

A Trust Vector Approach to Service-Oriented Applications

Lei Li

Department of Computing
Macquarie University
Sydney, NSW 2109, Australia
leili@ics.mq.edu.au

Yan Wang

Department of Computing
Macquarie University
Sydney, NSW 2109, Australia
yanwang@ics.mq.edu.au

Abstract

Trust is a critical issue in e-commerce and e-service environments. In some applications (such as eBay), the trust management mechanisms have been introduced to provide valuable information to buyers prior to placing orders and making payments. Meanwhile, the trust issue is also actively studied in the research community. However, most studies and applications focus on approaches that result in a single trust value to represent the trust level of sellers or service providers. Such a simple trust evaluation method may not be able to depict the trust history exactly and may leave misleading information to service customers. In this paper, we present a novel approach of trust vector consisting of three values to reflect the trust level with more indications.

1 Introduction

In e-commerce or e-service environments, the reputation-based trust status of a seller or a service provider is a very important concern from the view point of a buyer or a service customer. When there are a few sellers or service providers providing the same product or service, the buyer or service customer would like to order from the seller or service provider with the best transaction reputation. This is particularly important when the buyer or service customer has to select from unknown sellers or service providers.

In general, in a trust management mechanism enabled system, buyers or service customers can provide feedback and ratings after transactions. Then the trust management system can calculate the trust value based on collected ratings reflecting the quality of recent transactions. The trust value can be provided to buyers or service customers, by publishing it on web or responding to their requests.

Peer-to-Peer (P2P) networks can be used for information-sharing systems [2], where the trust issue is also actively studied (e.g. [12]). In a P2P system, it

is quite natural for a client peer to doubt if a serving peer can provide the complete file prior to any download action, which may be quite time-consuming and network bandwidth-consuming. Different from some trust management systems in e-commerce environments, in the P2P trust system, a requesting peer needs to enquire the trust data of a serving peer (target peer) from other peers which may have transacted with the serving peer [4, 6, 13]. The computation of the trust status of the serving peer from the collected trust ratings is then performed by the requesting peer, not a central management server, because of the decentralized architecture of P2P system.

In existing trust management studies, the *final trust level* is computed to reflect the *general* or *global* trust status of every service provider. For example, if the *final trust level* value is in the range of $[0, 1]$, 0.95 indicates a very good trust level for a service provider. Usually the final trust value is computed by taking the service trust ratings in a recent period into account, i.e. assigning higher weights to trust values of recent transactions or services as the latest transaction rating is the most important in trust evaluation [11][15].

Such a *final trust level* value can reflect the service reputation accumulated in a certain period. However, a single *final trust level* value cannot reflect the real trust status very well. For example, the single-value approach cannot reflect (i) the service trust trend of changes and (ii) the service performance consistency level. *Service trust trend* indicates that the service trust will become better or worse in forthcoming transactions. This is an important indication to tell buyers or service customers *to what extent a seller or a service provider is trustworthy for new transactions or services*. *Service performance consistency level* can indicate whether the service quality is being maintained at the level reflected by the final trust value, which makes sense no matter whether the trust value is low or high. For example, if the trust level of a service provider is *good* and its trust level is “*consistent*”, it indicates that the service provider has maintained the *good* level for a certain period.

Thus a good trust management system requires more comprehensive trust evaluation approaches providing more (precise) trust information that indicates not only the global trust level, but also the trust prediction *relevant to forthcoming transactions*. To serve for this purpose, in this paper, we propose a *service trust vector* consisting of a set of values, such as *final trust level*, *service trust trend* and *service performance consistency level*, which is applicable to e-commerce or e-service environments. We will also conduct empirical studies to study the properties of our proposed approaches.

This paper is organized as follows. In Section 2, we review the trust management approaches of eBay and some existing studies. Section 3 discusses *service trust trend* and *service performance consistency level* evaluations in our service trust vector. Some empirical studies are presented in Section 4 for further illustrating the properties of our model. Finally Section 5 concludes our work.

2 Background

2.1 Trust Management at eBay

eBay [1] is a typical Customer-to-Customer (C2C) web site, whose trust management mechanism is one of the earliest systems in applications.

At eBay, after each transaction, the buyer can give feedback to the system according to the service quality of the seller. The feedback (or rating) is stored by eBay (a centralized management architecture), which can be “positive”, “neutral” or “negative”. eBay calculates the feedback score $S = P - N$, where P is the number of positive feedback left by members (customers) and N is the number of negative feedback from members. Then S value can be displayed on the web pages. In addition, $R = \frac{P-N}{P+N}$ (e.g., $R = 99.1\%$) is called positive feedback rate, based on which a seller can be awarded as a “Power Seller” if $R \geq 98\%$ (98% is the threshold).

eBay also improves its trust service and provides rating data in 12 months listed in a table, which is divided by recent *1 month*, *6 months* and *12 months*. Thus, eBay provides some simple mechanisms of trust management and trust calculation and leaves some raw data to buyers for self-calculation. Obviously, it leaves much room for improvement.

