2010

PBTrust: A Priority-Based Trust model for service selection in general service-oriented environments

Xing Su
University of Wollongong, xs702@uowmail.edu.au

Minjie Zhang
University of Wollongong, minjie@uow.edu.au

Yi Mu
University of Wollongong, ymu@uow.edu.au

Kwang Mong Sim
Gwanju Institute of Science and Technology

Publication Details

Su, X., Zhang, M., Mu, Y. & Sim, K. Mong. (2010). PBTrust: A Priority-Based Trust model for service selection in general service-oriented environments. IEEE/IFIP 8th International Conference on Embedded and Ubiquitous Computing, EUC 2010 (pp. 841-848). USA: IEEE Computer society.

PBTrust: A Priority-Based Trust model for service selection in general service-oriented environments

Abstract
How to choose the best service provider (agent), which a service consumer can trust in terms of the quality and success rate of the service in an open and dynamic environment, is a challenging problem in many service-oriented applications such as Internet-based grid systems, e-trading systems, as well as service-oriented computing systems. This paper presents a Priority-Based Trust (PBTrust) model for service selection in general service-oriented environments. The PBTrust is robust and novel from several perspectives. (1) The reputation of a service provider is derived from referees who are third parties and had interactions with the provider in a rich context format, including attributes of the service, the priority distribution on attributes and a rating value for each attribute from a third party; (2) The concept of 'Similarity' is introduced to measure the difference in terms of distributions of priorities on attributes between requested service and a refereed service in order to precisely predict the performance of a potential provider on the requested service; (3) The concept of general performance of a service provider on a service in history is also introduced to improve the success rate on the requested service. The experimental results can prove that PBtrust has a better performance than that of the CR model in a service-oriented environment.

Keywords
PBTrust, Priority, Based, Trust, model, for, service, selection, general, service, oriented, environments

Disciplines
Physical Sciences and Mathematics

Publication Details
Su, X., Zhang, M., Mu, Y. & Sim, K. Mong. (2010). PBTrust: A Priority-Based Trust model for service selection in general service-oriented environments. IEEE/IFIP 8th International Conference on Embedded and Ubiquitous Computing, EUC 2010 (pp. 841-848). USA: IEEE Computer society.

2010 IEEE/IFIP International Conference on Embedded and Ubiquitous Computing

PBTrust: A Priority-Based Trust Model for Service Selection in General Service-Oriented Environments

Xing Su, Minjie Zhang, Yi Mu
School of Computer Science and Software Engineering
University of Wollongong,
Wollongong, NSW 2522, Australia
Email: {xs702,minjie,ymu}@uow.edu.au

Kwang Mong Sim
Gwanju Institute of Science and Technology
South Korea
Email: prof_sim_2002@yahoo.com

***Abstract*—How to choose the best service provider (agent), which a service consumer can trust in terms of the quality and success rate of the service in an open and dynamic environment, is a challenging problem in many service-oriented applications such as Internet-based grid systems, e-trading systems, as well as service-oriented computing systems. This paper presents a Priority-Based Trust (PBTrust) model for service selection in general service-oriented environments. The PBTrust is robust and novel from several perspectives. (1) The reputation of a service provider is derived from referees who are third parties and had interactions with the provider in a rich context format, including attributes of the service, the priority distribution on attributes and a rating value for each attribute from a third party; (2) The concept of 'Similarity' is introduced to measure the difference in terms of distributions of priorities on attributes between requested service and a refereed service in order to precisely predict the performance of a potential provider on the requested service; (3) The concept of general performance of a service provider on a service in history is also introduced to improve the success rate on the requested service. The experimental results can prove that PBtrust has a better performance than that of the CR model in a service-oriented environment.**

I. Introduction

Over the last few years, Multi-Agent Systems (MASs) have come to be perceived as a crucial technology, not only for effectively exploiting the increasing availability of diverse, heterogeneous, and distributed on-line information resources, but also as a framework for building large, complex, and service-oriented systems such as Internet-based grid systems [1], e-market places [2], [3], pervasive computing systems [4], as well as peer-to-peer systems [5]. Most of service-oriented systems are open and dynamic with heterogeneous/distributed resources owned by self/semi-self-interested agents and agents can also join and leave systems freely. An agent in such a system can be either a provider or a consumer for a service. The decision making for a service consumer to select a suitable provider with requested resources depends only on the incomplete information of the partner agents, the local view about the surrounding environment, as well as the experience of previous interaction with potential providers from third parties. Considering the interaction between self-interested agents, the 'trust' has played a central role in these interactions [6], [7], so how to evaluate trust value for a potential partner to complete a requested service becomes a very important and challenging issue in both MAS research and agent-based service-oriented systems.

