Trust Model for Optimized Cloud Services

P.S. Pawar[1,2], M. Rajarajan[1], S. Krishnan Nair[2], and A. Zisman[1]

[1] City University London, London EC1V 0HB, United Kingdom
r.muttukrishnan@city.ac.uk, a.zisman@soi.city.ac.uk
[2] British Telecommunications, Security Practice, Adastral Park, Ipswich IP5 3RE, UK
{pramod.s.pawar,srijith.nair}@bt.com

Abstract. Cloud computing with its inherent advantages draws attention for business critical applications, but concurrently expects high level of trust in cloud service providers. Reputation-based trust is emerging as a good choice to model trust of cloud service providers based on available evidence. Many existing reputation based systems either ignore or give less importance to uncertainty linked with the evidence. In this paper, we propose an uncertainty model and define our approach to compute opinion for cloud service providers. Using subjective logic operators along with the computed opinion values, we propose mechanisms to calculate the reputation of cloud service providers. We evaluate and compare our proposed model with existing reputation models.

Keywords: Cloud, Trust, Reputation, SLA, Subjective logic.

1 Introduction

Cloud computing has been recognised as an important new paradigm to support small and medium size businesses and general IT applications. The advantages of Cloud computing are multifold including better use and sharing of IT resources, unlimited scalability and flexibility, high level of automation, reduction of computer and software costs, and access to several services. However, despite the advantages and rapid growth of Cloud computing, it brings several security, privacy and trust issues that need immediate action. Trust is an important concept for cloud computing given the need for consumers in the cloud to select cost effective, trustworthy, and less risky services [2]. The issue of trust is also important for service providers to decide on the infrastructure provider that can comply with their needs, and to verify if the infrastructure providers maintain their agreements during service deployment.

The work presented in this paper is being developed under the FP7 EU-funded project called OPTIMIS [5][13] to support organisations to externalise services and applications to trustworthy cloud providers. More specifically, the project focuses on service and infrastructure providers. One of the main goals of OPTIMIS is to develop a toolkit to assist cloud service providers to supply optimised services based on four different aspects, namely *trust, risk, eco-efficiency*, and *cost*. As part of the overall goal in OPTIMIS, this paper, describes a trust model to support service providers (SP) to verify trustworthiness of infrastructure providers (IP) during deployment and operational phases of the services supplied by the service providers.

T. Dimitrakos et al. (Eds.): IFIPTM 2012, IFIP AICT 374, pp. 97–112, 2012.

The aim of the Service Provider (SP) is to offer efficient services to its customers using resources of the Infrastructure Provider (IP). The IP aims to maximize its profit by efficient use of its infrastructure resources ensuring that it provides good service to the SP and meeting all its requirements. The trust framework is active during the service deployment and service operation phases. The trustworthiness of the IP and the SP are monitored during these two phases of the service life cycle.

The scope and focus of this paper is mainly to evaluate the trustworthiness of the IP performed by the SP. During the *service deployment phase,* the objective of the SP is to select the most suitable IP for hosting its service based on the degree of trust expected from an IP. During the *service operation phase*, the SP monitors the IP's trust level and takes corrective actions. An example of an action is to select an alternative IP when the trust level of the IP is unacceptable, based on a negotiated level.

The trust model described in this paper calculates trust values based on three different parameters, namely (i) *compliance of SLA parameters* (e.g., when the IP fulfils the quality aspect specified in the SLA between an SP and the IP), (ii) *service and infrastructure providers satisfaction ratings* (e.g., when SP supplies a rating for the IP where the SP is being deployed), and (iii) *service and infrastructure provider behavior* (e.g., if the SP continues to choose the same IP independent of the rating that it has supplied for the IP). In the model, the satisfaction values can be either explicitly provided in terms of ranking measurements, or inferred based on relationships between the service and infrastructure providers, and behavior of the providers in terms of constant use of services, service providers, and infrastructure providers.

For each of the different parameters above, trust values are calculated based on an opinion model [8]. As in the case of [8][17], we have developed an opinion model that considers *belief*, *disbelief*, and *uncertainty* values. Our model is based on an extension of the Josang's opinion model [8], in which we consider uncertainty when calculating *belief* and *disbelief* values. In [8], uncertainty is considered based on the amount of evidence, in which uncertainty increases if the amount of evidence decreases. As in the case of [17], in our model uncertainty is considered based on the amount of evidence and on the dominance that exist between the positive and negative evidences. If the number of positive (belief) evidences is closer to the number of negative (disbelief) evidences, the uncertainty about the proposition increases. For example, if the number of times that an infrastructure provider (IP1) violates a quality property is the same as the number of times that IP1 does not violate the same property, the level of uncertainty of IP1 for that property increases.

In our model, as in the case of [17], but contrary to [8], the belief and disbelief values also consider uncertainty. The difference between our model and the model in [17] is with regards to uncertainty calculation. In [17], certainty is calculated as a *Probability Certainty Density Function (PCDF)* which is probability density function of the probability of positive experience. With no knowledge the uniform distribution has certainty of zero and as the knowledge increases the probability mass shifts, deviating from the uniform distribution, increasing the certainty towards one.

The remaining of this paper is structured as follows. Section 2 presents an example that will be used throughout the paper to illustrate the work. Section 3 describes the trust model used by the framework. Section 4 discusses the evaluation of the model. Section 5 provides an account of related work. Finally, Section 6 provides concluding remarks and future work.

2 Cloud Computing Example Scenario

In order to illustrate the work described in the paper, we present a Cloud computing *education application* that is being deployed for Bristish Telecom customers such as Universities and other education institutions. The education application allows Universities and education institutions to have virtual laboratory environments for students, staff, and all other members of the institutions hosted over the cloud, providing access to the institution's applications, desktops, and servers.

The key features of the application includes: i) flexibility to work from anywhere and anytime allowing the users to access the desktop and corporate applications from any PC, MAC, thin client or smartphone; ii) reduction of desktop management cost enabling the IT department to add, update, and remove applications in an easy way; iii) provision of good data security, good access control, and scalable storage platforms; iv) provision of scalability and elasticity for compute resources; v) comprehensive monitoring and management to support use and capacity planning and space usage; and vi) backup and recovery functions. The application has several components, namley: web interface, active directory, desktop delivery controller (DDC), virtual machines, and storage. The web interface passes user credentials to DDC, which authenticates users against the active directory. The virtual machine is a virtual desktop accessed by end users after receiving the connection details.