2.2 Trust Management in Other Environments

In [15], a trust evaluation approach is proposed for e-commerce applications which is based on the trust values of transactions in a recent period, rather than all of them. In this method, recent ratings are more important in the trust evaluation. In [8], fuzzy logic is applied to trust evaluation, which divides sellers or service providers into multiple

classes of reputation ranks (e.g. a 5-star seller, or a 4-star seller).

Trust issue is also actively pursued in Peer-to-Peer (P2P) information-sharing network (e.g. [2]). P2P trust evaluation relies on a polling algorithm (e.g. [3]), a binary rating system for calculating the global trust value of a given peer [4][13], or a voting reputation system (e.g. [6]) that calculates the final trust value combining the values returned by responding peers and the requesting peer’s experience with the given peer.

As pointed in [14], binary ratings work pretty well for file-sharing systems where a file is either the definitive correct version or wrong, but cannot accurately model richer services such as web services or e-commerce, where a Boolean may not adequately represent a client peer’s experience of the quality of service (QoS) with other serving peers, such as the quality of products the serving peer sends and the expected delivery time [14]. In most later studies on trust evaluation (e.g. [11], [12], [14]), a numeral rating system is adopted, where, for example, the rating is a value in the range of $[0, 1]$. Such a rating system is more suitable for complex applications, such as e-commerce or service-oriented applications.

In the literature, trust issue also receives much attention in service-oriented computing research. In [5], Lin *et al* propose a method of reputation-based trust evaluation in service-oriented environments based on a proposed architecture consisting of distributed trust management brokers. In [9], Vu *et al* present a model to evaluate and rank the trust and reputation of QoS-based services. In [10], an event-driven and rule-based trust management for service-oriented application is proposed, where a formula based approach is adopted for incremental trust computation. Moreover, the approach is adaptable to applications by incorporating rule management. Then, the computed result can be taken as a global trust value reflecting the accumulated trust level, which is not particularly relevant to forthcoming transactions.

2.3 Some Issues in Trust Evaluation

However, in most existing studies, a single-value service trust evaluation method is adopted, which is simple but cannot reflect the trust status precisely. In general, the *final trust level (FTL)* value is provided as an indication for the global trust level for a seller or service provider, which may be presumably taken as a prediction of trustworthiness for forthcoming transactions. Namely, if a seller or service provider has a very good *FTL* value, a forthcoming transaction will be very trustworthy; otherwise, it is not trustworthy. Obviously, this may be misleading. More information should be provided to reflect the global trust more precisely, which can be valuable to predict the trust level of a new transaction well. This information includes some aspects as

follows.

1. The *service trust trend (STT)* value should be computed which is useful to predict the future trust level of service quality.

STT value shows a general trend of changes in the service quality in the near future, which is important when we choose a seller or service provider with serious caution. Namely, given a set of sellers or service providers with the same *FTL* value, the one which is *becoming better* is more desirable to a buyer or service customer. Obviously, with a single *FTL* value only, it is impossible to illustrate this property.

2. The *service performance consistency level (SPCL)* value should be calculated to indicate whether the real service level is consistent in a certain period.

Given a set of sellers or service providers with the same *FTL* and *STT* values, the one which has higher consistency level is more desirable to a buyer or service customer. Especially, the *SPCL* value equals 1 if and only if the service quality is kept the same in a certain period. Meanwhile, the *SPCL* value equals 0 if and only if the service quality level changes most during a certain period.

In this paper, we propose a novel service trust vector containing *FTL*, *STT* and *SPCL* values. In addition, we also propose evaluation methods to compute the *STT* and *SPCL* values.

3 Service Trust Vector and Its Evaluation

As discussed above, a single *FTL* value cannot indicate the trust status of a seller or service provider well. Instead, *STT* and *SPCL* values should be computed in addition to the *FTL* value so as to provide more precise indication of the trust level to a buyer or service customer. This is the big concern of a buyer or a service customer particularly when the seller or service provider is unknown.

From another view, the *service trust vector* can be used to rebuild the *trust data curve* with a set of values, which depict the trust history well. Therefore, in addition to a single *FTL* value, which is used in most existing trust evaluations, *STT* and *SPCL* values are important as they, together with the *FTL* value, can describe the trust data curve better.

3.1 Final Trust Level (FTL) Evaluation

The calculation of *FTL* follows a common principle as follows, which appears in a number of studies [11][15].

Principle 1: The trust value is computed by taking the service quality in a recent period into account, and assigning higher weights to quality values of later services.

