, dt \quad (4)
\]

Based on the above analyses, we can determine whether a double-range SLA could be satisfied by the fluctuant quality of a cloud service, with the following definition (here, only a negative quality dimension \( q \), e.g., latency is considered for illustration purpose).

**Definition 4.** Fluctuant service quality satisfies double-range SLA. For cloud service \( CS \), its fluctuant quality over dimension \( q \), i.e., \( HR(q) = f(t) \) \((t \in [0, T]) \) satisfies the promised double-range SLA, i.e., \( \text{Double}_{\text{SLA}} = (\text{General}_{\text{SLA}}, \text{Peak}_{\text{SLA}}) \), iff the constraints in (5) and (6) are met simultaneously. Otherwise, the promised double-range SLA is violated.

\[
\begin{align*}
\text{Max} f(t) & \leq \text{Peak}_{\text{SLA}}, \,(t \in [0, T]) \quad (5) \\
\int_0^T (\text{General}_{\text{SLA}} - f(t)) \, dt & \geq 0 \quad (6)
\end{align*}
\]

Next, we explain the physical meanings of formulas (5) and (6), by the example in Figure 2. Concretely, the constraint in (5) ensures that the maximum of fluctuant service quality does not exceed the promised peak threshold \( \text{Peak}_{\text{SLA}} \); while the constraint in (6) ensures that the cloud user’s general satisfaction degree with cloud service \( CS \)’s fluctuant quality is always positive (or zero), even if \( CS \)’s fluctuant quality may violate the promised \( \text{General}_{\text{SLA}} \) sometimes (e.g., when \( t \in [t_1, t_2] \) in Figure 2).

### 2.3 Motivation

In subsection 2.1-2.2, we have introduced two forms of historical record (i.e., quality point, quality curve) and two forms of SLA (i.e., single-range SLA, double-range SLA). Next, with the above concepts, we illustrate the motivation of this paper in Figure 3. As Figure 3 shows, a cloud user wants to evaluate the quality of cloud service \( CS \), based on \( CS \)’s \( L \) historical records, i.e., \( HR_1, \ldots, HR_L \). Here, each record \( HR_i, (1 \leq i \leq L) \) consists of \( m \) (negative) quality dimensions, i.e., \( q_1, \ldots, q_m \), and records their quality values \( HR_i(q_j) \) \((1 \leq j \leq m)\) in the form of quality point or quality curve. Besides, \( HR \) also records the SLA corresponding to each \( HR_i(q_j) \) concretely, a single-range SLA, i.e., \( \text{General}_{\text{SLA}} \) is available for each quality point, while a double-range SLA, i.e., \( \text{Double}_{\text{SLA}} = (\text{General}_{\text{SLA}}, \text{Peak}_{\text{SLA}}) \) is available for each quality curve.

In this situation, it is a challenge to evaluate the quality of cloud service \( CS \), based on \( CS \)’s \( L \) historical records (with different quality forms and different SLA forms). Besides, it is another challenge to consider the SLA violation degree in service evaluation. In view of these challenges, a novel cloud service evaluation approach named \( \text{Vio}_{\text{degree}} \) is put forward in the next section.
3. A SLA VIOLATION DEGREE-AWARE CLOUD SERVICE EVALUATION APPROACH

In this section, a novel cloud service evaluation approach, i.e., Vio\_degree\_eva is introduced, which is based on cloud service CS’s L historical records with different quality forms and different SLA forms. The main idea of our proposed Vio\_degree\_eva is: first, we evaluate cloud user’s satisfaction degree with quality points; second, we evaluate cloud user’s satisfaction degree with quality curves; third, satisfaction degree integration. Concretely, the three steps of our proposed Vio\_degree\_eva approach are listed in Figure 4. To ease the following discussions, in Step 1 and Step 2, we only consider a negative quality dimension q and a historical record HR of cloud service CS; while in Step 3, all the m quality dimensions q_1,…q_m and L historical records HR_1,…HR_L of CS are considered.

**Step1: Quality point evaluation.** According to cloud service CS’s fixed quality HR_q = a and contracted single-range SLA General\_SLA, calculate cloud user’s satisfaction degree with quality point, i.e., Sat\_degree (CS, HR, q).