For evaluating our proposed model we consider a scenario in the education application with five Service Providers (SPs) and five Infrastructure Providers (IPs). An SP hosts the application with its multiple components either at one IP or at multiple IPs. The SP may also use a broker for the IP services. This example scenario considers that all the SPs host education applications. Fig. 1 shows the education application deployed by vairous SPs. As shown in the figure, each IP has multiple datacenter sites which may be geographically distributed. Each of these datacenters can have a large number of physical hosts/machines available with capabilities to execute multiple virtual machines.

The three datacenters of IP1 is composed of three, one, and one physical hosts, respectively. The IP1's datacenter with three physical hosts deploy five, three and one virtual machines, respectively. The figure shows that IP1 is in a federation with IP2 and IP3. In this case, IP1 is capable of leasing capacity from IP2 and IP3. Fig. 1 also shows a situation of a bursting scenario, in which organizations can scaleout their infrastructures and rent resources from third parties, as and when its is necessary. For example, as shown in Fig. 1, infrastructure provider IP1 may burst to infrastructure provider IP4 to meet the SLA requirements of any SP. Fig. 1 also shows the brokers that are associated with the IPs and are capable of renting infrastructure resources from all the IP's. The figure indicates that the SPs have deployed the application in the cloud environment with different constraints (options), as described below.

Option 1: The application is deployed at a single IP, with a constraint of having all components of the application on the same host. SP1 in the figure have all its virtual machines (VM1.1, VM1.2, and VM1.3) running on a single physical host of IP1.
Option 2: The application is deployed in a single datacenter of an IP. SP1 and SP2 have all its virtual machines running on the same datacenter of IP1.

Option 3: The application is deployed in a single IP's administration boundary (restrict usage of federation resources). SP1, SP2 and SP3 have all its virtual machines in the administration boundaries of IP1.
Option 4: The application is dployed in more than one IP. SP4 and SP5 deploy the application in IP1, IP4 and IP1, and IP5, respectively.

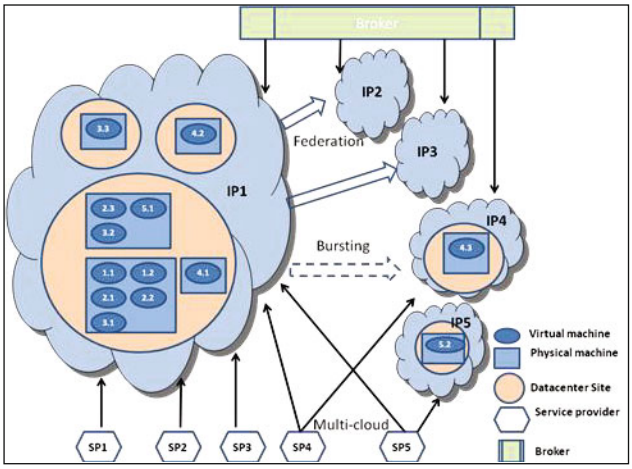

Fig. 1. Cloud computing educational application example

Several other deployment scenarios are possible, but for illustrative purpose we will concentrate on the above situations. Although Fig. 1 shows that SP1, SP2 and SP3 have currently deployed applications on only IP1, it is possible that they may have used other IPs (IP2, IP3, IP4 and IP5) in the past. Similarly, IP4 and IP5 have also used other IPs other than the current ones.

In the scenario, we assume that the institution that decides to use the education application above has SLAs with the SP describing expected quality of the services. The SLAs specify several indicators with which the SP is required to comply, and any violations may lead to penalty payments, as well as negative impact in the customer's satisfaction. Examples of SLA indicators are cpu, disk space, memory, and number of desktops. In order to meet the customer's requirements, the SP that uses the infrastructure services from the IPs also have SLAs with the IP. An SLA between an SP and an IP considers all the existing SLA's with the various customers and the possibility of growing the demand of the application. An SLA between an SP and IP represents elasticity requirements to support the SP to demand more resources dynamically based on the requirements. For example, when the application receives a request for a new desktop, it requests a virtual machine to be created in the infrastructure of the IP where the application is deployed. Similarly, the application can receive requests to increase memory, cpu, or disk space for the existing virtual desktops, which are forwarded to the IP to fulfil the requirements. If the IP, at any point of time fails to provide the requested resources, or is not able to maintain the resource requirements of existing virtual desktops, then this may lead to SLA violations for the corresponding indicators.

3 Trust Model

As described in Section 1, *Trustworthiness* of an IP is modelled using *opinion* obtained from three different computations, namely (i) *compliance of SLA parameters (SLA monitoring),* (ii) *service provider satisfaction ratings (SP ratings)*, and (iii) *service provider behavior (SP behavior).* The *opinion* is expressed in terms of *belief*, *disbelief*, *uncertainty* and *base rate* which is used in conjunction with the subjective logic [8].

The *opinion* of an entity (SP or IP) A for a proposition x is given as $W^A_x = (b^A_x, d^A_x, u^A_x, a^A_x)$, where b^A_x is the belief in the proposition, d^A_x is the disbelief in the proposition, u^A_x is the uncertainty of the proposition, a^A_x is base rate that provides the weight of uncertainty that contributes to the probability expectation. All b_x, d_x, u_x, a_x Є [0.0, 1.0], and $b_x+d_x+ u_x=1$.

The *trustworthiness (T)* of an IP is modelled as the expectation of the combined opinion of all the three computations. The opinions are combined using the conjunction operator, consensus operator, and the discounting operator in the subjective logic [8], as defined below:

T=Expectation $(W_{(SPB \otimes SPR) \wedge SLA})$	$W_{(SPB \otimes SPR) \wedge SLA} = (W_{SPB} \otimes W_{SPR}) \wedge W_{SLA}$

where $W_{SLA,}$ W_{SPR}, W_{SPB} are opinions obtained from the SLA monitoring (SLA), SP ratings (SPR), and SP behavior (SPB) values, respectively. The symbol $\wedge$ is the *conjunction operator* used to combine the opinions, and $\otimes$ is the *discounting operator* used as the recommendation operator. If $W_x = (b_x, d_x, u_x, a_x)$ and $W_y = (b_y, d_y, u_y, a_y)$, then $W_{x \wedge y} = (b_{x \wedge y}, d_{x \wedge y}, u_{x \wedge y}, a_{x \wedge y})$.