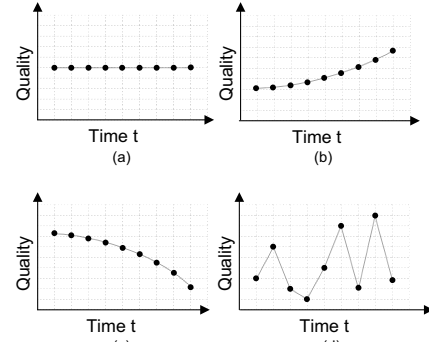


Figure 1. Several *STT* cases

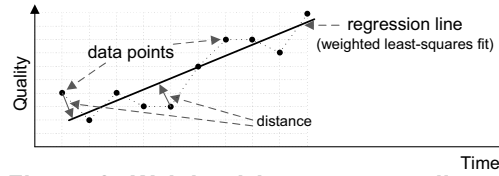


Figure 2. Weighted least squares linear regression

Based on *Principle 1*, the *FTL* value can be calculated as:

$$T_{FTL} = \frac{\sum_{i=1}^n w_i q_i}{\sum_{i=1}^n w_i}, \quad (1)$$

where w_i is the weight for service quality q_i at time i ($i = 1 \dots n$), which can be calculated as follows:

$$w_i = \alpha^{n-i}, \quad 0 < \alpha \leq 1. \quad (2)$$

Other weighting functions can also be used, such as [12].

3.2 Service Trust Trend (STT) Evaluation

STT aims to illustrate the trust of service quality trend of changes in a given period. Some typical cases of *STT* are depicted in Fig. 1, which are “coherent”, “up-going”, “dropping” and “uncertain” in sequence.

In order to evaluate *STT*, we refer to the idea of least squares linear regression [7]. Namely, the general trend of the data points of service quality can be represented by a *regression line*, which follows the general pattern of a set of data points, with no attempt to pass through each of them.

In addition, regarding *Principle 1*, we will introduce a *weighted least squares linear regression* method to evaluate *STT*, which is illustrated in Fig. 2. This method is used to obtain the best fit straight line from a set of given data points. This best fit straight line is characterized by the sum of weighted squared residuals having its least value, where a residual is the distance from a data points to the regression line (refer to Fig. 2). Once obtaining the regression line, its slope can be taken as our *STT* value.

Now we introduce the *STT* evaluation method.

Let $(t_1, q_1), (t_2, q_2), \dots, (t_n, q_n)$ denote the given data points in a certain period, where q_i ($q_i \in [0, 1]$) is the service quality value at time t_i ($t_i < t_{i+1}$). Then the *regression*

line can be represented as

$$q = a_0 + a_1 t, \quad (3)$$

where a_0 and a_1 are constants to be determined, and a_1 represents the *STT* value. As the distance from point (t_i, q_i) to the regression line is

$$d_i = \frac{|q_i - a_0 - a_1 t_i|}{\sqrt{1 + a_1^2}}, \quad (4)$$

the *sum of squares of the distance* can be defined as follows.

Definition 1: Based on the method of weighted least squares, the *sum of squares of the distance* can be calculated as follows:

$$S = \sum_{i=1}^n w_i^2 d_i^2 = \sum_{i=1}^n \frac{w_i^2 (q_i - a_0 - a_1 t_i)^2}{1 + a_1^2}. \quad (5)$$

Now our task is to minimize the *sum of squares of the distance* S with respect to the parameters a_0 and a_1 , with the method of undetermined coefficients.

Since function S is continuous and differentiable, based on Fermat's theorem in real analysis, the minimization point of S makes the first derivative of function S be zero, and the second derivative positive, which could be easily proved since a_1 is very small here. For this, we differentiate S with respect to a_0 and a_1 , and set the results to zero, which gives

$$\frac{\partial S}{\partial a_0} = -2 \sum_{i=1}^n \frac{w_i^2 (q_i - a_0 - a_1 t_i)}{1 + a_1^2} = 0,$$

and

$$\frac{\partial S}{\partial a_1} = -2 \sum_{i=1}^n \frac{w_i^2 (q_i - a_0 - a_1 t_i)(a_1 q_i - a_0 a_1 + t_i)}{(1 + a_1^2)^2} = 0,$$

i.e.

$$\sum_{i=1}^n w_i^2 q_i - a_0 \sum_{i=1}^n w_i^2 - a_1 \sum_{i=1}^n w_i^2 t_i = 0, \quad (6)$$

and

$$\begin{aligned} & \sum_{i=1}^n w_i^2 t_i q_i + a_0 a_1^2 \sum_{i=1}^n w_i^2 t_i + a_0^2 a_1 \sum_{i=1}^n w_i^2 \\ & - 2a_0 a_1 \sum_{i=1}^n w_i^2 q_i - a_1^2 \sum_{i=1}^n w_i^2 t_i q_i \\ & + a_1 \sum_{i=1}^n w_i^2 q_i^2 - a_0 \sum_{i=1}^n w_i^2 t_i - a_1 \sum_{i=1}^n w_i^2 t_i^2 = 0. \end{aligned} \quad (7)$$

Eqs. (6) and (7) can be solved for the unknown a_0 and a_1 , by substituting a_0 from Eq. (6) into Eq. (7) to obtain

$$a_1^2 + \frac{S_{wq2} S_w - S_{wq}^2 + S_{wt}^2 - S_{wt2} S_w}{S_{wq} S_{wt} - \sum_{i=1}^n w_i^2 t_i q_i S_w} a_1 - 1 = 0, \quad (8)$$