Nowadays, different models, mechanisms and approaches have been developed to help service consumers evaluate the trust values of potential service providers. SPORAS is one of widely accepted models [8]. In SPORAS, each consumer can rank the provider after interacting with it. The ranking results will be kept in a central database. Upon receiving a new ranking for an agent, SPORAS will update the global reputation for that agent. Since SPORAS employs a learning function for updating process, the reputation value of a provider can reflect its general performance. SPORAS is a simple model and can be implemented in different applications. However, the central management mechanism in SPORAS greatly limits its applications in open and dynamic environments, particularly is based not only on the reputation of a single agent but also the reputation of the group which the agent in service-oriented environments. REGRET is also a famous model proposed by Sabater and Sierra in 2001 [9]. The contribution of REGRET is to introduce the concept of neighborhood reputation and group reputation. However, REGRET model does not show how to build a social network with agents, which limits its application in open environments such as service oriented applications. Sen and Sajja proposed a service provider selection mechanism based on a probabilistic calculation of trust values given by a number of agents including providers and consumers [10]. When an interaction between a consumer and a provider happens, agents surrounding the interaction can make observations on the performance of the provider. A provider's performance on a service is calculated based on the observation from surrounding agents and updated accordingly using reinforcement learning rules.The mechanism has a good performance with a low number of liar agents but becomes worse when the surrounding agents include more liars.

978-0-7695-4322-2/10 $26.00 © 2010 IEEE
DOI 10.1109/EUC.2010.129

In order to meet the current challenges raised from service-oriented environments, researchers in MASs and service-oriented applications have proposed different trust and reputation models in recent years through the considerations of reputation, experience, and other features of open environments [11], [12], [13] [14]. The most famous model is the Certified Reputation (CR) model proposed by Huynh, Jennings and Shadbolt [15]. In the CR, an agent reputation is derived from references of third parties about its previous performance. Agents can collect and present such references actively to service consumers in order to get the trust from their potential partners. Since the CR model allows agents to evaluate trust themselves without using a central controller, the CR model can be adapted to work in a wide range of open and dynamic environments. However, there are still several limitations in the CR model. Firstly, in the CR model, a service is represented by a single item and the evaluation of a service given by a referee is represented by a single value. In the real world, a single value is hard or impossible to express many complex contexts related to a service, in terms of a provider's performance on different attributes such as speed, cost, quality, and constrains on a particular service, and the priorities on individual attributes. Secondly, although a service provider can offer several references to represent its previous performance, but these references can only reflect the best/better behavior/s of the service provider, without the indication of its general performance for all services in the provider's history.

This paper proposes a Priority-Based Trust model (PBTrust) to overcome the limitations of the CR model. PBTrust model consists of four modules which are the Request Module, Reply Module, Priority-based Trust Calculation Module and Evaluation Module. The features of PBTrust model are to produce the reputation for a potential service provider from 4 perspectives, which are the provider's experience on the service, the similarity of priorities distributions on attributes between the referenced service and the requested service, the suitability of the potential provider for the requested service and the time effectiveness of ratings from third parties. The merits of PBTrust model include: (1) The service is represented by a matrix including attributes and priority distribution on attributes to reflect the rich context of a general service. The evaluation result for a service from a referee is expressed by a n-tuple, and a value of each element in the n-tuple represents the rating from the referee for the corresponding attribute in the service. (2) The concept of 'Similarity' is introduced to measure the difference in terms of distributions of priorities on attributes between requested service and a refereed service in order to precisely predict the performance of a potential provider on the requested service; (3) we extend the CR model by not only deriving the reputation from third party references, but also considering the general performance of a provider on a particular service in history. The experimental results show that our model has significant improvement in comparison with the CR model especially when contexts of services are more complicated.

The rest of paper is organized as follows. Section 2 is the detail problem description and definitions. Section 3 is the principle of PBTrust. The detail introduction of PBTrust model is presented in Section 4. Section 5 is the experiment and comparison. The paper is concluded and the future work is outlined in Section 6.