**Step2: Quality curve evaluation** According to cloud service CS’s fluctuant quality HR_q = f(t) (t ∈[0, T]) and contracted double-range SLA Double\_SLA, calculate cloud user’s satisfaction degree with quality curve, i.e., Sat\_degree (CS, HR, q).

**Step3: Satisfaction degree integration.** By integrating the user satisfaction degrees Sat\_degree (CS, HR_q, q_i) (1 ≤ i ≤ L, 1 ≤ j ≤ m) obtained in Step 1 and Step 2, we evaluate the quality of cloud service CS.

![Figure 4. Three Steps of Cloud Service Evaluation Approach Vio\_degree\_eva](image)

**3.1 Step 1: Quality point evaluation.**

In this step, we calculate cloud user’s satisfaction degree with dimension q’s fixed quality in the historical record HR of cloud service CS, i.e., Sat\_degree (CS, HR, q), based on CS’s fixed quality HR_q = a (i.e., quality point) and contracted single-range SLA constraint General\_SLA. Next, we introduce the concrete calculation process by considering the example in Figure 5.

In Figure 5, the relationship between cloud service CS’s quality point and corresponding SLA constraint General\_SLA (here, only a negative quality dimension q is discussed for simplicity) is divided into three categories:

(a) service quality violates General\_SLA (see Figure 5(a))

(b) service quality satisfies General\_SLA (see Figure 5(b))

(c) service quality just equals General\_SLA (see Figure 5(c))

![Figure 5. Three Relationships of Quality Point and General\_SLA](image)

In Figure 5(a), cloud service CS’s quality corresponding to historical record HR violates the contracted SLA; therefore, cloud user’s satisfaction degree is low. Here, we utilize satisfaction degree range [-1, 0] to depict this kind of SLA violation; and the farther a quality point is away from General\_SLA upper bound, the smaller satisfaction degree a cloud user will get. While in Figure 5(b), cloud service CS’s quality corresponding to historical record HR satisfies the contracted SLA; therefore, cloud user’s satisfaction degree is high. Here, we utilize satisfaction degree range (0, 1] to depict this kind of SLA satisfaction; and the farther a quality point is away from General\_SLA upper bound, the larger satisfaction degree a cloud user will get. Finally in Figure 5(c), cloud service CS’s quality corresponding to historical record HR just equals the contracted SLA; in this situation, we set the user satisfaction degree 0.

Besides, we argue that cloud user’s satisfaction degree with service quality also obeys the “Marginal Effect” in Social Psychology domain. Let’s consider the example in Figure 6. In Figure 6(a), quality points A and B violate the contracted SLA, and they are both far away from the General\_SLA upper bound (however, A is close to B); in this situation, cloud user’s satisfaction degrees with A and B should be close too (actually, both near -1). Similarly, in Figure 6(b), quality points C and D satisfy the contracted SLA, and they are both far away from the General\_SLA upper bound (however, C is close to D); in this situation, cloud user’s satisfaction degrees with C and D should be close too (actually, both near 1). Therefore, “Marginal Effect” should be considered in user satisfaction degree model.
3.2 Step2: Quality curve evaluation.

In this step, we calculate cloud user’s satisfaction degree with dimension $q$’s fluctuant quality in the historical record $HR$ of cloud service $CS$, i.e., $Sat\_degree$ $(CS, HR, q)$, based on $CS$’s fluctuant quality $HR(q) = f(t)$ $(t \in [0, T])$ (i.e., quality curve) and contracted double-range SLA constraint Double$_{SLA}$ = ($Gen_{SLA}$, Peak$_{SLA}$). Next, we introduce the concrete calculation process by considering the example in Figure 2.

As introduced in Figure 2, we can calculate user satisfaction degree with dimension $q$’s fluctuant service quality in the historical record $HR$ of cloud service $CS$, i.e., $Sat\_degree$ $(CS, HR, q)$, with the area covered by $HR(q) = f(t)$, $Gen_{SLA}$, $t = 0$ and $t = T$. Concretely, user satisfaction degree $Sat\_degree$ $(CS, HR, q)$ could be calculated by formula (4); however, according to (4), $Sat\_degree$ $(CS, HR, q) \in (-\infty, +\infty)$ holds, which cannot reflect the meaning of user satisfaction degree very well. Therefore, in this step, we improve formula (4) to make the user satisfaction degree belong to range $[-1, 1]$. Next, we introduce the detailed improvement process.