Consider A and B two agents, where $W^A_B = (b^A_B, d^A_B, u^A_B, a^A_B)$ is A's opinion about B's advice, and let x be the proposition where $W^B_x = (b^B_x, d^B_x, u^B_x, a^B_x)$ is B's opinion about x expressed as an advice to A. In this case, W^{AB}_x is called the discounting ($\otimes$) of W^B_x by W^A_B and is given as $W^{AB}_x = W^A_B \otimes W^B_x = (b^{AB}_x, d^{AB}_x, u^{AB}_x, a^{AB}_x)$.

Opinion Representation. For a proposition x, the opinion is given by

$$W_x = (b_x, d_x, u_x, a_x) \text{ , with}$$

$b_x = c\ r / t$	$d_x = c\ s\ / t$	$u_x = t / (r\ s + f^2 + 1)$	$c = 1 - u_x$

where: r is the amount of positive evidence; s is the amount of negative evidence; t is the total evidence given as $t=r+s$; c or $c(t)$ *or* $c(r,s)$ is certainty that is a function of the total evidence; and f is the distance of focus to the centre of an ellipse.

The proposed opinion model considers two aspects of uncertainty due to the evidence at hand, namely: i) as the amount of evidence increases the uncertainty reduces; and ii) in a given total evidence, as the positive or negative evidence dominates, the uncertainty decreases, and as the positive and negative evidence equals, the uncertainty increases. These two aspects of uncertainty exhibit behavior similar to the properties of an ellipse, considering its size and shape, controlled by its axis and area.

In our model, uncertainty is defined as a function of an ellipse area and shape. More specifically, the uncertainty model is derived using the properties of an ellipse wherein the positive and negative evidence is mapped to the major and minor

semi-axes of an ellipse. The first aspect of uncertainty (i.e. increases in evidence, decreases the uncertainty) is achieved by using the area of the ellipse given by the product of its two semi-axes. As the positive and negative evidence is being mapped to the major and minor semi-axes of ellipse, the increase in the major and minor semi-axes results in the increase of the area of ellipse and decrease of the uncertainty. The second aspect of uncertainty is due to dominance between positive and negative evidence, which is captured using the shape of an ellipse. The shape of an ellipse is a function of its two semi-axes. The positive and negative evidence being mapped to the semi-axes of an ellipse, as the major semi-axis continues to dominate, the distance of focus with the centre is a positive value and as the two semi-axes equals, this distance approaches to zero, transforming to a circle.

The change in major and minor semi-axes affects the distance of focus with the centre which is given as $f = sqrt\ (\ a^2 - b^2)$. If the total evidence is fixed to a constant, the variation of the positive and negative evidence affects the shape of the ellipse. If the positive and negative evidence equals, this makes $f = 0$, transforming the ellipse to a circle. This adds to a highest uncertainty in a given total evidence. As the positive and negative evidence continues to dominate, this leads to a positive value for f and this value is maximum when either positive or negative evidence in the total evidence is zero. This adds to a lowest uncertainty in a given total evidence. Both properties of uncertainty are captured in the uncertainty definition below:

$u = t / (r\ s + f^2 + 1)$	for $t \geq 1$	and	$u = 1$	for $t < 1$

where r is the amount of positive evidence; s is the amount of negative evidence; t is the total evidence given as $t=r+s$; and f is the distance of focus to the centre of an ellipse given as $f = sqrt\ (r^2 - s^2)$ considering $r > s$; The certainty in the opinion model and the expectation of the opinion about a proposition x is given as:

$c(t) = 1 - u$	$E(x) = b_x + a_x u_x$

where $c(t)$ is the function of total evidence t and can also be represented as a function of positive and negative evidence given as $c(r,s)$. The opinion model uses certainty $c(t)$ to model the *belief*, *disbelief* and *uncertainty*.

SLA Monitoring. The SLA monitoring determines the opinion about an IP from the SLAs that the IP have established with the SPs for their services. The SP for each of its service has a single SLA that includes several indicators (e.g.; cpu, memory, disk space, number of virtual machines (vms)). For each indicator of an SLA, there is an associated monitor that evaluates the compliance/non-compliance of the indicator.

The SLA monitoring opinion about an IP is a two-step process. In the first step, a *consensus opinion* is created for an indicator type (e.g.; cpu) based on information from all the monitors verifying the compliance of the indicator. This opinion indicates the trust of an IP only based on the indicator used to create the *consensus opinion*. In the second step, a *conjunction opinion* is created about the IP for either a set of indicators or for all the indicators based on the requirement. The *conjunction opinion* indicates the trust of an IP for the set of indicators based on SLA monitoring.

Consider that there are *m* indicator types and *n* monitors associated with each indicator type. In this case, the opinion of the SLA monitoring is given as:

$$W_{SLA} = W_1^{(M1,1),\ldots,(M1,n)} \wedge W_2^{(M2,1),\ldots,(M2,n)} \wedge \ldots \wedge W_m^{(Mm,1),\ldots,(Mm,n)}$$

where, $W_1^{(M1,1),\,(M1,2),(M1,3),\ldots,(M1,n)}$ is the consensus opinion for the indicator type '1' given by monitors M1,1 to M1, *n* belonging to different SLAs. If $W^A_x = (b^A_x, d^A_x, u^A_x, a^A_x)$ *and* $W^B_x = (b^B_x, d^B_x, u^B_x, a^B_x)$ are the opinions given by agent *A* and agent *B,* respectively for the same proposition *x,* then the *consensus opinion* is given as in [8] by: $W^{A,B}_x = W^A_x \oplus W^B_x = (b^{A,B}_x, d^{A,B}_x, u^{A,B}_x, a^{A,B}_x)$

Example. In order to illustrate, consider the education application described in Section 2. Consider a case wherein, at that end of academic year most university students need high computation resources such as large number of virtual machines, memory space, cpu and disk space for doing individual projects. For each of the Universities the requested resource to the SP is within the agreed SLA. The SP demands resources from the IP. As in the example scenario, since IP1 have all five SPs hosting the education application, the demand to increase the resources occurs almost in the same time frame. Given the constraint that IP1 cannot acquire resources from other IPs for these applications, there is a violation of the SLA after verifying that IP1 has no additional resource of its own to be provided.