II. Problem Description and Definitions

In general, a service can be described by a number of attributes such as price, time, quality, etc. For different requests, the priority on different attributes of the same service can be different. In order to deal with the relationships between attributes and their corresponding priorities, we make a service description in a formal way.

Suppose there are n attributes used to describe a requested service and each attribute is in a requested priority as the condition to complete the service. The service can be represented by n attributes and their corresponding priorities, respectively.

Definition 1: A *service description* is represented by $SDes$ and is defined in the following matrix format.

$$SDes = \begin{pmatrix} A_1 & A_2 & A_3 & ... & A_n \\ W_1 & W_2 & W_3 & ... & W_n \end{pmatrix} \quad (1)$$

where A_i indicates the i^{th} attribute and W_i is the priority value of the i^{th} attribute A_i and $\sum_{i=1}^{n} W_i = 1$.

Definition 2: Let $Ratings$ represent the rating of performance of a provider on a service, given by a referee and $Ratings$ is defined as a n-tuple, $Ratings =< R_1, R_2, ..., R_n >$, where R_i indicates the rating value of i^{th} attribute of the service (recall Definition 1). Here the range of R_i is [0, 100], where 0 and 100 represent the worst and best performance for i_th attribute.

In the CR model, the references of a provider can only reflect its several good performances so it is hard for a consumer to have a general view about whether the provider has a consistent performance on the requested service. In order to solve this problem, the concept of *service experience* of a provider on a certain service is introduced in this model and is defined below.

Definition 3: The *service experience* of a provider on a service is defined as a 2-tuple, $Exp =< SRate, SNum >$, where $SRate$ indicates the success rate of the provider on this service and $SNum$ indicates the total number of success times on the same service.

Definition 4: A *service request* is defined as a 4-tuple, $SR =< CID, SDes, RN, Sthreshold >$, where CID is the service consumer's ID, $SDes$ indicates the service,

which is a 2 by n matrix representing the requested attributes and their priorities, (recall Definition 1), RN is the number of references that CID requests and $0 < RN$, and $Sthreshold$ is the threshold of the success rate for a provider to qualify for providing the service.

Definition 5: A *reference* Rf is defined as a 4-tuple, $Rf =< RfID, SDes, Ratings, T >$, where $RfID$ is the ID of the referee, $SDes$, (recall Definition 1), is the service description conducted by the provider for the referee, $Ratings$ indicates the performance for the service, given by $RfID$ for each attribute of the service, (recall Definition 2), and T is the time in the completion of the service.

Definition 6: A *service reply* is defined as a 3-tuple, $SR =< SPID, RfSet, Exp >$, where $SPID$ is the ID of the service provider, $RfSet$ is the set of references, including several best references, provided by different referees to the provider's service before and the number of references can be determined by consumers, and Exp is the service experience, (recall Definition 3), indicating the provider's general performance on this service.

III. Principle and Basic Modules in PBTrust

PBTrust consists of four modules which are the Request Module, Reply Module, Priority-based Trust Calculation Module and Evaluation Module. In this section, the purpose of each module is described briefly. The principle and each function in Priority-based Trust Calculation Module will be introduced in detail in Section 4.

A. Request Module

The objective of the Request Module is to create a *service request* based on the request from a consumer.

For example, Consumer C in an e-market place requests a service described by 3 attributes, i.e. cost, speed, and quality with corresponding priorities for each attribute as (0.3, 0.5, 0.2), respectively. C requests 2 references and the requested success rate for a potential provider on the service in history should be at lest 70%. Based on this service request, the Request Module will generate a service description as follows using the format of Definition 1.

$$SDes = \begin{pmatrix} Cost & Speed & Quality \\ 0.3 & 0.5 & 0.2 \end{pmatrix}$$

Then, a service request SR will be produced based on the service description and requirements of the Consumer C in the format defined by Definition 4.

$$SR =< C,\ SDes,\ 2,\ 0.7 >$$

The above example will be used for the explanation of rest modules.

B. Reply Module

When a potential provider P can offer the service based on the requirement from Consumer C, P will provide the following information: the provider ID, two reference reports, as well as service experience on the service before including success rate and total success times.