Concretely, formula (8) is recruited to replace formula (4) to calculate user satisfaction degree $Sat\_degree$ $(CS, HR, q)$. Here, if $\max(f(t)) > Peak_{SLA}$ ($t \in [0, T]$), i.e., Peak$_{SLA}$ is violated, then $Sat\_degree$ $(CS, HR, q) = -1$ and no further calculation is needed. Otherwise, Peak$_{SLA}$ is satisfied; in this situation, we further calculate $Sat\_degree$ $(CS, HR, q)$ by the second and third equations in formula (8). Concretely, if $\int_0^t (Gen_{SLA} - f(t)) \, dt < 0$ holds, then user satisfaction degree $Sat\_degree$ $(CS, HR, q)$ is equal to the ratio between $\int_0^t (Gen_{SLA} - f(t)) \, dt$ and $\int_0^t f(t) \, dt$, which belongs to range $(-1, 0)$; else if $\int_0^t (Gen_{SLA} - f(t)) \, dt \geq 0$ holds, then $Sat\_degree$ $(CS, HR, q)$ is equal to the ratio between $\int_0^t (Gen_{SLA} - f(t)) \, dt$ and $\int_0^t Gen_{SLA} \, dt$, which belongs to range $[0, 1]$. Therefore, through formula (8), we can calculate user satisfaction degree with fluctuant quality curve, i.e., $Sat\_degree$ $(CS, HR, q)$ ($\in [-1, 1]$).

$$Sat\_degree\ (CS, HR, q) = \begin{cases} -1, & \text{if } \max(f(t)) > Peak_{SLA} \\ \frac{\int_0^t (Gen_{SLA} - f(t)) \, dt}{\int_0^t f(t) \, dt}, & \text{if } \int_0^t (Gen_{SLA} - f(t)) \, dt < 0 \\ \frac{\int_0^t (Gen_{SLA} - f(t)) \, dt}{\int_0^t Gen_{SLA} \, dt}, & \text{if } \int_0^t (Gen_{SLA} - f(t)) \, dt \geq 0 \end{cases}$$

(8)
3.3 Step3: Satisfaction degree integration.

In Step1–Step2, we have obtained user satisfaction degree with a dimension \( q \)'s quality (quality point or quality curve) corresponding to historical record \( HR \) of cloud service \( CS \), i.e., \( Sat\_degree \) (\( CS \), \( HR \), \( q \)). However, a cloud service often has multiple historical records and each record contains multiple quality values over different quality dimensions. Here, we assume that cloud service \( CS \) owns \( L \) historical records \( H R_1, \ldots, H R_L \) and each historical record \( H R_i \) (\( 1 \leq i \leq L \)) contains quality information over \( m \) dimensions \( q_1, \ldots, q_m \). Next, in this step, we will introduce how to evaluate cloud service \( CS \), by integrating the multiple satisfaction degree \( Sat\_degree \) (\( CS \), \( HR \), \( q \)) (\( 1 \leq i \leq L, 1 \leq j \leq m \)) derived in Step1 and Step2.

The integration process is based on two kinds of weight, i.e., weight \( w_i \) of historical record \( H R_i \) (\( 1 \leq i \leq L \), \( \sum w_i = 1 \)) and weight \( \omega_j \) of quality dimension \( q_j \) (\( 1 \leq j \leq m \), \( \sum \omega_j = 1 \)) (here, we assume that \( w_1 \) and \( \omega_1 \) are known already, because the discussions of weight design are out of the scope of this paper). Concretely, the quality of cloud service \( CS \) could be evaluated by formula (9), where \( Quality(CS) \) denotes the evaluated quality of \( CS \).

\[
\text{Quality}(CS) = \frac{1}{L} \sum_{i=1}^{L} w_i \times (\sum_{j=1}^{m} \omega_j \times Sat\_degree(CS, HR_i, q_j)) \tag{9}
\]

According to (9), the evaluated quality of cloud service \( CS \), i.e., \( Quality(CS) \in [-1, 1] \) holds. Here, we evaluate the quality of cloud service \( CS \) by considering both \( CS \)'s historical quality and \( CS \)'s agreed SLA, because it does not make any sense to consider \( CS \)'s historical quality only without comparing its corresponding SLA constraints. Therefore, we argue that formula (9) is more reasonable for cloud service evaluation.

