Using the Improved PROMETHEE for Selection of Trustworthy Cloud Database Servers

Jagpreet Sidhu[1,2] and Sarbjeet Singh[1]
[1]University Institute of Engineering and Technology, Panjab University, India
[2]Department of Computer Science and Engineering and Information Technology, Jaypee University of Information Technology, India

Abstract: *The adoption of cloud computing transfers control of resources to cloud service providers. This transformation gives rise to variety of security and privacy issues which results into lack of trust of Cloud Client (CC) on Cloud Service Provider (CSP). Clients need a sense of trust on service provider in order to migrate their businesses to cloud platform. In this paper, an attempt has been made to design an improved Preference Ranking Organization Method for Enrichment Evaluations (PROMETHEE) method based selection technique for choosing trustworthy Cloud Database Servers (CDSs). The selection technique utilizes multi attribute decision making approach for selecting trustworthy CDSs. The technique makes use of benchmark parameters to evaluate selection index of trustworthy CDSs. The selection index assists CCs in choosing the trustworthy CSPs. To demonstrate the proposed technique's applicability to real cloud environment, a case study based evaluation has been performed. The case study has been designed and demonstrated using real cloud data collected from Cloud Harmony Reports. This data serves as the dataset for trust evaluation and CDS selection. The results demonstrate the effectiveness of the proposed selection technique in real cloud environment.*

Keywords: *Cloud computing, cloud database servers, trust model, trustworthiness, multi attribute decision making, PROMETHEE, improved PROMETHEE, selection technique.*

Received April 11, 2015; accepted January 13, 2016

1. Introduction

A long apprehended vision of computer scientist to build computing as utility (e.g., electricity) has been achieved through cloud computing. Cloud computing as a technology has achieved its goals of being readily available, economical, robust, elastic and flexible. It provides economical access of computing and storage resources to organizations which have limited finances and enable them to access state of the art technologies over the network as services on leased basis. The resources like software, platform and infrastructure are available as services over the network to enormous audience. Although there are many definitions of cloud computing in computing domain, the most articulated one is given by National Institute of Standards and Technology (NIST) as:

"Cloud computing is a model for enabling ubiquitous, convenient, on-demand network access to a shared pool of configurable computing resources that can be rapidly provisioned and released with minimal management effort or service provider interaction" [21].

Cloud computing promise various benefits such as reduced expenses and simplicity to service providers and service requesters. For instance, it only took 24 hours at the cost of merely $240 for the New York Times to archive its 11 million articles (1851-1980) using Amazon Web Services [9, 22, 23, 30]. However, there are concerns about the trustworthiness of cloud-based services. Many potential users, such as small and medium businesses, which are increasingly realizing the business merits of cloud computing [31], are still reluctant to believe that cloud computing can offer them trustworthy and satisfactory services [6].

Cloud Client (CCs) need a sense of trust on Cloud Service Providers (CSPs) to confidentially migrate their business to cloud. As a partial solution to the problem of establishing trust, a Service Level Agreement (SLA) is signed between CC and CSP. SLA is a formal agreement to promise what is possible to provide and provide what is promised [34]. CC uses SLA as a legally binding description of what provider promised to provide. The CSP uses it to have a definite, binding record of what is to be delivered [27]. However, even with this contact formed, CCs still have lack of trust on CSPs' intentions on obeying the agreed SLA. For example, CSPs can still provide users with less CPU and memory resources than that specified and agreed in the SLA, which allows service providers to support more users to book more profits. This act of CSPs results in lack of trust. Fortunately, tools are available to allow users to monitor and verify compliance of Cloud Database Servers (CDS) as per SLA [5, 8, 11, 29]. These tools provide a monitoring on contact but do not address the problem related to selection of trustworthy CDSs for service provisioning or migration.