Suppose that P received 3 reference reports for its previous performance on the same service from different consumers representing by a set $\{Rf_1, Rf_2, Rf_3\}$, and each element in the set is in the format defined by Definition 5. P will pick up two best reference reports to represent its previous performance on the service, say Rf_2 and Rf_3.

Suppose that the success rate of P on the service is 70% and total success times to complete the service is 35.

The reply information from P responding to the request from C is as follows, (recall Definition 6).

$$SR =<\text{P},\ \{Rf_2,\ Rf_3\},\ (0.7,\ 35) >$$

If more than one service providers have the requested service and also have the intention to provide the service, this module will generate more than one replies.

C. Priority-based Trust Calculation Module

This Module is the core of PBTrust model. The main purpose of this module is to calculate the trust values of potential providers based on reference reports from third parties, service experience of providers, the time weights of references, and the similarities between the requested service description and the one from reference reports in terms of different priorities on same attributes. These trust values will help a consumer to select the best provider which the consumer can trust to complete the service. The final trust value for each potential provider is produced from several calculation results in four perspectives, which are the provider's experience on the service, the similarity of priorities distributions on attributes between the referenced service and the requested service, the suitability of the potential provider for the requested service and the time effectiveness of ratings from third parties. The detail design and calculations for each perspective will be introduced in Section 4.

D. Evaluation Module

This module includes two components. One is to generate a reference report from a consumer for a provider based on the performance of a completed service and the other is to update the record of service experience of a provider when a new reference is available for the provider.

1) Reference report generation: We use the same example as in Request Module and Reply Module to demonstrate how to generate a reference report in this module. After completing the requested service, Consumer C evaluates the performance of Provider P on the service. The evaluation

result is represented in a reference report, (recall Definition 5) shown as follows.

$$Rf =< C, SDes, < 60, 40, 90 >, 12/7/2008 > .$$

The above reference report shows the evaluation result from Consumer C on the service $SDes$, completed on 12 July 2008. From Consumer C's rating, we can see that C was not happy with the speed of the service i.e. the second attribute of $SDes$, was very happy with the quality of the service, i.e. the 3rd attribute of $SDes$, and was satisfied with the cost of the service i.e. the 1st attribute of $SDes$.

2) Service experience updating: The service experience updating is based on the consumer's judgement on the newly completed service from the provider. The result of judgement can be in two values, either 'success' or 'fail', relating to the performance of the provider.

The service experience Exp includes two elements $SNum$ and $SRate$, (recall Definition 3). $SNum$ and $SRate$ can be updated by the following two formulas.

$$SNum = \begin{cases} SNum' + 1 & judgement : success \\ SNum' & judgement : fail \end{cases} \quad (2)$$

$$SRate = \begin{cases} \frac{SNum'+1}{SNum'/SRate'+1} & judgement = `success' \\ \frac{SNum'}{SNum'/SRate'+1} & judgement : fail \end{cases} \quad (3)$$

Where the $SNum'$ and $SRate'$ represent the total success times and the success rate before updating, respectively.

Suppose that Consumer C is satisfied with the service provided by Provider P, C will give the evaluation result, 'success' for P on this service. In this situation, Formulas 2 and 3 will be used to update the record of P's experience from (0.7, 35) to (0.706, 36).

Suppose that Consumer C is not happy with the service, C will give elevation result, 'fail' for P on this service. In this situation, Formulas 2 and 3 will be used to update the record of P's experience from (0.7, 35) to (0.686, 35).

By using this updating method, PBTrust can dynamically update records of service experience for all agents in open environments and can also accumulate information to show general performance of each agent, without a central control mechanism.

IV. Priority-based Trust Calculation

The Priority-based Trust Calculation Module is used to produce the reputation values for potential service providers from four perspectives, which are the provider's experience on the service, the similarity of priorities distributions on attributes between the referenced service and the requested service, the suitability of the potential provider for the requested service and the time effectiveness of ratings from third parties. These perspectives have the contributions to the final reputation value from different views and are defined by separate formulas. This section gives the detail introduction of this module.