With the above three steps of \( Vio\_degree\_eva \), we can evaluate the quality of a cloud service, based on its multiple forms of historical records (i.e., quality point and quality curve) and multiple SLA forms (i.e., single-range SLA and double-range SLA). More formally, the pseudo code of our proposal is specified as below.

### 4. Experiments

In this section, a set of experiments are designed and deployed to validate the feasibility of our proposed cloud service evaluation approach \( Vio\_degree\_eva \).

4.1 The Dataset and Experiment Deployment

In our proposed \( Vio\_degree\_eva \) approach, both quality point (corresponding to Single-rang SLA, i.e., \( General_{SLA} \)) and quality curve (corresponding to Double-rang SLA, i.e., \( General_{SLA}, Peak_{SLA} \)) are considered. However, to the best of our knowledge, there is no present dataset that contains these two kinds of quality data. Fortunately, we can construct the experiment dataset with advanced software tools.

#### Algorithm: \( Vio\_degree\_eva \) (\( CS, HR, Q \))

**Input:** \( CS \): a cloud service

\( HR = \{HR_1, \ldots, HR_L\} \): a set of historical records of \( CS \)

\( Q = \{q_1, \ldots, q_m\} \): a set of quality dimensions of \( CS \)

**Output:** \( Quality(CS) \): evaluated quality of cloud service \( CS \)

1: \( Quality(CS) = 0 \)

2: \( for \ i = 1 \ to \ L \ do \)

3: \( \quad for \ j = 1 \ to \ m \ do \)

4: \( \quad \quad if \ HR(q_j) \ is \ a \ quality \ point \ then \)

5: \( \quad \quad \quad \quad \text{calculate} \ Sat\_degree(CS, HR_i, q) \ by \ (7) \)

6: \( \quad \quad \quad \quad \text{else if} \ HR(q_j) \ is \ a \ quality \ curve \ then \)

7: \( \quad \quad \quad \quad \quad \text{calculate} \ Sat\_degree(CS, HR_i, q) \ by \ (8) \)

8: \( \quad \quad \quad \quad \text{end if} \)

9: \( \quad \quad \quad \text{end if} \)

10: \( \quad \quad \text{Quality}(CS) = \text{Quality}(CS) + Sat\_degree(CS, HR_i, q) \)

11: \( \quad \text{end for} \)

12: \( \text{end for} \)

13: \( \text{Quality}(CS) = \text{Quality}(CS) \ / \ (L \cdot m) \)

14: \( \text{return} \ Quality(CS) \)

Concretely, the quality curve set of cloud services, i.e., \( Dataset\_quality\_curve \) could be generated with the help of performance-monitoring software \( CloudWatch \) [6]. \( CloudWatch \) is developed by \( Amazon \) and could provide monitoring service over a series of quality dimensions of cloud services. Corresponding to \( Dataset\_quality\_curve \), the double-range SLA, i.e., \( General_{SLA}, Peak_{SLA} \) could be generated randomly by considering the range of fluctuant quality (as only negative quality dimensions are considered, \( General_{SLA} \) should be smaller than \( Peak_{SLA} \) in the constructed double-range SLA). Besides, the quality point set of cloud services, i.e., \( Dataset\_quality\_point \) could be generated from \( WS\_DREAM \) [7] published by Dr. Zibin Zheng in 2011. \( WS\_DREAM \) consists of 4532 services from public sources on the web, and 142 distributed computers from Planet-Lab are employed for evaluating the real service quality performance in 64 time intervals. Corresponding to \( Dataset\_quality\_point \), the single-range SLA, i.e., \( General_{SLA} \) could be generated randomly by considering the range of quality point data. Besides, we assume weight \( w_i = 1/L \) (\( 1 \leq i \leq L \)) and \( \omega_j = 1/m \) (\( 1 \leq j \leq m \)). For a cloud user, it is difficult to accurately evaluate his/her satisfaction degree with a quality curve by (8), as the service quality fluctuates continuously; therefore, sampling technique is recruited here for quality curve, so as to achieve approximate evaluation results. Some symbols recruited in the experiments as well as their specifications could be found in Table1.
4.2 Experiment Results

In this subsection, we compare our evaluation approach $Vio_{degree\_eva}$ with another two evaluation approaches, i.e., $SD\_HCF$ [8] (only quality point is considered) and $FL\_FL$[9] (only quality curve is considered). Concretely, five evaluation profiles were tested. The parameter values recruited in experiments are listed in Table 1.