In the scenario IP1 has five SLAs, with each of the SPs (SP1 to SP5) for four different indicator types (cpu, memory, disk, and virtual machine). Assume SLA1 with SP1, SLA2 with SP2, and so on. Consider the existence of monitors associated with each indicator of the SLAs. Assume four monitors (M1, M2, M3 and M4) to be associated with SLA1 for cpu, memory, disk space, and virtual machine, respectively. Similarly, monitors M5 to M8, M9 to M12, M13 to M16 and M17 to M20 are associated with SLA2, SLA3, SLA4 and SL5, for the various SLA indicators.

Each of the monitors associated with the indicators provides information about the compliance of the respective indicator for an IP. If we consider that monitors M1, M2, M3 and M4 indicated 150 compliances and 10 non-compliance (150 positive evidence and 10 negative evidence) for IP1. The opinions given by the monitors for SLA1 are calculated using the proposed opinion model as :

$$W_{CPU}^{M1} = (b^{M1}_{CPU}, d^{M1}_{CPU}, u^{M1}_{CPU}) = (0.93122, 0.062082, 0.006694)$$

$$W_{mem}^{M2} = W_{disk}^{M3} = W_{vm}^{M4} = (0.93122, 0.062082, 0.006694)$$

If we consider that all the other monitors M5-M20 associated with SLA2, SLA3, SLA4 and SLA5 also have 150 compliance and 10 non-compliance indicators, the opinion provided by these monitors are the same as the above ones.

The opinion for IP1 with respect to *cpu* is given as the *consensus opinion* of the five monitors M1, M5, M9, M13 and M17 as follows:

$W_{CPU}^{M1,M5,M9,M13,M17} = (b^{M1,M5,M9,M13,M17}_{CPU}, d^{M1,M5,M9,M13,M17}_{CPU}, u^{M1,M5,M9,M13,M17}_{CPU}) =$ (0.936238, 0.062416, 0.001346)

Similarly, the opinion for IP1 based on memory, disk and virtual machine is:

$W_{mem}^{M2,M6,M10,M14,M18} = W_{disk}^{M3,M7,M11,M15,M19} = W_{VM}^{M4,M8,M12,M16,M20}$ =(0.936238, 0.062416, 0.001346)

The overall opinion for IP1 based on all the indicators of the SLAs is given as the *conjunction opinion* of all *consensus opinions* for each of the indicator as follows:

$W_{SLA} = W_{CPU}^{M1,M5,M9,M13,M17} \wedge W_{mem}^{M2,M6,M10,M14,M18} \wedge W_{disk}^{M3,M7,M11,M15,M19} \wedge W_{VM}^{M4,M8,M12,M16,M20} = (0.768325, 0.227246, 0.004428)$

SP Behavior. The SP behavior is defined in terms of the number of times the SP has used the infrastructure of an IP against the SPs total usage. An SP using a single IP for the majority of the times indicates the SPs good behavior towards an IP. The SP may use the infrastructure of an IP for one or more indicators specified in the SLA.

Consider that there are *m* indicator types that the IP has negotiated from all the 'q` SPs in the past. Let there be *m* monitors associated with each of the SPs to monitor how many times the SP used this IP for a given indicator, against its total usage for that indicator. Suppose that SP1 used IP1 five times, IP2 three times, and IP3 four times for cpu usage. This indicates that for cpu total usage of 12 times, SP1 has used IP1 five times. This information is used to model the opinion of SP1's behavior towards IP1 for cpu usage. Assume monitor M1,1 associated with the indicator of type '1' to monitor SP1's behavior towards IP1. In this case, the opinion is represented as $W_{SP1}^{M1,1}$. A single overall behavior of an SP towards an IP is given as a consensus opinion of all its indicators. The behavior of SP1 towards IP1 is given as:

$$(W_{SP1}^{M1,1} \oplus W_{SP1}^{M2,1} \oplus W_{SP1}^{M3,1} \oplus \ldots \oplus W_{SP1}^{Mm,1})$$

All 'q' behavior of SP towards an IP is given as the conjunction opinion as:

$$W_{SPB} = (W_{SP1}^{M1,1} \oplus \ldots \oplus W_{SP1}^{Mm,1}) \wedge \ldots \wedge (W_{SPq}^{M1,q} \oplus \ldots \oplus W_{SPq}^{Mm,q})$$

Example. In order to illustrate consider the education application described in Section 2 with monitors M1, M2, M3 and M4 verifying the compliance of the cpu, memory, disk and virtual machine usage, respectively, for SP1, and monitors M6-M8, M9-M12, M13-M16, and M17-M20 for SP2, SP3, SP4 and SP5. Suppose that monitor M1 associated with SP1, records that SP1 has opted to use IP1 for 200 times against SP1's 250 times total cpu usage. The opinion for the behavior of SP1 towards IP1 for cpu usage is calculated as:

$$W_{SP1}^{M1} = (b^{M1}_{SP1}, d^{M1}_{SP1}, u^{M1}_{SP1}) = (0.79579, 0.198947, 0.005263).$$

Similarly, assume that M2, M3 and M4 record the same usage as M1 for memory, disk space, and virtual machine, respectively. The opinions are calculated as:

$$W_{SP1}^{M2} = W_{SP1}^{M3} = W_{SP1}^{M3} = W_{SP1}^{M4} = (0.79579, 0.198947, 0.005263)$$

Consider that SP2 and SP3 have the same evidence as in the case of SP1, with the associated monitors for these SPs providing evidences as monitors M1, M2, M3 and M4. Consider SP4 with monitors M13-M16 and SP5 with monitors M17-M20 using other IPs different from IP1 for its resources consumption. Assume the monitors for SP4 and SP5 provide 100 positive evidences and 150 negative evidences for each of its indicators. This evidence is transformed to the opinions below:

$W_{SP4}^{M13} = W_{SP5}^{M17} = W_{SP4}^{M14} = W_{SP5}^{M18} = W_{SP4}^{M15} = W_{SP5}^{M19} = W_{SP4}^{M16} = W_{SP5}^{M20} = (0.39636, 0.594546, 0.009091)$

The behavior of SP1 towards IP1 (and of SP2 and SP3) are calculated as:

$W_{SP1}^{M1...M4} = W_{SP1}^{M1} \oplus W_{SP1}^{M2} \oplus W_{SP1}^{M3} \oplus W_{SP1}^{M4} = (0.798943, 0.199736, 0.001321)$

The behavior of SP4 and SP5 towards IP1 based is given as:

$$W_{SP4}^{M13M14M15M16} = W_{SP5}^{M17M18M19M20} = (0.399085, 0.598627, 0.002288)$$

The total SPs behavior towards an IP is given as the *conjunction* opinion of all SPs towards a single IP, given as:

$W_{SPB} = W_{SP1}^{M1...M4} \wedge W_{SP2}^{M5...M8} \wedge W_{SP3}^{M9...M12} \wedge W_{SP4}^{M13...M16} \wedge W_{SP5}^{M17...M20} = (0.081223, 0.917435, 0.001342)$

SP Ratings. The service provider satisfaction rating is calculated based on the rates of the services given by an SP using an IP. The SP provides separate ratings for each SLA indicators of the IP's services. The ratings are used to form an opinion about an IP. Similar to the other cases, the computation of SP ratings to provide an opinion about an IP is based on consensus and conjunction ratings. Consider q SPs available and each of these SPs providing its opinion for one or more of the m indicator types that the IP supports. The service provider satisfaction rating is calculated as:

$$W_{SPR} = W_1^{SP1,SP2...,SPq} \wedge W_2^{SP1,SP2...,SPq} \wedge \ldots \wedge W_m^{SP1,SP2...,SPq}$$

where, $W_i^{SP1,SP2...,SPq}$ is the consensus opinion for indicator type 'i' from SP1 to SPq.

Example. As an example, suppose that SP1 has provided 100 excellent and 5 worst ratings for each of cpu, memory, disk, and virtual machine indicators. These ratings are transformed into 100 positive and 5 negative evidences for each of these indicators, as per the mapping described above. Based on the evidence of ratings for IP1, the opinion that SP1 has about IP1 for its indicators is given as:

$$W_{CPU}^{SP1} = (b^{SP1}_{CPU}, d^{SP1}_{CPU}, u^{SP1}_{CPU}) = (0.94284, 0.047142, 0.010023)$$

$$W_{mem}^{SP1} = W_{disk}^{SP1} = W_{vm}^{SP1} = (0.94284, 0.047142, 0.010023)$$

Suppose that SP2, SP3, SP4 and SP5 have provided (200 excellent, 5 worst), (200 excellent, 10 worst), (200 excellent, 20 worst), (200 excellent, 30 worst) ratings, respectively for IP1 for each of the four different indicators. These evidences provide the following opinions of SP2, SP3, SP4 and SP5 about IP1, calculated as:

$$W_{CPU}^{SP2} = W_{mem}^{SP2} = W_{disk}^{SP2} = W_{vm}^{SP2} = (0.97073, 0.024268, 0.005003)$$
$$W_{CPU}^{SP3} = W_{mem}^{SP3} = W_{disk}^{SP3} = W_{vm}^{SP3} = (0.94761, 0.04738, 0.005012)$$
$$W_{CPU}^{SP4} = W_{mem}^{SP4} = W_{disk}^{SP4} = W_{vm}^{SP4} = (0.90450, 0.09045, 0.005046)$$
$$W_{CPU}^{SP5} = W_{mem}^{SP5} = W_{disk}^{SP5} = W_{vm}^{SP5} = (0.86513, 0.12977, 0.0051)$$

The capability of IP1 for cpu, memory, disk, and virtual machine are given as the consensus of all SP's opinion by:

$$W_{CPU}^{SP1} \oplus W_{CPU}^{SP2} \oplus W_{CPU}^{SP3} \oplus W_{CPU}^{SP4} \oplus W_{CPU}^{SP5} = (0.928743, 0.070133, 0.001124)$$

$W_{mem}^{SP1\ldots SP5} = W_{disk}^{SP1\ldots SP5} = W_{VM}^{SP1\ldots SP5} = (0.928743, 0.070133, 0.001124)$

The overall opinion formed for IP1 based on the ratings from the SPs is given as:

$$W_{SPR} = W_{CPU} \wedge W_{mem} \wedge W_{disk} \wedge W_{VM} = (0.744015, 0.252376, 0.003609)$$

SP Ratings Discounted by SP Behavior. The proposed trust model uses the behavior of the SP for discounting the opinion provided by the SP in SP ratings, for a particular indicator. More specifically, in the SP ratings, if SP1 is evaluating IP1 and is informed about the opinion of IP1 from SP2 regarding cpu indicator, this opinion of SP2 is discounted using SP2's behavior about cpu towards IP1.

In the case of SP behavior, if monitor M1,2 is associated with indicator type '1' to monitor SP2's behavior towards IP1, then this opinion is represented as $W_{SP2}^{M1,2}$. In the case of SP ratings, SP1 being informed about opinion from SP2 for IP1 based on indicator type '1' is represented as W_1^{SP2}. Based on the behavior of SP2 towards IP1 for cpu indicator, SP2's opinion for cpu is discounted. In other words, the opinion W_1^{SP2} is discounted by $W_{SP2}^{M1,2}$ value and is given as $W^{(M1,2)SP2}{}_1 = W^{M1,2}{}_{SP2} \otimes W_1^{SP2} = (b^{(M1,2)SP2}{}_1, d^{(M1,2)SP2}{}_1, u^{(M1,2)SP2}{}_1, a^{(M1,2)SP2}{}_1)$

SP ratings after discounting opinions using the SP behavior for each of the indicator, also follows the two-step process of *consensus* and *conjunction* to get the combined opinion of SP rating and SP behavior which are given as follows:

$$W_{(SPR \otimes SPB)} = W_{SPB} \otimes W_{SPR} = (W^{M1,1}{}_{SP1} \otimes W_1^{SP1}) \oplus (W^{M1,2}{}_{SP2} \otimes W_1^{SP2}) \oplus \ldots \oplus (W^{M1,q}{}_{SPq} \otimes W_1^{SPq}) \wedge (W^{M2,1}{}_{SP1} \otimes W_2^{SP1}) \oplus (W^{M2,2}{}_{SP2} \otimes W_2^{SP2}) \oplus \ldots \oplus (W^{M2,q}{}_{SPq} \otimes W_2^{SPq}) \wedge \ldots \wedge (W^{Mm,1}{}_{SP1} \otimes W_m^{SP1}) \oplus (W^{Mm,2}{}_{SP2} \otimes W_m^{SP2}) \oplus \ldots \oplus (W^{Mm,q}{}_{SPq} \otimes W_m^{SPq})$$