This paper aims to design and demonstrate a selection technique for selecting trustworthy CDSs in cloud environment. The proposed selection technique is based on improved PROMETHEE method. The rest of the paper is organized as follows: section 2 describes related work on trust evaluation problem in cloud computing. Section 3 describes proposed selection technique. Section 4 presents a case study to demonstrate the proposed selection technique. Section 5 present's results and their discussion and section 6 summarizes the work with conclusion.

2. Problem Formulation

Trustworthiness is generally associated to "levels of confidence in something or someone" [8]. Though trust is a fascinating subject and social scientists have researched into the concept and developed theories around trust [8], but there is no generally agreed definition of trust for computing. Trust definition given by "Diego Gambetta" is the most articulated definition of trust in computing discipline.

Diego Gambetta defines trust as "a particular level of subjective probability with which an agent assesses that another agent or group of agents will perform a particular action, both before he can monitor such action (or independently of his capacity ever to be able to monitor it) and in a context in which it affects his own action" [10].

Doelitzscher *et al*. [5] presents a Security Audit as Service (SAaS) architecture. The architecture aims to increase transparency between CC and cloud service provider through audits. This results in increasing trust of CC on cloud infrastructure. The model aims to conduct security audit of cloud to calculate trust in the system. Shetty [29] presents a network traffic analysis tool. As per authors, security of data in cloud depends on a secure cloud computing system and network. They used various technologies like IP geolocation, Router IP analysis and online data mining for securing cloud and its network. Liu *et al*. [20] presents Consistency as a Service (CaaS), a cloud auditing service for compliance monitoring of cloud service providers. In CaaS, a group of users constitute themselves as auditors and check the cloud provider for compliance of services promised and delivered. Gowrigolla *et al.* [13] proposed a mechanism to maintain privacy and security measures in cloud. This mechanism allows data to be encrypted on cloud without loss of accessibility or functionality for authorized parties. Guo *et al*. [14] proposed ETEC, which considers direct (time-variant) and recommendation (space-variant) trust with comprehensive evaluation. The model provide a helpful measure to enhance the robustness, fault tolerance and security of cloud computing. Chaowen *et al*. [4] proposed a trust model like Beth *et al.* [1], Jøsang [16], Jøsang *et al.* [17], and Kamvar *et al.* [18] to evaluate trust degree by history of interactions and reputation of trustee's outer information. Trusted Computing Group (TCG) helps in evaluating trustees inner attributes which are combined with reputation from historical interactions [4]. Ko *et al*. [19] proposed a detective trust framework which shifted its concerns to integrity and accountability of data stored in cloud. Authors aimed at building a single point of view for accountability of cloud Database servers [19]. Wang *et al*. [33] proposed an Audit-based trustworthiness verification scheme for monitoring the integrity of cloud servers. The main contribution of this novel architecture is to monitor the trustworthiness of a "large" public cloud by a TTP deployed on a "small" private cloud. Zhang *et al*. [35] proposed a novel Multiple-Level TRUST (ML-TRUST) management framework for wireless sensor networks. Filali and Yagoubi [7] presented a trust management framework, focusing on provider selection problem in cloud environment. Performance metrics to select most suitable service providers are investigated. Chandran *et al*. [3] designed a fuzzy-logic based trust and reputation model for cloud. Ghosh *et al*. [12] proposed SelCSP, a framework for selection of trustworthy and competent service provider.

Though there are many efforts to address trust issues between cloud service provider and CCs by providing data, storage and network security but no efforts have been made in evaluating trustworthiness of CDSs as a multi attribute decision making problem based on benchmark parameters. This work is an attempt in this direction. There is a need to evaluate trustworthiness of CDSs based on monitoring results of parameters agreed in SLA. Next section introduces trustworthiness evaluation as a Multiple Criteria Decision Making (MCDM) problem and presents the design of PROMETHEE method based selection technique for selecting trustworthy service providers.

3. PROMETHEE Method Based Service Selection Technique

Relative ranking of CDSs based on benchmark parameters and features offered by them is an important piece of information which assists CCs in choosing best service providers as per their requirements. But the problem of ranking CDSs is a complex decision making problem. There are numerous benchmark parameters and sub-parameters which make evaluation of relative ranking complex. A solution is needed for this complex selection problem involving multi parameters or attributes.