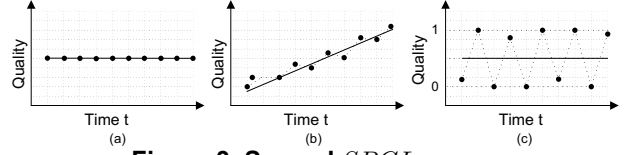


Figure 3. Several *SPCL* cases

where $S_w = \sum_{i=1}^n w_i^2$, $S_{wt} = \sum_{i=1}^n w_i^2 t_i$, $S_{wq} = \sum_{i=1}^n w_i^2 q_i$, $S_{wt2} = \sum_{i=1}^n w_i^2 t_i^2$, and $S_{wq2} = \sum_{i=1}^n w_i^2 q_i^2$. Obviously, it is easy to obtain a very small real solution of Eq. (8) about a_1 . Due to the space constraint, we ignore the detailed step for calculating a_1 .

Furthermore, by substituting the solution of a_1 back to Eq. (6), we can obtain

$$a_0 = \frac{\sum_{i=1}^n w_i^2 q_i - a_1 \sum_{i=1}^n w_i^2 t_i}{\sum_{i=1}^n w_i^2}. \quad (9)$$

Thus, based on the method of weighted least squares linear regression, we can obtain the *STT* value $T_{STT} = a_1$, which is determined from Eq. (8). However, in order to determine the four cases of *STT*, another factor should be taken into account. It will be introduced in Section 3.3.

3.3 Service Performance Consistency Level (*SPCL*) Evaluation

The *SPCL* value offers the indication of the consistency level of service quality in a certain period. Some typical *SPCL* cases are depicted in Fig. 3. In sequence, they are “absolutely consistent”, “relatively consistent”, and “inconsistent”.

Prior to presenting the detailed *SPCL* evaluation method, we firstly introduce some definitions.

Definition 2: The *predicted value of regression line* is

$$q_{tr}(t_i) = a_0 + a_1 t_i, \quad (10)$$

where a_0 and a_1 are decided by Eqs. (9) and (8) respectively.

Definition 3: According to *Principle 1*, the *weighted average distance* is

$$q_{tdis} = \frac{\sum_{i=1}^n w_i |q_i - q_{tr}(t_i)|}{\sqrt{1 + a_1^2} \sum_{i=1}^n w_i} \quad (11)$$

for n quality values q_i in period t .

Now let us introduce a principle about the *SPCL* evaluation as follows.

Principle 2: The *SPCL* value T_{SPCL} is a monotonically decreasing function of the weighted mean distance q_{tdis} .

According to *Principle 2*, we have the following *SPCL* evaluation formula.

Definition 4: The *SPCL* value is

$$T_{SPCL} = 1 - 2q_{tdis} = 1 - 2 \frac{\sum_{i=1}^n w_i |q_i - q_{tr}(t_i)|}{\sqrt{1 + a_1^2} \sum_{i=1}^n w_i}. \quad (12)$$

From the above definition, the properties of $SPCL$ can be analyzed as follows:

Property 1.1: If $\epsilon_1 < T_{SPCL} \leq 1$ ($0 \ll \epsilon_1 < 1$ is the threshold), $SPCL$ is *absolutely consistent* (refer to Fig. 3(a)), i.e. the service performance remains unchanged.

Property 1.2: If $\epsilon_2 < T_{SPCL} < \epsilon_1$ ($0 < \epsilon_2 \ll 1$ is the threshold), $SPCL$ is *relatively consistent* (refer to Fig. 3(b)), i.e. the service performance is consistent in a certain level.

Property 1.3: If $T_{SPCL} < \epsilon_2$, $SPCL$ is *inconsistent* (refer to Fig. 3(c)), i.e. the service performance is not consistent.

Now, with T_{STT} and T_{SPCL} , the four cases of STT can be determined as follows.

Property 2.1: If $T_{SPCL} > \epsilon_2$ and $|T_{STT}| < \epsilon_3$ ($0 < \epsilon_3 \ll 1$ is the threshold), STT is *coherent* (refer to Fig. 1(a)), i.e. the service quality remains at the same level.

Property 2.2: If $T_{SPCL} > \epsilon_2$ and $T_{STT} > \epsilon_3$, STT is *up-going* (refer to Fig. 1(b)), i.e. the service quality is becoming better.

Property 2.3: If $T_{SPCL} > \epsilon_2$ and $T_{STT} < -\epsilon_3$, STT is *dropping* (refer to Fig. 1(c)), i.e. the service quality is turning worse.

Property 2.4: If $T_{SPCL} < \epsilon_2$, STT is *uncertain* (refer to Fig. 1(d)), i.e. the service quality is not reliable.

3.4 Service Quality Trust Vector

Based on the above discussion, we have the following definition about the service quality trust vector.