A. Design Consideration and the Principle of Priority-based Trust Calculation

In order to produce reliable and robust trust values for potential service providers, we develop a priority-based trust calculation mechanism based on the following considerations. Firstly, the third party reference is used to derive the reputation of providers. Secondly, the term 'suitability' is introduced to predict the potential performance of a provider for requested service based on the information from a third party reference about the provider's previous performance and the information of new priority requested by the consumer. Thirdly, the similarity measurement between the priority distribution on attributes of the service from a reference report and the priority distribution on attributes of the service requested from a consumer is also considered. Fourthly, the timestamp of the reference report is taken into account to reduce the contribution of out-of-date references from third parties. Fifthly, the service experience is also used for the trust calculation. Sixthly, the influence of all ratings from different referees are also considered. Finally, the trust value of a potential provider is calculated based on the above factors.

Based the design consideration, we develop the following formula for the priority-based trust calculation in PBTrust model.

$$Trust = EW \times \frac{\sum_{k=1}^{RN} Sim_k \times SInd_k \times TStamp_k}{RN} \quad (4)$$

where, EW represents the experience weight of the provider, Sim_k refers to the similarity of priority distribution of attributes in the service from the k^{th} reference report of requested service, $SInd_k$ is the suitability indicator based on the information of the k^{th} reference's ratings and the priorities in the requested services, $TStamp_k$ represents timestamp for the k^{th} reference, and RN is the number of references requested by the consumer and $RN > 0$.

The detail design for calculation of items EW, Sim_k, $SInd_k$, and $TStamp_k$ in Formula 4 are introduced in following subsections, respectively.

B. Experience Weight Calculation

Experience weight EW represents the general performance of a service provider on this service. The higher the experience weight, the more contribution to the trust calculation. EW is constructed by two factors, Fsr and Fsn. Fsr represents the contribution to EW from the successful rate while Fsn is the contribution to the EW from the total number of successful performance of the provider in history. EW is defined by the following formula.

$$EW = Fsr \times Fsn \quad (5)$$

The Fsr can be obtained by the following formula.

$$Fsr = \begin{cases} 0 & SRate < Sthreshold \\ 1 & SRate \geq Sthreshold \end{cases} \quad (6)$$

The idea behind Formula 6 is to evaluate whether the success rate on the service by the provider meets the basic requirement by the consumer, i.e. $SRate \geq Sthreshold$ and vice versa. If $SRate \geq Sthreshold$, the provider can reach the basic requirement to provide the request service, but whether it can be selected to offer service will be based on its trust value, $Fsr = 1$. Otherwise, $Fsr = 0$.

Fsn is defined by Formula 7.

$$Fsn = 1 - e^{-\frac{SNum}{\lambda}} \tag{7}$$

The reason for calculating Fsn by using an exponential increasing function is that the high success number on the service means the rich experience. When the success number achieves a very large value, the increase of Fsn becomes slowly. Here λ is a coefficient to control the speed changing in the curve which can be adjusted by users based on different application domains.

C. Similarity Calculation

To what extent, can the reference reflect the potential performance on the requested service? To answer this question, we should consider the similarity of priorities between the requested service and the referenced service. In PBTrust model, we use a matrix to describe a service (recall Definition 1). Since attributes in both requested service and a referenced service are in the same order, we can omit attributes during similarity calculation. Now, a service description matrix becomes a vector which includes priority values for corresponding attributes. We can use dot product of two vectors.If angle between two vectors' direction are named θ, the dot product of two vectors indicates the cosine value of angle θ in mathematics.

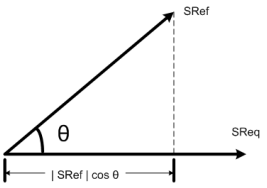

Figure 1. Dot product of two vectors

From figure.1 we can see that the vector $SReq$ means the priorities vector of service request and the $SRef$ indicates the priorities vector of service reference. θ is the angle between vector $SReq$ and $SRef$. Because all priorities of attributes are positive numbers and the sum of them is 1, so the range of angle θ is [0°, 90°], so the range of $\cos\theta$ is [0,1]. If $\theta = 0°$ and $\cos\theta = 1$ means there are no difference between two vectors' direction and the attributes priorities of requested service and referenced service are the same, so the provider's performance in reference can completely reflect the requested service. Oppositely, if $\theta = 90°$ and $\cos\theta = 0$ means there are the biggest difference between two vectors' direction and the attributes priorities of requested service and referenced service are totally different, so the provider's performance in reference can't reflect the requested service. For example, there are a service request with the attributes Cost, Speed and Quality and the priority of each attributes is as vector $Req =< 0.3, 0.5, 0.2 >$. Then two of the references with the same attributes and orders from service provider 1 and 2's vectors are $=< 0.3, 0.5, 0.2 >$ and $=< 0, 0, 1 >$. Through calculating, we can see that the similarity of provider 1's reference and request service is 1 and provider 2's reference and request service is about 0.32. That means that the service of service provider 1's reference is the same service as the request service and the reference offered by service provider 2 is a service that a little like the requested service. So when we calculate the reputation of two service providers we must give different weights for that two different reference based on reference similarity.