**Profile 1: Utilization rate of historical quality records**

Due to the sparse cloud service quality feedback, we should make full use of all the collected historical quality records. However, for different cloud service evaluation approaches (i.e., $SD\_HCF$, $FL\_FL$, $Vio_{degree\_eva}$), their utilization rates of historical quality records are not the same. In this profile, we design a set of experiments to compare their utilization rates. Concretely, let $L=200$ and $m=15$, then there are totally 200$\times$15=3000 historical quality records. We assume $RATIO_{point\_curve}=1/2$, which means that there are 1000 quality points and 2000 quality curves. Next, we evaluate the 3000 historical quality records by three approaches respectively and calculate the corresponding user satisfaction degree distribution.

The concrete experiment results are shown in Figure 8, where two observations are available. (1) All the 3000 historical quality records are utilized in our proposed $Vio_{degree\_eva}$, while only 1000 quality points are considered in $SD\_HCF$ and only 2000 quality curves are considered in $FL\_FL$. Namely, both $SD\_HCF$ and $FL\_FL$ waste some precious historical execution information of cloud services, so their utilization rates are smaller than $Vio_{degree\_eva}$. (2) In $SD\_HCF$, the user satisfaction degrees with most (actually 54%) quality points are equal to 0, which means that a SLA violation occurs and the violation degree is not considered in $SD\_HCF$. Likewise, in $FL\_FL$, the user satisfaction degrees with most (actually 62%) quality curves are equal to 0, which means that a SLA (either $Peak_{sla}$ or $General_{sla}$) violation occurs and the violation degree is omitted. Besides, in $FL\_FL$, there is no quality curve whose satisfaction degree is in range (0, 0.5) (i.e., two red cylinders are absent from Figure 8), which seems unreasonable to some extent. While in our proposed $Vio_{degree\_eva}$, the user satisfaction degree is relatively “average” (i.e., utilization rate of historical quality records is high), and negative satisfaction degree is presented to indicate both SLA violation result and SLA violation degree.

**Profile 2: Time cost with respect to $L$**

In this profile, we compare the time cost of three approaches with respect to the number of historical records of cloud service CS, i.e., $L$. Concretely, the number of quality dimensions, i.e., $m=15$ holds; $RATIO_{point\_curve}=1$ holds in $Vio_{degree\_eva}$; for each quality curve, the sampling point number $K=10$ holds in both $FL\_FL$ and $Vio_{degree\_eva}$; $L$ is varied from 200 to 1000.

The experiment results are shown in Figure 9. It can be seen from Figure 9 that the time costs of three approaches all grow approximately linearly with the rise of $L$. Furthermore, $SD\_HCF$ outperforms the other two approaches in time cost because it only considers the simple quality points without sampling cost; in contrast, $FL\_FL$ does not perform very well in time cost, since sampling is

<table>
<thead>
<tr>
<th>Symbol</th>
<th>Value</th>
<th>Specification</th>
</tr>
</thead>
<tbody>
<tr>
<td>$L$</td>
<td>200,400,600,800,1000</td>
<td>Number of historical records of cloud service CS</td>
</tr>
<tr>
<td>$m$</td>
<td>3,6,9,12,15</td>
<td>Number of quality dimensions of cloud service CS</td>
</tr>
<tr>
<td>$K$</td>
<td>10,20,30,40,50</td>
<td>Number of sampling points for each quality curve</td>
</tr>
<tr>
<td>$RATIO_{point_curve}$</td>
<td>0.5, 1</td>
<td>Ratio between quality point number and quality curve number</td>
</tr>
</tbody>
</table>

Table 1. Values and specifications of experiment symbols
needed for each quality curve considered in FL-FL. Also, as Figure 9 shows, our proposed Vio_degree_eva realizes a medium-sized time cost between SD-HCF and FL-FL, as both quality points and quality curves are considered in Vio_degree_eva.