In general, such problems fall into the category of Multiple Attribute Decision Making (MADM), where decision makers choose or rank alternatives on the basis of evaluation of several attribute. Decision making involves managing trade-offs or compromises among a number of criteria that are in conflict with each other. The solution to relative ranking on multiple criteria is defined as MCDM [25]. Without a structured technique, the evaluation of trustworthiness of different cloud database servers would be very difficult given the number of attributes and criteria involved. In addition, the challenge is how to compare each cloud Database server based on each benchmark parameter (attribute), how to quantify them and how to aggregate them in a meaningful metric [12].

Preference Ranking Organization Method for Enrichment Evaluations (PROMETHEE) was introduced by Brans *et al*. [2] and belongs to the category of outranking methods. The PROMETHEE method is improved by incorporating AHP method for deciding the attributes' weights [25, 26].

The improved PROMETHEE method involves a pairwise comparison of alternatives on each single attribute in order to determine partial binary relations denoting the strength of preference of an alternative 'a' over alternative 'b'. In the evaluation table, the alternatives are evaluated on different attributes. These evaluations involve mainly quantitative data. The implementation of improved PROMETHEE requires additional types of information, namely:

- Information on the relative importance that is the weights of the attributes considered.
- Information on the decision maker preference function, which he/she uses when comparing the contribution of the alternatives in terms of each separate attribute [25].

The procedure for selection technique based on improved PROMETHEE method for selection of trustworthy CDSs among numerous CDSs is described below [32]:

- *Step* 1: Let CDS={cds_1, $cds_{2,\ldots\ldots,}cds_m$} be a set of n CDS and let BP={bp_1, $bp_{2,\ldots\ldots,}$ bp_n} be a consistent family of m benchmark criteria. The basic data related to such a problem can be written in a table containing n^*m evaluations. Each line corresponds to a CDS and each column corresponds to a benchmark criterion.

$$CDS_{mxn} = \begin{array}{c} \\ CDS_1 \\ CDS_2 \\ CDS_3 \\ \vdots \\ CDS_{m-1} \\ CDS_m \\ \\ \end{array} \begin{bmatrix} B_{P_1} & B_{P_2} & B_{P_3} & \ldots & B_{P_{n-1}} & B_{P_n} \\ B_{P_{11}} & B_{P_{12}} & B_{P_{13}} & \ldots & B_{P_{1(n-1)}} & B_{P_{1n}} \\ B_{P_{21}} & B_{P_{22}} & B_{P_{23}} & \ldots & B_{P_{2(n-1)}} & B_{P_{2n}} \\ B_{P_{31}} & B_{P_{32}} & B_{P_{33}} & \ldots & B_{P_{3(n-1)}} & B_{P_{3n}} \\ \vdots & \vdots & \vdots & & \vdots & \vdots \\ B_{P_{(m-1)1}} & B_{P_{(m-1)2}} & B_{P_{(m-1)3}} & \ldots & B_{P_{(m-1)(n-1)}} & B_{P_{(m-1)n}} \\ B_{P_{m1}} & B_{P_{m2}} & B_{P_{m3}} & \ldots & B_{P_{m(n-1)}} & B_{P_{mn}} \end{bmatrix}_{mxn}$$

- *Step* 2: Calculate preference degree, let bp_j (a) be the value of a criterion j for a CDS_a. We note d_j (CDS_a, CDS_b), the difference of value of a criterion j for two decisions a and b.

$$d_j(CDS_a, CDS_b) = bp_j(CDS_a) - bp_j(CDS_b) \quad (1)$$

P_j (CDS_a, CDS_b) is the value of the preference degree of a criterion j for two decisions *CDSa* and *CDSb*. The preference functions used to compute these preference degrees are defined such as:

$$p_j(CDS_a, CDS_b) = F(d_j(CDS_a, CDS_b)\ with\ \forall x \in\ ]-\infty, \infty[\ , 0 \leq F(x) \leq 1 \quad (2)$$