Definition 5: The *service quality trust vector* \bar{T} consists of the F_{TL} value $T_{F_{TL}}$, the STT value T_{STT} , and the $SPCL$ value T_{SPCL}

$$\bar{T} = \langle T_{F_{TL}}, T_{STT}, T_{SPCL} \rangle, \quad (13)$$

where $T_{F_{TL}}$ is defined in Eq. (1), T_{STT} is decided by Eq. (8), and T_{SPCL} is defined in Eq. (12).

Moreover, with trust vectors, all service providers form a partial order set. Given two service providers P_i , P_j with service quality trust vectors $\bar{T}_i = \langle T_{F_{TL}_i}, T_{STT}_i, T_{SPCL}_i \rangle$, and $\bar{T}_j = \langle T_{F_{TL}_j}, T_{STT}_j, T_{SPCL}_j \rangle$ respectively, they are comparable in the following cases:

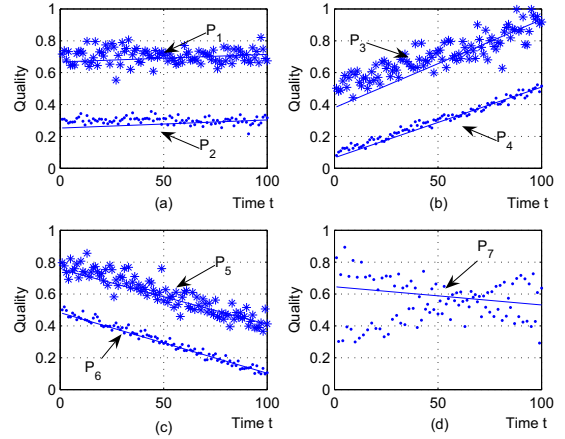


Figure 4. Study 1

1. If $T_{STT}_i = T_{STT}_j$, $T_{SPCL}_i = T_{SPCL}_j$, and $T_{F_{TL}_i} < T_{F_{TL}_j}$, P_j is more preferable. We denote it as $P_j > P_i$ or $P_i < P_j$.
2. If $T_{F_{TL}_i} = T_{F_{TL}_j}$, $T_{SPCL}_i = T_{SPCL}_j$, and $T_{STT}_i < T_{STT}_j$, P_j is more preferable. This is denoted as $P_j > P_i$ or $P_i < P_j$.
3. If $T_{F_{TL}_i} = T_{F_{TL}_j}$, $T_{STT}_i = T_{STT}_j$, and $T_{SPCL}_i < T_{SPCL}_j$, P_j is more preferable. We denote it as $P_j > P_i$ or $P_i < P_j$.

4 Empirical Studies

In this section, we illustrate the results of conducted simulations to study the proposed service trust vector approach, and explain why the service trust vector is necessary and important.

In these studies, we set $\epsilon_1 = 0.94$ and $\epsilon_2 = 0.85$ which are the thresholds to determine *absolutely consistent*, *relatively consistent*, *inconsistent SPCL* and *uncertain STT* (refer to Property 1.1-1.3 & 2.4), and set $\epsilon_3 = 0.0006$ which is the threshold for determining *coherent*, *up-going* and *dropping STT* together with ϵ_2 (refer to Property 2.1-2.3). Meanwhile, we set the parameter $\alpha = 0.95$ in the weighting function Eq. (2).

4.1 Study 1

In this study, we focus on four cases about STT as follows. The computed results are listed in Table 1, and the best fit straight line for each service provider in this study is also plotted in Fig. 4.

Case 1: In this case, as plotted in Fig. 4(a), there are two service providers P_1 and P_2 . According to Table 1, as $T_{SPCL}_1 > \epsilon_2$, $|T_{STT}_1| < \epsilon_3$, $T_{SPCL}_2 > \epsilon_2$

and $|T_{STT_2}| < \epsilon_3$, based on *Property 2.1*, they both have *coherent STT*. In contrast, according to Table 1, as $\epsilon_2 < T_{SPCL_1} < \epsilon_1$, based on *Property 1.2*, P_1 has *relatively consistent SPCL*. Meanwhile, as $T_{SPCL_2} > \epsilon_1$, according to *Property 1.1*, P_2 has *absolutely consistent SPCL*.

Case 2: As plotted in Fig. 4(b), there are two service providers P_3 and P_4 in this case. Based on Table 1, as $T_{SPCL_3} > \epsilon_2$, $T_{STT_3} > \epsilon_3$, $T_{SPCL_4} > \epsilon_2$ and $T_{STT_4} > \epsilon_3$, according to *Property 2.2*, they both have *up-going STT*. In contrast, based on Table 1, as $\epsilon_2 < T_{SPCL_3} < \epsilon_1$, according to *Property 1.2*, P_3 has *relatively consistent SPCL*. Meanwhile, as $T_{SPCL_4} > \epsilon_1$, based on *Property 1.1*, P_4 has *absolutely consistent SPCL*.