**Profile3: Time cost with respect to m**

In this profile, we compare the time cost of three approaches with respect to the number of quality dimensions of cloud service CS, i.e., m. Concretely, the number of historical records, i.e., \( L = 1000 \) holds; \( RATIO_{point/curve} = 1 \) holds in Vio_degree_eva; for each quality curve, the sampling point number \( K = 10 \) holds in both FL-FL and Vio_degree_eva; \( m \) is varied from 3 to 15.

The experiment results are shown in Figure 10. It can be seen from Figure 10 that the time costs of three approaches all grow approximately linearly with the rise of \( m \), because all the \( m \) quality dimensions need to be considered in the three approaches. Furthermore, similar with Figure 9, SD-HCF outperforms the other two approaches in time cost because it only considers the quality points without sampling cost; while FL-FL and Vio_degree_eva achieve similar but poor performance in time cost, as sampling is needed for each quality curve considered in these two approaches.

**Profile4: Time cost with respect to K**

In this profile, we compare the time cost of three approaches with respect to the number of sampling points, i.e., \( K \). Concretely, the number of historical records, i.e., \( L = 1000 \), the number of quality dimensions, i.e., \( m = 15 \); \( RATIO_{point/curve} = 1 \) holds in Vio_degree_eva; the sampling point number \( K \) is varied from 10 to 50 in both FL-FL and Vio_degree_eva.

The experiment results are shown in Figure 11. It can be seen from Figure 11 that the time cost of SD-HCF is small and stays relatively stable, this is because only quality points (without sampling, so we can regard \( K = 1 \) holds constantly) are considered in SD-HCF. While the time costs of FL-FL and Vio_degree_eva both increase approximately linearly with the growth of \( K \), this is because all the \( K \) sampling points should be considered in these two approaches. However, as Figure 11 shows, our proposed Vio_degree_eva performs better than FL-FL, because only partial historical quality records (i.e., quality curve) in Vio_degree_eva need sampling.

**Profile5: User satisfaction degree with respect to K**

In both FL-FL and Vio_degree_eva, sampling technique is recruited to simplify the calculation of user satisfaction degree with quality curve (see formula (8)). Concretely, each continuous quality curve is divided into \( K \) discrete quality points. Next, we test the correlation between \( K \) and derived user satisfaction degree (by FL-FL and Vio_degree_eva). Concretely, we consider a quality curve \( HR(q) \) corresponding to quality dimension \( q \) and historical record \( HR \) of cloud service CS. Then \( HR(q) \) is divided into \( K \) (\( K=10, 20, 30, 40, 50 \)) discrete quality points, through which the user satisfaction degree with \( HR(q) \) could be calculated by (8). We compare the user satisfaction degree and the comparison results are shown in Figure 12.
In Figure 12, three observations are available. (1) For both FL-FL and \( \text{Vio}_\text{degree}_{\text{eva}} \), user satisfaction degrees fluctuate slightly with the rise of \( K \) in most cases (i.e., when \( K \) is varied from 20 to 50). And user satisfaction degree stays approximately stable when \( K \) is large (i.e., \( K = 40, 50 \)), this is because when \( K \) is large enough, the \( K \) sampled quality points can approach the original continuous quality curve \( HR(q) \) infinitly. (2) When \( K \) is small (i.e., \( K = 10 \)), a distortion of satisfaction degree is observed; this is because when \( K \) is small, the \( K \) sampled quality points cannot represent the original continuous quality curve \( HR(q) \) very well. (3) User satisfaction degree calculated by FL-FL is larger than that by \( \text{Vio}_\text{degree}_{\text{eva}} \), this is because an additional smoothing operation (i.e., \( \text{Sat}_\text{degree} = (\text{Sat}_\text{degree} + 1) / 2 \)) is adopted in FL-FL.

5. Evaluation

In this section, we first analyze the time complexity of \( \text{Vio}_\text{degree}_{\text{eva}} \) introduced in Section 3, to evaluate the efficiency of our proposal. Afterwards, a comparison with related work is presented, which is followed by discussions regarding the paper limitations and our future work.