4 Evaluation

In order to evaluate the proposed trust model, we have developed a prototype tool. We used this tool to evaluate the model in three different experiments. More specifically, in the first set of experiments we provide a comparison of the proposed opinion model with other existing models using data set from Amazon marketplace (www.amazon.co.uk). In the second and third sets of experiments, we use the example of the cloud computing scenario described in Section 2 to evaluate the use of the various parameters considered in our model. In the second set of experiments we analyze the proposed model for each individual parameter, namely (a) SLA monitoring, (b) SP ratings, and (c) SP behavior. In the third set of experiments, we analyze the model when considering combinations of the parameters in order to see if the use of more than one parameter provides better trust values.

4.1 Comparison of the Proposed Model

The dataset of Amazon marketplace used in this evaluation includes rating received by users for four sellers for a same music track CD. The seller1, seller2, seller3 and seller4 are rated by 618, 154, 422, and 314 unique users respectively. This data set contains ratings in the range of 1 to 5, for each seller, provided by the users. The rating is converted to the form <r:positive, s:negative> evidence such that r+s=1. More specifically, rating 1 maps to <0,1>, rating 2 maps to <0.25,0.75>, rating 3 maps to

<0.5,0.5>, rating 4 maps to <0.75, 0.25>, and rating 5 maps to <1,0>. A user performing the $(i+1)^{th}$ transaction has access to all the previous *i* ratings.

We compared the proposed model with Josang's [8] and Wang's [17] approaches. For all the three models, the experiment takes previous *i* ratings to predict the $(i+1)^{th}$ rating and calculates the expectation $E=b+au$ to predict the $(i+1)^{th}$ rating. The belief is calculated using the *i* previous ratings and the base rate is considered as 0.5. Fig. 2 shows the experimental results for a single seller. One time stamp on the x-axis represent 25 transactions and the y-axis represents errors that are computed as the average of 25 prediction errors based on the ratings. The results show that our model has lower prediction error when compared to Josang's [8] and Wang's [17] approaches. Table 1 summarizes the experiment performed for four sellers for the same music track CD.

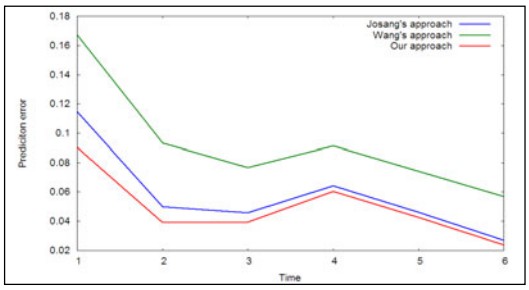

Fig. 2. Average prediction error for a Seller based on the ratings [1,5]

Table 1. Average prediction error for 4 sellers based on the ratings [1,5]

Approach	Seller1	Seller2	Seller3	Seller4
Josang's	0.10619	0.05736	0.06219	0.10809
Wang's	0.12753	0.09278	0.09415	0.14004
Our	0.10456	0.04878	0.05848	0.10449

4.2 Experiments Using Individual Parameters

SLA Monitoring. In this experiment, we consider only the SLA monitoring parameters with four resources (cpu, memory, disk, VM) associated with IP1 as fixed. We considered that the resource demand requests are sent by all SPs with incremental resources requirements. While IP1 is able to provide the demanded resources, IP1 is considered compliant with the SLA and this increases the positive evidence maintained by the SPs for IP1. At a certain point the requested resources exceed the capacity of the IP1 resulting in SLA violations. The SLA violations, add to the negative evidence maintained by the SPs for IP1. Fig. 3 shows that the reputation increases when each of the SPs have positive evidence; a maximum reputation is achieved by IP1 when each of the SPs had positive evidence of 150. After this point, the SLA violations accumulate negative evidences causing a reduction on the reputation.

SP Rating. In this experiment we considered that all the SPs used IP1 and rated IP1 for its performance based on cpu, memory, disk and virtual machine indicators. These ratings are preserved by the SPs for evaluating the IPs. The experiment starts with IP1 receiving positive ratings from each of the SPs. Each time the ratings are provided to IP1, SP1 calculates the reputation of IP1 taking into account its own ratings as well as the ratings of the other SP2 to SP5 providers. When a degraded performance is observed (i.e.; there are SLA violations), the SPs rate IP1 with negative ratings. In this experiment, the SP1's positive and negative evidence is fixed as 200 positive and 50 negative evidences. As shown in Fig. 3 the increase in the positive ratings received by SP1 from other SPs, increase the reputation until the positive evidence reaches 150. As SP1 starts receiving negative ratings from other SPs, the reputation reduces.

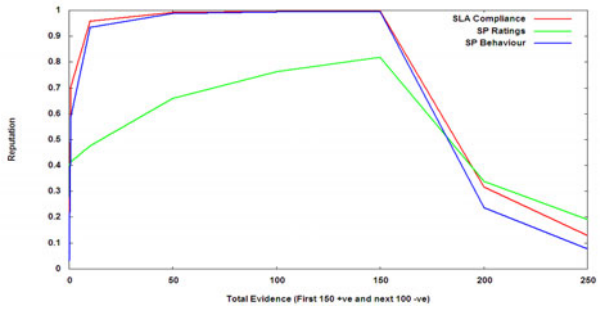

Fig. 3. Reputation based on SLA monitoring, SP Ratings and SP Behavior only

SP Behavior. In this case, the experiment begins with all SPs using only IP1 for all its resources (cpu, memory, disk space, and virtual machine). The positive behavior of all SPs increases the positive evidence for all SPs, which increases the reputation of IP1 in terms of SPs behaving towards IP1. A degraded performance observed from IP1 may lead to SPs changing their infrastructure provider. This reduces the SPs positive behavior towards IP1 and increases the negative evidence for all SPs, reducing the reputation of IP1. Fig. 3 shows the results of this experiment.