Case 3: In this case, as plotted in Fig. 4(c), there are two service providers P_5 and P_6 . According to Table 1, as $T_{SPCL_5} > \epsilon_2$, $T_{STT_5} < -\epsilon_3$, $T_{SPCL_6} > \epsilon_2$ and $T_{STT_6} < -\epsilon_3$, based on *Property 2.3*, they both have *dropping STT*. In contrast, according to Table 1, as $\epsilon_2 < T_{SPCL_5} < \epsilon_1$, based on *Property 1.2*, P_5 has *relatively consistent SPCL*. Meanwhile, as $T_{SPCL_6} > \epsilon_1$, according to *Property 1.1*, P_6 has *absolutely consistent SPCL*.

Case 4: As plotted in Fig. 4(d), there is one service provider P_7 in this case. According to Table 1, as $T_{SPCL_7} < \epsilon_2$, it has *uncertain STT* based on *Property 2.4*, and *inconsistent SPCL* based on *Property 1.3*.

	T_{FTL}	T_{STT}	T_{SPCL}
P_1	0.7069	0.0005	0.9286
P_2	0.2968	0.0005	0.9573
P_3	0.8364	0.0055	0.8948
P_4	0.4288	0.0044	0.9693
P_5	0.4759	-0.0035	0.9301
P_6	0.1737	-0.0038	0.9710
P_7	0.5472	-0.0011	0.7874

Table 1. Study 1 results

4.2 Study 2

In this study, we introduce two cases. In *Case 1*, we conduct a study to illustrate why *SPCL* is necessary and important. In *Case 2*, we aim to explain by examples the introduction of weight function (2) is important. The computed results are listed in Table 2 and Table 3, and the best fit straight line for each service provider in the study is also plotted in Fig. 5.

Case 1: In this case, as plotted in Fig. 5(a)(b), there are two service providers P_8 and P_9 . According to Table 2, as $T_{FTL_8} \approx T_{FTL_9}$, $T_{STT_8} = T_{STT_9}$, and

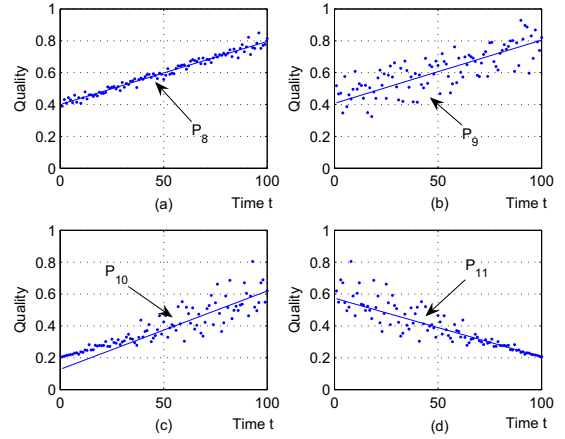


Figure 5. Study 2

	T_{FTL}	T_{STT}	T_{SPCL}
P_8	0.7232	0.0040	0.9591
P_9	0.7305	0.0040	0.8621
P_{10}	0.5299	0.0049	0.8661
P_{11}	0.2734	-0.0037	0.9731

Table 2. Study 2 results with w_i

$T_{SPCL_8} > T_{SPCL_9}$, we can conclude that $P_8 > P_9$. This case indicates that the our *SPCL* model is useful for depicting the trust history.

Case 2: As plotted in Fig. 5(c)(d), there are two service providers P_{10} and P_{11} in this case, which have the following property: the service quality q_i at time t_i of P_{10} in Fig. 5(c) is the same as the one at time $100 - t_i$ of P_{11} in Fig. 5(d).

Without weight function (2), from Table 3, we obtain $T_{FTL_{10}} = T_{FTL_{11}}$, $T_{SPCL_{10}} = T_{SPCL_{11}}$ and $T_{STT_{10}} = -T_{STT_{11}}$. We can conclude that, without weight function (2), P_{10} and P_{11} have the same *FTL*, *SPCL* and the absolute value of *STT*.

However, after introducing the weight function, we have much different values, which are listed in Table 2. As plotted in Fig. 5(c)(d), the latest q_i of P_{10} is larger than the one of P_{11} , which is proven by $T_{FTL_{10}} = 0.5299 > 0.3971$ (refer to $T_{FTL_{10}}$ and $T_{FTL_{11}}$ in Table 3) $> T_{FTL_{11}} = 0.2734$ in Table 2 and Table 3. Similarly, $T_{STT_{10}} = 0.0049 > 0.0041 > |T_{STT_{11}}| = 0.0037$ proves that P_{10} is becoming better and P_{11} is turning worse. Meanwhile, as $T_{SPCL_{10}} = 0.8661 < 0.9126 < T_{SPCL_{11}} = 0.9731$, it proves that P_{10} is turning less consistent and P_{11} is becoming more consistent.

So we can see that with weight function (2) the service quality history can be described more precisely.