- *Step* 3: In the PROMETHEE method suggested by Brans *et al*. [2], there is no systematic way to assign weights of relative importance of attributes. Hence, in the improved PROMETHEE method AHP method [32] is suggested for deciding the weights (Wm) of relative importance of the attributes [28].
- *Step* 4: Calculate the global preference index by aggregating the preference degrees of all criteria for each pair. Let *C* be the set of considered criteria and W_j the weight associated to the criterion *j*. The global preference index for a pair of possible decision a and b is computed as follows:

$$\Pi\ (CDS_a, CDS_b) = \textstyle\sum_{j\in BP} W_j * P_j(CDS_a, CDS_b) \quad (3)$$

- *Step* 5: For each possible decision a, we compute the positive outranking flow $\emptyset^+(\mathrm{CDSa})$ and the negative outranking flow $\emptyset^-(\mathrm{CDSa})$. Let A be the set of possible decisions and m the number of possible decisions. The positive outranking flow of a possible decision *a* is computed by the following Equation:

$$\emptyset^+(CDS_a) = \textstyle\sum_{x\epsilon A} \pi(CDS_a, x) \quad (4)$$

The negative outranking flow of a possible decision *a* is computed by the following Equation:

$$\emptyset^-(CDS_a a) = \textstyle\sum_{x\epsilon A} \pi(CDS_a, x) \quad (5)$$

- *Step* 6: Establish a Trustworthy selection index between the possible CDS. The ranking is based on the net outranking flows (trustworthy index). The net outranking flow $\emptyset(CDS_a)$ of a possible decision CDS_a is computed as follows:

$$\emptyset(CDS_a) = \emptyset^+(CDS_a) - \emptyset^-(CDS_a) \quad (6)$$

The higher the value of the *CDS* for a decision, the better trustworthy the *CDS* is. In our trustworthy *CDS* selection context where we are only interested in the best decision to make, we will choose the decision that maximises the net outranking flows.

4. Case Study: Selection of Trustworthy CDSs Based on Benchmark Parameters

The case study has been demonstrated using a sample dataset extracted from Cloud Harmony benchmarks report on CDSs. Cloud Harmony report examines how to use benchmarks to compare performance among cloud database services. CDS covered in this report include Amazon Web Services (AWS), DigitalOcean, Google Cloud Platform, Microsoft Azure, Rackspace Cloud, and SoftLayer. The purpose of this report is to help decision makers decipher often competing claims of performance superiority from vendors and reviewers, and to provide meaningful insight about cloud performance [15]. The sample dataset consists of performance of 18 CDSs on 10 benchmark parameters. The Benchmark parameters involved are SPECint (SP_{int}), $SPEC_{fp}$ (SP_{fp}), Memory Performance on Scale (MP_{sc}), Memory Performance on Triad (MP_{td}), Sequential Read/Write Disk Performance (SRW_{dp}), Random Read/Write Disk Performance (RRW_{dp}), Sequential Disk Read/Write Performance Consistency (SRW_{pc}), Random Disk Read/Write Performance Consistency (RRW_{pc}), Network Latency (N_l) and Cost On Demand (C_{od}). Relative weights for these parameters were evaluated by employing AHP method [32]. AHP method finds the relative importance among various Benchmark parameters [24, 28]. From this relative importance, it evaluates the relative weight for each benchmark parameter.

During the case study we presume that Benchmark parameters viz. SPECint (SP_{int}), SPECfp (SP_{fp}), Memory Performance on Scale (MP_{sc}), Memory Performance on Triad (MP_{td}), Sequential Read/Write Disk Performance (SRW_{dp}), Random Read/Write Disk Performance (RRW_{dp}), Sequential Disk Read/Write Performance Consistency (SRW_{pc}), Random Disk Read/Write Performance Consistency (RRW_{pc}), Network Latency (N_l) and Cost On Demand (C_{od}) as beneficial attributes. As Cloud Harmony performance report [15] is benchmark testing report which comprises of performance monitored by third party on a positive scale. From this dataset, the Decision Matrix shown in Table 1 is obtained. This is 18x10 matrix representing 18 CDSs (database servers) and 10 parameters.