D. Suitability Indicator Calculation

The purpose of suitability indicator is to predict the potential performance of a provider on the requested service by using two pieces of information, reference ratings and the priorities of attributes in the requested service. The suitability indicator of the i^{th} reference can be calculated by the following formula.

$$SInd_i = \sum_{k=1}^{n} R_k \times CW_k \tag{8}$$

where CW_k represents the weight of k^{th} attributes for the requested service by the consumer, and R_k is rating vale for the k^{th} attribute given by the i^{th} referee.

E. Timestamp Calculation

The purpose of using the timestamp to evaluate the influence of references on the trust value is to eliminate or reduce the effect of out-of-date ratings depending on the value of T in a reference, (recall Definition 5). The method for the timestamp calculation is borrowed from the same concept used in the CR model [15]. Timestamp for the i^{th} reference report is calculated by the following formula.

$$TStamp_i = e^{-\frac{\Delta t(i)}{\lambda}} \tag{9}$$

where $\Delta t(i)$ means the time difference between the time when the i^{th} reference was generated and the current time, and λ is an coefficient to control the speed changing in the time curve depending on application domains.

V. Experiment and Comparison

The purpose of this experiment is to approve that our priority-based trust calculation and rich context format to represent reputation ratings can provide better performance than the CR model in service provider selection. Due to the page limitation, the experiment provided in this paper

Provider No.	Ratings
P_1	< 100, 50, 0 >
P_2	<100, 0, 50 >
P_3	<50, 100, 0 >
P_4	<50, 0, 100 >
P_5	<0, 100, 50 >
P_6	<0, 50, 100 >

Table I
REFERENCE RATINGS FOR PROVIDERS

emphasizes only on the two items in trust calculation module, Similarity of Priority Distribution on attributes between reference service and requested service and Suitability Indicator (recall Formula 3). Timestamp item (*TStamp*) and Experience Weight item (*EW*) are not tested in this experiment.

A. Experimental Setting

In the experiment, we use 6 service providers, 60 service consumers, and 4 different scenarios under different similarities to test PBTrust performance in a open environment. The notable model, CR model, was used as the benchmark to evaluate the test results of PBTrust.

Provider setting: In this experiment, each service provider only needs to provide 1 reference report to show their previous performance on requested service. The service used in the experiment contains 3 attributes, which are cost, speed, and quality. In order to simplify the experiment, we let 6 reference reports provided from 6 potential providers, respectively under the same priority distribution on attributes. The values of service experience of six providers are all set as <100%,100>. The reason for this setting is to remove influence of Item EW (experience weight, recall Formula 3) for the trust calculation since testing EW is not the main purpose of this experiment. The service description used in 6 reference reports is described as follows based on Definition 1.

$$SDes = \begin{pmatrix} Cost & Speed & Quality \\ 0.0 & 0.3 & 0.7 \end{pmatrix}$$

From the service description, we can know that the quality was the most important attribute in the service and the cost is not an issue for previous consumers who provided reference reports.

We name six providers as P_1, P_2, P_3, P_4, P_5, and P_6. Six reference reports show the performance of six providers under the same service description. These reports give ratings of providers for their performance in each attributes show in the following table.

Consumer setting: The consumer sends the following 3 service requests with different priority distribution on attributes of the service. The purpose for using three difference cases is to see the difference between PBTrust and CR models when the similarity value, between priority distributions on attributes in reference services and that of in requested service, changes. Similarity value (Sim) is calculated by Formula 7.

The requested priority distributions for three cases are given below.