	T_{FTL}	T_{STT}	T_{SPCL}
P_8	0.6003	0.0040	0.9687
P_9	0.6133	0.0035	0.8635
P_{10}	0.3971	0.0041	0.9126
P_{11}	0.3971	-0.0041	0.9126

Table 3. Study 2 results without w_i

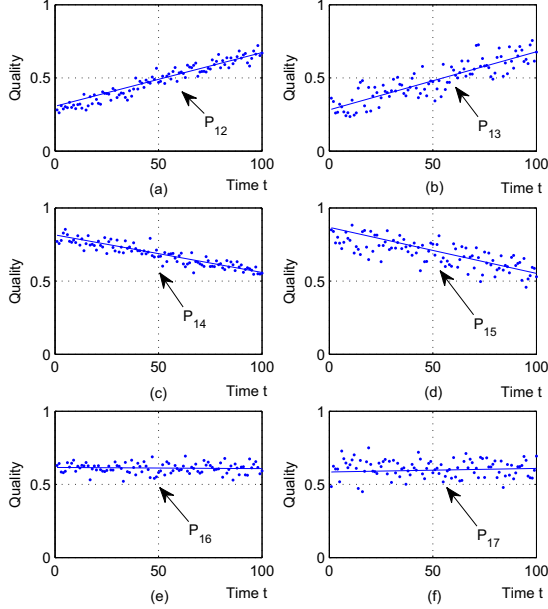


Figure 6. Study 3

4.3 Study 3

In this study, we conduct a study with six cases to illustrate why the trust vector is necessary and important. The computed results are listed in Table 4, and the best fit straight line for each service provider in this study is also plotted in Fig. 6.

	T_{FTL}	T_{STT}	T_{SPCL}
P_{12}	0.6055	0.0037	0.9472
P_{13}	0.6059	0.0040	0.8966
P_{14}	0.6053	-0.0026	0.9602
P_{15}	0.6076	-0.0032	0.9015
P_{16}	0.6092	-0.0001	0.9417
P_{17}	0.6065	0.0003	0.9039

Table 4. Study 3 results

In each case, as plotted in Fig. 6, there is one service provider. According to Table 4, all six cases have almost the same FTL , but different STT or $SPCL$. Based on *Properties 1.1-1.3 & 2.1-2.3*, we can determine the typical cases of STT and $SPCL$ in Table 5. Meanwhile, we can conclude that $P_{12} > P_{13}$, $P_{12} > P_{14}$, $P_{13} > P_{15}$, $P_{14} > P_{15}$, $P_{16} > P_{14}$, $P_{16} > P_{17}$, and $P_{17} > P_{15}$. Namely, with

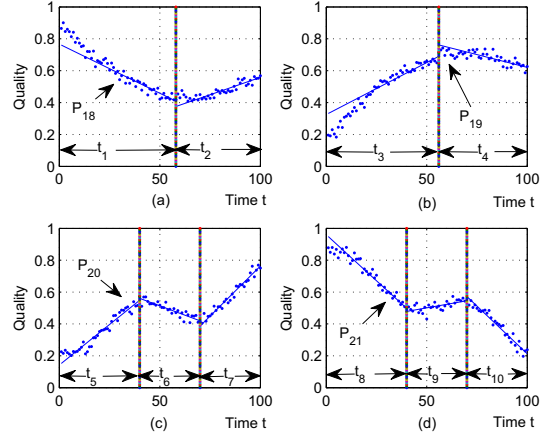


Figure 7. Study 4

solo FTL , it is not likely to depict the trust history exactly.

	T_{STT}	T_{SPCL}
P_{12}	<i>up-going</i>	<i>absolutely consistent</i>
P_{13}	<i>up-going</i>	<i>relatively consistent</i>
P_{14}	<i>dropping</i>	<i>absolutely consistent</i>
P_{15}	<i>dropping</i>	<i>relatively consistent</i>
P_{16}	<i>coherent</i>	<i>absolutely consistent</i>
P_{17}	<i>coherent</i>	<i>relatively consistent</i>

Table 5. Study 3 results about typical cases

Therefore, in this study, we can notice that the trust vector including T_{FTL} , T_{STT} and T_{SPCL} can describe the history of trust data more precisely, than the solo FTL .

4.4 Study 4

Until now we consider only one time period t in all cases. In order to evaluate the trust history of service quality better, multiple time periods may have to be introduced, where each period corresponds to the trust trend with high T_{SPCL} value. There are four cases in this study. The computed results are listed in Table 6. The best fit straight line for each case with one service provider is also plotted in Fig. 7.

According to Table 6, based on *Properties 2.1-2.3*, we can determine the typical cases of STT in Table 7. Obviously, it is easy to see that each period has *absolutely consistent SPCL* (i.e. high T_{SPCL} value).

From the above results, we can see that in case of dynamic trust trend, proper multiple time periods should be determined so as to describe the transaction history better.