Relative weights for benchmark parameters are evaluated by measuring the relative importance among parameters [24, 28]. A pair-wise comparison matrix RI_{10*10} using a scale of relative importance among various benchmark parameters is shown in Table 2.

Table 1. Decision Matrix CDS_{18*10}.

CDS	SP_{int}	SP_{fp}	MP_{sc}	MP_{td}	SRW_{dp}	RRW_{dp}	SRW_{pc}	RRW_{pc}	N_l	C_{od}
Amazon EC2 (S)	81.4300	79.8800	12.8200	13.1600	59342.3300	39039.6700	0.2789	0.2573	0.1500	0.2100
DigitalOcean (S)	65.4600	81.2400	9.0000	9.8800	35334.0000	35333.6700	0.1822	0.1846	0.4500	0.1190
Google (S)	69.3700	70.6700	13.7000	13.4900	5620.0000	4290.0000	0.3316	0.2706	0.6500	0.2800
Microsoft Azure (S)	45.6000	42.3200	5.4800	5.5200	63526.3300	63493.3300	0.2775	0.2681	0.5900	0.2400
Rackspace (S)	106.2600	113.7300	10.7500	10.4300	185523.3300	78577.0000	0.3934	0.3553	0.4200	0.6800
SoftLayer (S)	77.1100	86.1600	10.4500	9.7800	110998.6700	97954.3300	0.2639	0.3513	0.1000	0.2360
Amazon EC2 (M)	154.1900	147.3900	13.0000	12.8900	132608.0000	37042.3300	0.2497	0.2639	0.1400	0.4200
DigitalOcean (M)	136.9500	141.3000	9.3100	9.8700	44756.6700	44768.6700	0.1691	0.1728	0.3900	0.2380
Google (M)	133.7500	126.0600	13.4800	13.6600	11182.6700	8579.6700	0.3339	0.3206	0.6600	0.5600
Microsoft Azure (M)	75.8200	80.4800	2.2100	2.1900	63373.3300	63188.6700	0.3899	0.2736	0.6000	0.4800
Rackspace (M)	182.8900	177.1000	10.9300	10.6200	269966.0000	331863.6700	0.3066	0.3519	0.2900	1.3600
SoftLayer (M)	148.7300	153.9900	10.7900	10.0800	27792.6700	13275.0000	0.4154	0.3588	0.1200	0.4380
Amazon EC2 (L)	277.5600	230.9200	12.9100	12.9200	66685.3300	35778.0000	0.2479	0.2566	0.1100	0.8400
DigitalOcean (L)	184.2400	195.5200	9.1900	10.0500	59677.6700	59691.6700	0.1473	0.1496	0.3600	4.4700
Google (L)	225.1400	184.4500	11.9200	12.6800	22177.0000	17266.0000	0.2915	0.3261	0.6600	0.7050
Microsoft Azure (L)	363.0000	286.6300	11.6200	11.1600	63373.3300	63188.6700	0.3899	0.2736	0.6000	1.1200
Rackspace (L)	315.1900	272.6500	10.9000	10.5100	119773.3300	55184.0000	0.4012	0.4036	0.2300	2.7200
SoftLayer (L)	280.8900	259.5900	9.5700	9.4300	111194.3300	107788.0000	0.2602	0.2175	0.1300	0.7940

Table 2. Relative importance matrix RI_{10*10}.