Case 1: $Sim = 1$

$$SDes = \begin{pmatrix} Cost & Speed & Quality \\ 0.0 & 0.3 & 0.7 \end{pmatrix}$$

Case 2: $Sim \approx 0.1$

$$SDes = \begin{pmatrix} Cost & Speed & Quality \\ 0.8 & 0.2 & 0.0 \end{pmatrix}$$

Case 3: $Sim \approx 0.77$

$$SDes = \begin{pmatrix} Cost & Speed & Quality \\ 0.2 & 0.5 & 0.3 \end{pmatrix}$$

B. Experiment Analysis

Case 1: Sim=1

The requested service has the same priority distribution with completed service of 6 potential providers. The trust values by using PBTrsut and the CR models of 6 providers are listed separately in the following table.

Provider No.	Trust in PBTust	Trust in CR
P_1	15	15
P_2	35	35
P_3	30	30
P_4	70	70
P_5	65	65
P_6	85	85

Table II
TRUST VALUES IN CASE 1

From Table 2, we can see that P_6 has the highest trust value in both models and will be selected by the consumer in Case 1 by both models. The performance of two models are same when similarity value is 1, i.e. the distribution of priorities on attributes in both referenced and requested services are same.

Case 2: $Sim \approx 0.1$

The similarity value is very low in this case. That means the requested service has very different emphases on attributes comparing with the referenced service. In this case, Attribute 'cost' has the highest priority value while the same attribute in the referenced service has the lowest priority value. That means this attribute should play an important role for evaluating a trust value on a potential provider for this requested service. The trust values of 6 providers by using PBTrsut and the CR models are listed separately in Table 3. From Table 3, we can see that P_1 has the highest trust value in PBTrust and P_6 has the highest trust value in the CR model.

From Table 1 we can see the reference ratings for P_1 is < 100, 50, 0 > and for P_6 is <0, 50, 100 >. Even if the overall ratings for P_1 and P_6 are same, the ratings on

Provider No.	Trust in PBTust	Trust in CR
P_1	8.6	15
P_2	7.6	35
P_3	5.7	30
P_4	3.8	70
P_5	1.9	65
P_6	1	85

Table III
TRUST VALUES IN CASE 2

individual attributes are different. P_6's performance on Attribute, 'Cost', is very low and cannot meet the requirement in the requested service. P_1 has the best performance on Attribute, 'Cost', and same performance as P_6 on Attribute, 'Speed'. Even if P_1's performance on Attribute, 'Quality', this attribute is not considered seriously in the consumer. Therefore, P_1 is the most suitable provider for Case 2. The experimental result shows that PBTrust model selected the right provider while the CR model chose a wrong one. P_6 has a risk to complete the service in Case 2.

Case 3: $Sim \approx 0.77$

Case 1 and Case 2 are two extreme cases. Case 3 represents one of normal general cases. The similarity value is 0.77 in this case. The consumer put 'Speed' in the first priority and also considers 'Quality' and 'Cost' as the second and third priorities, respectively. The trust values of 6 providers by using PBTrsut and the CR models are listed separately in Table 4. From Table 4, we can see that P_5

Provider No.	Trust in PBTust	Trust in CR
P_1	34.5	15
P_2	26.8	35
P_3	46	30
P_4	30.7	70
P_5	49.8	65
P_6	42.2	85

Table IV
TRUST VALUES IN CASE 3

has the highest trust value in PBTrust and P_6 still has the highest trust value in the CR model.

If we look at the detail ratings for these two potential providers from Table 1, we can get the reference ratings for P_5 is $<$ 0, 100, 50 $>$ and for P_6 is $<$0, 50, 100 $>$. The overall ratings for P_5 and P_6 are same but the ratings on the most important attribute, 'Speed' shows the big difference. Obviously, the performance of P_5 on this attribute is much better than that of P_6. Even if P_6's performance on Attribute, 'Quality', is higher than that of P_5, the consumer assigned the a lower priority on this attribute for the requested service. For overall consideration, P_5 is a more suitable provider than that of P_6 in this case.

VI. CONCLUSION AND FUTURE WORK

In this paper, the PBTrust model was proposed for selection of service providers in general service-oriented environments. In PBTrust, the trust generation on a service provider was based on the consideration of third party evaluation, its overall performance in the history, the suitability for the requested service under requested priories of attributes of the service by the consumer, as well as weighted ratings from third party references based on time stamps. Since the reputation of a service provider is derived from third party referees in a rich context format,it can be easily used to handle different types of services. In PBTrust, the record of agent experience can also be updated dynamically without using a central mechanism. This feature can make PBTrust be able to work in open and dynamic environments. The experimental result demonstrated that PBTrust could perform better than the CR model for service selection in service-oriented environments.