5 Conclusions

In this paper, we propose a trust vector approach to service-oriented applications, which includes *final trust level (FTL)*, *service trust trend (STT)*, and *service performance consistency level (SPCL)*. Corresponding evalua-

		T_{FTL}	T_{STT}	T_{SPCL}
P_{18}	t_1	0.5107	-0.0062	0.9590
	t_2	0.4998	0.0044	0.9663
P_{19}	t_3	0.5842	0.0065	0.9534
	t_4	0.6646	-0.0032	0.9615
P_{20}	t_5	0.4159	0.0102	0.9545
	t_6	0.4717	-0.0047	0.9683
P_{21}	t_7	0.6246	0.0126	0.9655
	t_8	0.6477	-0.0116	0.9622
	t_9	0.5206	0.0022	0.9649
	t_{10}	0.3449	-0.0120	0.9567

Table 6. Study 4 results

		T_{STT}
P_{18}	$t_1 = [0, 58]$	<i>dropping</i>
	$t_2 = [58, 100]$	<i>up-going</i>
P_{19}	$t_3 = [0, 56]$	<i>up-going</i>
	$t_4 = [56, 100]$	<i>dropping</i>
P_{20}	$t_5 = [0, 40]$	<i>up-going</i>
	$t_6 = [40, 70]$	<i>dropping</i>
P_{21}	$t_7 = [70, 100]$	<i>up-going</i>
	$t_8 = [0, 40]$	<i>dropping</i>
	$t_9 = [40, 70]$	<i>up-going</i>
	$t_{10} = [70, 100]$	<i>dropping</i>

Table 7. Study 4 results about typical STT

tion methods for STT and $SPCL$ are also proposed. From our analytical and empirical studies, we can see that the proposed approach can depict trust history exactly. It offers more information to service customers for their decision-making in the selection of trustworthy service providers.

For future work, as we have discussed in Section 4.4, a multiple time period detection algorithm should be studied for the case with various trust trends, with which an accurate trust vector can be determined for each period.

References

- [1] *eBay*. <http://www.eBay.com/>.
- [2] *GNutella*. <http://www.gnutella.com/>.
- [3] E. Damiani, S. D. C. di Vimercati, S. Paraboschi, P. Samarati, and F. Violante. A reputation based approach for choosing reliable resources in peertopeer networks. In *Proceedings of ACM CCS'02*, pages 207–216, Washington DC, USA, November 2002.
- [4] S. D. Kamvar, M. T. Schlosser, and H. Garcia-Molina. The eigentrust algorithm for reputation management in P2P networks. In *Proceedings of the 12th International WWW Conference*, Budapest, Hungary, May 2003.
- [5] K.-J. Lin, H. Lu, T. Yu, and C. en Tai. A reputation and trust management broker framework for web applications. In *Proceedings of The 2005 IEEE International Conference on e-Technology, e-Commerce and e-Service (EEE'05)*, pages 262–269, March 2005.
- [6] S. Marti and H. Garcia-Molina. Limited reputation sharing in P2P systems. In *Proceedings of ACM EC'04*, pages 91–101, New York, USA, May 2004.
- [7] S. Rao. *Applied Numerical Methods for Engineers and Scientists*. Prentice Hall, 2002.
- [8] S. Song, K. Hwang, R. Zhou, and Y.-K. Kwok. Trusted p2p transactions with fuzzy reputation aggregation. *IEEE Internet Computing*, 9(6):24–34, 2005.
- [9] L.-H. Vu, M. Hauswirth, and K. Aberer. Qos-based service selection and ranking with trust and reputation management. In *Proceedings of 13th International Conference on Cooperative Information Systems (CoopIS 2005)*, Oct 31-Nov 4 2005.
- [10] Y. Wang, K.-J. Lin, D. S. Wong, and V. Varadharajan. The design of a rule-based and event-driven trust management framework. In *The IEEE International Conference on e-Business Engineering (ICEBE 2007)*, pages 97–104, Hong Kong, October 2007.
- [11] Y. Wang and V. Varadharajan. Interaction trust evaluation in decentralized environments. In K. Bauknecht, M. Bichler, and B. Pröll, editors, *Proceedings of 5th International Conference on Electronic Commerce and Web Technologies (EC-Web'04)*, volume LNCS 3182, Springer-Verlag, pages 144–153, Zaragoza, Spain, August-September 2004.
- [12] Y. Wang and V. Varadharajan. *Trust²*: Developing trust in peer-to-peer environments. In *Proceedings of 2005 IEEE International Conference on Services Computing (SCC 2005)*, pages 24–31, Orlando, Florida, USA, July 2005.
- [13] L. Xiong and L. Liu. PeerTrust: Supporting reputation-based trust for peer-to-peer electronic communities. *IEEE Trans. on Knowledge and Data Engineering*, 16(7):843–857, 2004.
- [14] B. Yu, M. P. Singh, and K. Sycara. Developing trust in large-scale peer-to-peer systems. In *Proceedings of 2004 IEEE First Symposium on Multi-Agent Security and Survivability*, pages 1–10, August 2004.
- [15] G. Zacharia and P. Maes. Trust management through reputation mechanisms. *Applied Artificial Intelligence Journal*, 9:881–908, 2000.