Parameters	SP_{int}	SP_{fp}	MP_{sc}	MP_{td}	SRW_{dp}	RRW_{dp}	SRW_{pc}	RRW_{pc}	N_l	C_{od}
SP_{int}	1.00	1.00	1.29	1.29	1.50	1.50	1.50	1.50	1.13	1.13
SP_{fp}	1.00	1.00	1.29	1.29	1.50	1.50	1.50	1.50	1.13	1.13
MP_{sc}	0.78	0.78	1.00	1.00	1.17	1.17	1.17	1.17	0.88	0.88
MP_{td}	0.78	0.78	1.00	1.00	1.17	1.17	1.17	1.17	0.88	0.88
SRW_{dp}	0.67	0.67	0.86	0.86	1.00	1.00	1.00	1.00	0.75	0.75
RRW_{dp}	0.67	0.67	0.86	0.86	1.00	1.00	1.00	1.00	0.75	0.75
SRW_{pc}	0.67	0.67	0.86	0.86	1.00	1.00	1.00	1.00	0.75	0.75
RRW_{pc}	0.67	0.67	0.86	0.86	1.00	1.00	1.00	1.00	0.75	0.75
N_l	0.89	0.89	1.14	1.14	1.33	1.33	1.33	1.33	1.00	1.00
C_{od}	0.89	0.89	1.14	1.14	1.33	1.33	1.33	1.33	1.00	1.00

Table 3. Positive outranking flow $\emptyset^+$.

CDS	$\emptyset^+$
Amazon EC2 (S)	6.430556
DigitalOcean (S)	3.527778
Google (S)	7.388889
Microsoft Azure (S)	5.055556
Rackspace (S)	10.18056
SoftLayer (S)	6.055556
Amazon EC2 (M)	8.791667
DigitalOcean (M)	5.361111
Google (M)	9.819444
Microsoft Azure (M)	6.5
Rackspace (M)	11.97222
SoftLayer (M)	8
Amazon EC2 (L)	9.694444
DigitalOcean (L)	8.194444
Google (L)	10.625
Microsoft Azure (L)	12.80556
Rackspace (L)	12.86111
SoftLayer (L)	9.180556

Table 4. Negative outranking flow $\emptyset^-$.

CDS	$\emptyset^-$
Amazon EC2 (S)	10.56944444
DigitalOcean (S)	13.47222222
Google (S)	9.611111111
Microsoft Azure (S)	11.94444444
Rackspace (S)	6.819444444
SoftLayer (S)	10.94444444
Amazon EC2 (M)	8.208333333
DigitalOcean (M)	11.63888889
Google (M)	7.069444444
Microsoft Azure (M)	10.05555556
Rackspace (M)	5.027777778
SoftLayer (M)	9
Amazon EC2 (L)	7.305555556
DigitalOcean (L)	8.805555556
Google (L)	6.263888889
Microsoft Azure (L)	3.75
Rackspace (L)	4.138888889
SoftLayer (L)	7.819444444

The Weight (W_j) assessed from matrix RI_{10*10} for benchmark parameters are: SP_{int}=0.1250, SP_{fp} =0.1250, MP_{sc} =0.0972, MP_{td} =0.0972, SRW_{dp} = 0.0833, RRW_{dp} =0.0833, SRW_{pc} =0.0833, RRW_{pc} =0.0833, N_l =0.1111and C_{od} =0.1111.

The positive outranking flow $\emptyset^+$ for each CDS is obtained which is shown in Table 3.

The negative outranking flow $\emptyset^-$ for each CDS is obtained which is shown in Table 4.

The net outranking flows $\emptyset$ of possible CDSs are computed in Table 5:

Table 5. Net outranking flow $\emptyset$.

CDS	$\emptyset$
Amazon EC2 (S)	-4.13889
DigitalOcean (S)	-9.94444
Google (S)	-2.22222
Microsoft Azure (S)	-6.88889
Rackspace (S)	3.361111
SoftLayer (S)	-4.88889
Amazon EC2 (M)	0.583333
DigitalOcean (M)	-6.27778
Google (M)	2.75
Microsoft Azure (M)	-3.55556
Rackspace (M)	6.944444
SoftLayer (M)	-1
Amazon EC2 (L)	2.388889
DigitalOcean (L)	-0.61111
Google (L)	4.361111
Microsoft Azure (L)	9.055556
Rackspace (L)	8.722222
SoftLayer (L)	1.361111

Table 6. Trustworthy selection index arranged in descending order.