In summary, the experiments with individual parameters considered show an increase in the reputation with SLA compliance evidence for SLA monitoring, and positive SP ratings and positive SP behavior towards an IP. Also violations of SLA, negative SP rating values, and negative behavior of an SP reduces the reputation of an IP.

4.3 Experiments Using Combination of Parameters

Combination of SP Rating and SP Behavior. In this experiment, we consider IP1 with positive ratings from all the SPs. SP1 calculates the reputation of IP1 considering its own ratings as well as ratings of SP2, SP3, SP4 and SP5. The ratings provided by SP2, SP3, SP4 and SP5 are first discounted using SPs behavior towards IP1. When maintaining constant SP ratings by all SPs, the SP behavior of SP2, SP3, SP4 and SP5 changes by increasing the positive behavior of these SPs for initially zero positive behavior to a very high value. Fig. 4 (a) shows that (i) as the SP behavior becomes

more positive, the reputation of IP1 increases; (ii) when SP1 has less evidence, there is a large variation, which causes a bigger impact of the other SP behavior and as the SP1's amount of evidence increases, the reputation has less impact of SP behavior.

Combination of SP Rating and SLA Monitoring. In this experiment, to calculate the opinion of IP1 based on SP ratings, we consider all past provided SP ratings. We maintained constant opinions about IP1 and considered that the positive evidence of SLA compliance is varied from zero to a high amount of positive evidence for all SPs (SP1 to SP5). From Fig. 4 (b). it is observed that when the positive evidence from the SLA monitoring increases, the reputation of IP1 also increases.

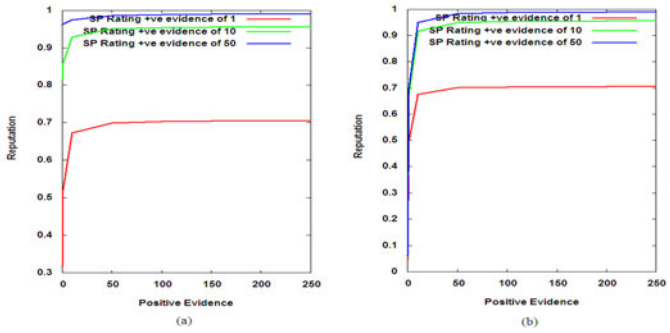

Fig. 4. Reputation based on (a) SP ratings and SP behavior, (b) SP ratings and SLA monitoring

Combination of SP Rating, Behavior and SLA Monitoring. In these experiments we calculated the reputation using all parameters. We considered the values of two of the parameters fixed and varied the third parameter, as explained below.

Effect of SP behavior. The SP rating is fixed at total of 10 positive evidences by each of the SPs. The SLA monitoring is fixed at 50 positive evidences as total evidence by each SP towards IP1. The SP behavior for SP1 to SP5 is varied from zero positive to a positive evidence of 250 in a total evidence of 250. Fig. 5 shows that with the increase in the positive evidence of SP behavior the reputation of IP1 increases.

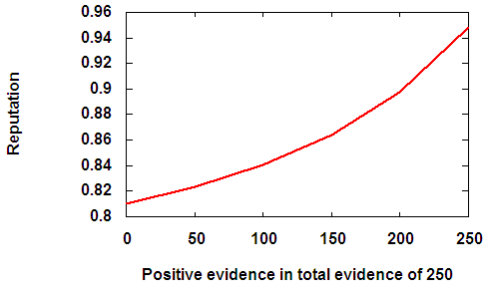

Fig. 5. Effect of SP behavior

Effect of SLA monitoring. The SP ratings provided by all SPs for IP1 and the SP behavior for all SPs are fixed. The total evidence consists of only positive evidence obtained from SLA monitoring, which is varied from zero to 250. Fig. 6(a) shows that the reputation of IP1 increases with the increase in positive evidence obtained.

The effect of SLA monitoring information is important to evaluate reputation of an IP during the operational phase. In a cloud environment, when the SPs deploy their services on a particular IP, the services are retained for significantly longer duration. This results in less frequent updates of SP ratings and SP behavior. The provision of updates of compliance/non-compliance SLA monitoring information at regular intervals may have significant impact on the reputation of an IP, as shown in Fig. 6(a).

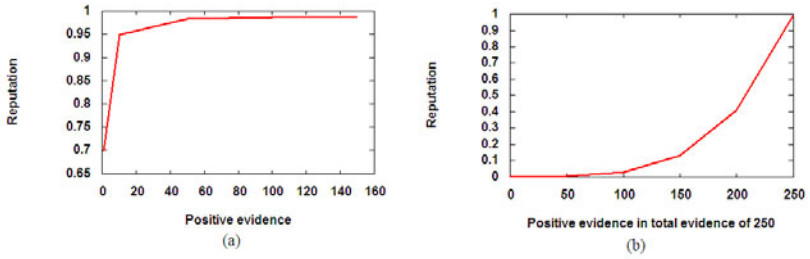

Fig. 6. (a) Effect of SLA compliance; (b) Effect of SP rating

Effect of SP ratings. The SP behavior of all SPs towards an IP and the SLA violation for an IP provided by all SPs are fixed. The positive evidence from all SPs for IP1 is varied from zero to 250 in a total evidence of 250. Fig. 6(b) shows that as the positive evidence increases and the negative evidence reduces, the reputation of IP1 increases.

5 Related Work

Trust and reputation have been the focus of research in several open systems such as e-commerce, peer-to-peer, and multi-agent systems [1] [7] [10][14]. Some trust and reputation approaches have been suggested for web-service systems [3] [4] [12][15][16]. In general, the web-services based approaches are limited [16]. For example, majority of these approaches rely on the use of a centralized repository to store and collect specific QoS feedback from consumers about a service. An exception is found in [15] that uses different QoS registries organized in a P2P way for groups of service providers, but this approach is still limited to specific quality types of feedback and requires overhead of communication due to the use of complex structures. The trust model for P2P systems in [18] considers transactions and shared experiences as recommendations and uses Bayesian estimation methods to compute trust values. The Beta reputation model in [9] is based on beta distribution that considers two parameters, positive evidence and negative evidence to estimates the reputation of an entity. Both models [18][9] are based on the belief theory, but in [18] the use of Bayesian estimation expects probabilities for each question of interest. The work in [9] has a mapping between opinion space and evidence space [8] and the opinion model allows operate with uncertain probabilities.