5.1 Time Complexity Analysis

Suppose that cloud service CS owns \( L \) historical records, each record corresponds to \( m \) quality dimensions, and the sampling point number is \( K \) for each quality curve. Then there are \( L^m \) quality points or quality curves totally. Next, we discuss the time complexity of \( \text{Vio}_\text{degree}_{\text{eva}} \) by considering the following two extreme situations.

(1) There are \( L^m \) quality points.

If \( HR(q) \) is a quality point, then according to Step1, user satisfaction degree \( \text{Sat}_\text{degree} \) \( (\text{CS}, \ HR, \ q) \) could be calculated by formula (7), whose time complexity is \( O(1) \). As there are totally \( L^m \) quality points, the time complexity of satisfaction degree integration in Step3 is \( O(L^m) \). With the above analyses, time complexity of \( \text{Vio}_\text{degree}_{\text{eva}} \) is \( O(L^m) \).

(2) There are \( L^m \) quality curves.

If \( HR(q) \) is a quality curve, then according to Step2, user satisfaction degree \( \text{Sat}_\text{degree} \) \( (\text{CS}, \ HR, \ q) \) could be calculated by formula (8), whose time complexity is \( O(K) \). As there are totally \( L^m \) quality curves, the time complexity of satisfaction degree integration in Step3 is \( O(L^m) \). With the above analyses, time complexity of \( \text{Vio}_\text{degree}_{\text{eva}} \) is \( O(L^m*K) \).

With the above analyses, a conclusion could be drawn that the time complexity of \( \text{Vio}_\text{degree}_{\text{eva}} \) is \( O(L^{m*K}) \) (In case (1), only fixed quality points are present, so we can regard the sampling point number \( K = 1 \) approximately). Namely, the time complexity of \( \text{Vio}_\text{degree}_{\text{eva}} \) is affected by historical record number \( L \), quality dimension number \( m \) and sampling point number \( K \), in an approximately linear manner, which has also been validated by the experiment results in Section 4.

5.2 Related Works and Comparison Analyses

The dynamic and unstable execution environment of cloud services makes it a great challenge to accurately evaluate the service quality. A monitoring mechanism is provided in [10] to dynamically collect the running service quality level and compare it with the contracted SLA between users and services, so as to determine whether SLA constraints are violated. However, the SLA monitoring/verification work cannot be enacted and executed by cloud users or cloud providers, because cloud users lose direct control of their business execution in cloud, while cloud providers may not be trustable from the perspective of cloud users. In view of this, a third-party cloud service broker (CSB) is proposed in [11] to complete the tasks of service quality monitoring and SLA verification. Similarly, a SLA verification framework is put forward in [12], which leverages a third-party auditor (TPA) to realize the effective detection of SLA violations.

While the above works mainly focus on the SLA monitoring or verification, few of them consider the cloud service evaluation based on the monitored historical quality data. In view of this, works [11][13][14] recruit the historical records of services for service composition, service evaluation and service trust calculation, respectively. Furthermore, in [15], weight is assigned to each historical record for better service evaluation and selection decisions. However, the above works only consider the historical records in the form of quality points, while neglect the fluctuant characteristic of cloud service quality during the long service-period.

In [3], the authors have observed the long-term business relationship between cloud users and cloud service providers, as well as the fluctuant service quality during the long service-period. However, the paper only focuses on the cloud service composition (different from our paper), and does not discuss the flexible SLA as well as the SLA violation degree. In our previous work [9], a flexible SLA concept is put forward to accommodate the fluctuant cloud service quality; and furthermore, a cloud service could be evaluated by comparing its fluctuant service quality and flexible SLA. However, further discussion about SLA violation degree is absent from [9]; besides, [9] only discusses the quality curve of cloud service, without considering the quality point as well as their integration problem.

In view of the above shortcomings, a SLA violation degree-aware cloud service evaluation approach \( \text{Vio}_\text{degree}_{\text{eva}} \) is put forward in this paper. \( \text{Vio}_\text{degree}_{\text{eva}} \) not only considers two kinds of historical records (i.e., quality point and quality curve), but also introduces SLA violation degree in cloud service evaluation. Finally, through a set of experiments, we validate the feasibility of our proposal, in terms of evaluation effectiveness and efficiency.
5.3 Further Discussions
In our proposed Vio_degree_eva approach, we assume that the cloud service’s historical quality records are known already. While in fact, due to privacy concern and poor incentive, it is really a challenging task to collect the historical quality records of a cloud service. Besides, most historical quality data is published by cloud providers, which is not always trustable from the cloud user perspective. Therefore, in the future, we will consider these shortcomings and study a trusted and privacy-aware cloud service evaluation approach.