The experiment in this paper was in a relatively simple scenario and conducted in a laboratory environment. In the future, we will do the further experiments with complex scenarios and test PBTrust in real world applications.

REFERENCES

[1] I. Foster, C. Kesselman, and S. Tuecke, "The anatomy of the grid: Enabling scalable virtual organizations," *The Interntional Journal of High Performance Computing Applications*, vol. 15, no. 3, pp. 200–222, 2001.

[2] H. M. Deitel, P. J. Deitel, and T. R. Nieto, *e-Business & e-Commerce How to Program*. Prentice Hall, 2001.

[3] F. Ren, M. Zhang, and K. M. Sim, "Adaptive conceding strategies for automated trading agents in dynamic, open markets," *Decision Support Systems*, vol. 48, no. 2, pp. 331–341, 2009.

[4] D. Saha and A. Mukherjee, "Pervasive computing: a paradigm for the 21st century," *IEEE Computer*, vol. 36, no. 3, pp. 25–31, 2003.

[5] R. Steinmetz and K. Wehrle, Eds., *Peer-to-Peer Systems and Applications*, ser. Lecture Notes in Computer Science. Springer, 2005, vol. 3485.

[6] D. Gambetta, *Trust:Making and Breaking Cooperative Relations*. Department of Sociology,University of Oxford, 2000, ch. 13, pp. 213–237.

[7] S. D. Ramchurn, D. Huynh, and N. R. Jennings, "Trust in multi-agent systems," *The Knowledge Engineering Review*, vol. 19, pp. 1–25, 2004.

[8] G. Zacharia and P. Maes., "Trust management through reputation mechanisms," *Applied Artificial Intelligence*, vol. 14, no. 9, pp. 881–907, 2000.

[9] J. Sabater and C. Sierra, "Regret: reputation in gregarious societies," in *Proceedings of the fifth international conference on Autonomous agents*, 2001, pp. 194 – 195.

[10] S. Sen and N. Sajja, "Robustness of reputation-based trust: Boolean case," in *AAMAS'02: Proceedings of the first international joint conference on Autonomous agents and multiagent systems*, 2002, pp. 288–293.

[11] A. S. Ali, S. A. Ludwig, and O. F. Rana, "A cognitive trust-based approach forweb service discovery and selection," in *Proceedings IEEE of the Third European Conference on Web Services (ECOWS'05)*, 2005, pp. 1–5.

[12] K. K. Fullam and K. S. Barber, "Dynamically learning sources of trust information: experience vs. reputation," in *AAMAS'07: Proceedings of the 6th international joint conference on Autonomous agents and multiagent systems*, Honolulu, Hawaii, 2007, pp. 1055–1062.

[13] H. Billhardt, R. Hermoso, S. Ossowski, and R. Centeno, "Trust-based service provider selection in open environments," in *SAC '07: Proceedings of the 2007 ACM symposium on Applied computing*. ACM, 2007, pp. 1375–1380.

[14] M. Sharmin, S. I. Ahamed, S. Ahmed, and H. Li, "Ssrd+: A privacy-aware trust and security model for resource discovery in pervasive computing environment," in *COMPSAC '06: Proceedings of the 30th Annual International Computer Software and Applications Conference*. IEEE Computer Society, 2006, pp. 67–70.

[15] T. D. Huynh, N. R. Jennings, and N. R. Shadbolt, "Certified reputation: How an agent can trust a stranger," in *AAMAS'06: Proceedings of the fifth international joint conference on Autonomous agents and multiagent systems*, 2006, pp. 1217–1224.

[16] J. Sabater and C. Sierra, "Reputation and social network analysis in multi-agent systems," in *AAMAS'02: Proceedings of the first international joint conference on Autonomous agents and multiagent systems*. Bologna, Italy: ACM, 2002, pp. 475 – 482.

[17] M. Sensoy and P. Yolum, "A context-aware approach for service selection using ontologies," in *AAMAS'06: Proceedings of the fifth international joint conference on Autonomous agents and multiagent systems*, 2006.