Trust is closely related to the concept of uncertainty. However, many of the existing reputation systems have not considered uncertainty in their work. Exceptions are found in the works described in [8][11][17]. The belief model in [8] uses metric called *opinion* to describe belief and disbelief about a proposition as well as the degree of uncertainty regarding probability of an event. The work on [17] proposes *opinion* metric as in [8] but giving importance to uncertainty due to the evidence that impacts the belief and disbelief about a proposition. In [8] the uncertainty is modeled only based on the amount of total evidence; i.e. as the total evidence increases the uncertainty decreases. In [17] the uncertainty also takes into account the amount of positive and negative evidence contained in the total evidence; i.e. given the total evidence the uncertainty is highest when the positive and negative evidence in the total evidence is equal, and the uncertainty reduces as the two evidences dominates.

In Cloud environment, trust based on reputation systems have been discussed in [5][6][2]. In [5], trust is one of the core component used by SP, along with risk, eco-efficiency and cost for evaluating the IP for their service. The work in [6] identifies several vulnerabilities in the existing cloud services provided by Google, IBM, Amazon and proposes an architecture to reinforce the security and privacy in the cloud applications. It suggests a hierarchy of P2P reputation system to protect cloud resources. However, there is no reputation model proposed [6]. Alhamad *et al*. [2] proposes a trust model for cloud computing based on the usage of SLA information. This work describes the requirements and benefits of using SLA for trust modeling in cloud environment, provides a high level architecture capturing major functionalities required, and provides a protocol for the trust model. As in [2] our model also includes SLA compliance information to model trust. We complement the trust model with SP ratings and SP behavior to assist modeling comprehensive trust aspects of an IP. Contrary to [2], we also provide a trust model to evaluate the trust of an IP.

The approach presented in this paper complements existing approaches for reputation of cloud computing environments. Different from existing works, our approach considers several parameters to calculate trustworthiness of infrastructure providers.

6 Conclusion and Final Remarks

This paper presents a trust model to support service providers to verify trustworthiness of infrastructure providers in cloud computing environments. The model calculates trust values based on different parameters, namely (i) SLA monitoring compliance, (ii) service provider ratings, and (ii) service provider behavior. The trust values are calculated based on an opinion model in terms of belief, disbelief, uncertainty and base rate. The work has been evaluated in different sets of experiments. We are currently extending the model to consider relationships that may exist between service providers and infrastructure providers, and use them as another parameter when calculating trust values. We are also performing some more experiments to evaluate the work in other scenarios.

Acknowledgement. This work has been partially supported by the EU within the 7th Framework Programme under contract ICT-257115 - Optimized Infrastructure Services (OPTIMIS). We also acknowledge Theo Dimitrakos, chief security researcher, BT, UK, for providing vital inputs towards the work in this paper.

References

1. Adler, B.T., de Alfaro, L.: A Content-driven Reputation System for Wikepedia. In: Proc. of World Wide Web Conference (2007)
2. Alhamad, M., Dillon, T., Chang, E.: SLA-Based Trust Model for Cloud Computing. In: 13th International Conference on Network-Based Information Systems (2010)
3. Artz, D., Gill, Y.: A Survey of Trust in Computer Science and the Semantic Web. Web Semantics 5(2) (2007)
4. Chang, E., Dillon, T.S., Hussain, F.K.: Trust and reputation for service-oriented environments: technologies for building business intelligence and consumer confidence. Wiley (2006)
5. Ferrer, A.J., Hernández, F., Tordsson, J., Elmroth, E., Ali-Eldin, A., Zsigri, C., Sirvent, R., Guitart, J., Badia, R.M., Djemame, K., Ziegler, W., Dimitrakos, T., Nair, S.K., Kousiouris, G., Konstanteli, K., Varvarigou, T., Hudzia, B., Kipp, A., Wesner, S., Corrales, M., Forgó, N., Sharif, T., Sheridan, C.: OPTIMIS: a Holistic Approach to Cloud Service Provisioning. Future Generation Computer Systems 28(1), 66–77 (2012)
6. Hwang, K., Kulkarni, S., Hu, Y.: Cloud Security with Virtualized Defense and Reputation-based Trust Management. In: Eighth IEEE International Conference on Dependable, Autonomic and Secure Computing (2009)
7. Josang, A., Ismail, R., Boyd, C.: A Survey of Trust and Reputation Systems for Online Service Provision. Decision Support Systems 43(2) (2007)
8. Josang, A.: A Logic for Uncertain Probabilities. International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems 9(3), 279311 (2001)
9. Josang, A., Ismail, R.: The Beta Reputation System. In: Proceedings of the 15th Bled Electronic Commerce Conference e-Reality: Constructing the e-Economy (2002)
10. Kokash, N., van den Heuvel, W.J., D'Andrea V.: Leveraging Web Services Discovery with Customizable Hybrid Matching. In: Int. Conf. on Web Services (2006)
11. Li, F., Wu, J.: Uncertainty Modeling and Reduction in MANETs. IEEE Transactions on Mobile Computing 9(7) (2010)
12. Maximillen, E.M., Singh, M. P.: Reputation and Endorsement for Web Services. SIGecom Exchanges 3(1) (2002)
13. OPTIMIS. Optimized Infrastructure Services, `http://www.optimis-project.eu`
14. Pujol, J.M., Sanguesa, R., Delgado, J.: Extracting Reputation in Multi Agent Systems by Means of Social Network Topology. In: Proc. International Joint Conference Autonomous Agents and Multiagent Systems (2002)
15. Vu, L.-H., Hauswirth, M., Aberer, K.: QoS-Based Service Selection and Ranking with Trust and Reputation Management. In: Meersman, R. (ed.) OTM 2005, Part I. LNCS, vol. 3760, pp. 466–483. Springer, Heidelberg (2005)
16. Wang, Y., Vassileva, J.: Towards Trust and Reputation Based Web Service Selection: A Survey. International Transaction Systems Science and Applications 3(2) (2007)
17. Wang, Y., Singh, M.P.: Evidence-Based Trust: A Mathematical Model Geared for Multiagent Systems. ACM Transactions on Autonomous and Adaptive Systems 5(4), Article 14 (2010)
18. Wu, P., Wu, G.: A Reputation-Based Trust Model for P2P Systems. In: International Conference on Computational Intelligence and Security (2009)