6. CONCLUSIONS
It is a challenge to accurately evaluate the quality of cloud services, due to the long service-period and unstable quality performance. In view of this challenge, we classify the historical quality records of cloud services into two categories, i.e., quality point and quality curve; furthermore, two forms of SLA (i.e., single-range SLA corresponding to quality point and double-range SLA corresponding to quality curve) are brought forth; finally, we calculate the SLA violation degree and develop a SLA violation degree-aware cloud service evaluation approach Vio_degree_eva. Through a set of experiments, we validate the feasibility of Vio_degree_eva in terms of effectiveness and efficiency. In the future, we will discuss the privacy issues in cloud service evaluation, and study a trusted and privacy-aware evaluation approach.

7. ACKNOWLEDGMENT
We would like to thank Dr. Zibin Zheng for the dataset WS-DREAM recruited partially in our experiments. (Partial work of this paper has been accepted by ICWS 2014).

8. REFERENCES

Authors
Lianyong Qi received his PhD degree in Department of Computer Science and Technology from Nanjing University, China, in 2011. In 2010, he visited the Department of Information and Communication Technology, Swinburne University of Technology. Now, he is a lecturer of the School of Information Science and Engineering, Qufu Normal University, China. Also, he is doing his postdoctoral research in the Postdoctoral Research Centre for Software Engineering, Nanjing University, China. He is now chairing a NSFC project and has published nearly 20 research papers in international journals (e.g., TCC, CCPE, JCSS, Computing) and international conferences (e.g., ICWS, HPCC, APSCC). His research interests include big data, cloud computing and services computing.

Wanchun Dou received his PhD degree in Mechanical and Electronic Engineering from Nanjing University of Science and Technology, China, in 2001. From Apr. 2001 to Dec. 2002, he did his postdoctoral research in the Department of Computer Science and Technology, Nanjing University, China. Now, he is a full professor of the State Key Laboratory for Novel Software Technology, Nanjing University, China. From Apr. 2005 to Jun. 2005 and from Nov. 2008 to Feb. 2009, he respectively visited the Department of Computer Science and Engineering, Hong Kong University of Science and Technology, as a
visiting scholar. Up to now, he has chaired three NSFC projects and published more than 60 research papers in international journals (e.g., TPDS, TETC, FGCS, JCSS, JWSR, CCPE, IJHPCA) and international conferences (e.g., ICWS, HPCC, ICPADS). His research interests include workflow, cloud computing and service computing.

**Xuyun Zhang** is a researcher at the Machine Learning Group, Australia's Information Communications Technology Research Centre of Excellence (NICTA), working with mentors Prof. Chris Leckie and Prof. Ramamohanarao (Rao) Kotagiri from The University of Melbourne. He studied his PhD degree in the Faculty of Engineering and Information Technology (FEIT) at the University of Technology Sydney (UTS), supervised by A/Prof. Jinjun Chen. He also worked at the Australian national scientific body Commonwealth Scientific and Industrial Research Organization (CSIRO) as a co-supervised PhD student, with Principal Research Scientist Surya Nepal. His research interest includes Cloud Computing, Big Data, scalable Machine learning and Data Mining, Large Graph Processing, Privacy and Security, Web Service Composition, QoS.

**Yuming Zhou** received his PhD degree from Southeast University, China, in 2003. Now, he is a full professor of the State Key Laboratory for Novel Software Technology, Nanjing University, China. From Feb. 2005 to Feb. 2008, he visited Hong Kong Polytechnic University, as a Research Associate and Research Fellow. He has published some research papers in international journals (e.g., TSE, TOSEM, JSS, JCSS, SPE, IST, JCST) and conferences (e.g., ICSM, SCAM, APSEC, SEKE, SAC, Metrics). His research interests include software metrics, software evolution, and program understanding and analysis.