CDS	$\emptyset$
Microsoft Azure (L)	9.055555556
Rackspace (L)	8.722222222
Rackspace (M)	6.944444444
Google (L)	4.361111111
Rackspace (S)	3.361111111
Google (M)	2.75
Amazon EC2 (L)	2.388888889
SoftLayer (L)	1.361111111
Amazon EC2 (M)	0.583333333
DigitalOcean (L)	-0.611111111
SoftLayer (M)	-1
Google (S)	-2.222222222
Microsoft Azure (M)	-3.555555556
Amazon EC2(S)	-4.138888889
SoftLayer (S)	-4.888888889
DigitalOcean (M)	-6.277777778
Microsoft Azure (S)	-6.888888889
DigitalOcean (S)	-9.944444444

The selection index is evaluated on a scale ranging from-10 to 10. Value 10 indicates significant trustworthiness on CDS and -10 indicates insignificant trustworthiness on CDS.

To determine the selection sequence of CDSs by trustworthy selection index, CDSs are arranged in descending order of score. This is shown in Table 6 as: CDS (Microsoft Azure (L),Rackspace (L), Rackspace(M), Google(L), Rackspace(S), Google (M), Amazon EC2(L), SoftLayer(L), Amazon EC2(M), DigitalOcean(L), SoftLayer(M), Google(S), Microsoft Azure(M), Amazon EC2(S), SoftLayer(S), DigitalOcean(M), Microsoft Azure(S) and DigitalOcean(S)).

5. Results and Discussion

During literature survey, a standard report Cloud harmony for performance evaluation and service selection was investigated. This report examines how to use benchmarks to compare performance of cloud computing services [15]. This report consists of results of benchmarks on cloud database servers. Benchmarks have been used for measuring and comparing system performance for decades and used correctly, are also useful for evaluating cloud performance. The benchmarks selected during test are reputable and relevant to real workloads. The benchmarks results on CDS are collected on many benchmark parameters. The proposed technique has been demonstrated by extracting a sample dataset from cloud harmony report [15]. We choose three database server instances from each provider, based on the specifications as illustrated in Table 7.

Table 7. Cloud Database Server instance specifications.

Database Server	Type CPU Cores	Memory	Storage
Small (S)	4	8-16 GB	50 GB Data Volume
Medium (M)	8	16-32 GB	100 GB Data Volume
Large (L)	16	32-64 GB	200 GB Data Volume

The proposed technique has been simulated on 18 CDSs. To validate the proposed selection technique, a case study is performed on the sample dataset as illustrated in section 4.

Figure 1 illustrates CDSs with benchmark parameter from sample dataset. Figure 1 provides a comprehensive illustration of benchmark parameters with their benchmark scores employed during the current study. It also shows an illustrative comparison among various CDSs and their benchmark parameters.

Figure 2 illustrates relative importance among benchmark parameters evaluated by AHP method.

From this illustration it is clear that relative importance of Benchmark parameter SPECint (SP_{int}), compared with other benchmark parameters (SP_{fp}, MP_{sc}, MP_{td}, SRW_{dp}, RRW, SRW_{pc}, RRW_{pc}, N_l and C_{od}) is (1.00, 1.29, 1.29, 1.50, 1.50, 1.50, 1.50, 1.13 and 1.13). The relative importance of benchmark parameter Network Latency (N_l) compared with other benchmark parameters (SP_{int}, SP_{fp}, MP_{sc}, MP_{td}, SRW_{dp}, RRW, SRW_{pc}, RRW_{pc} and C_{od}) is (0.89, 0.89, 1.14, 1.14, 1.33, 1.33, 1.33, 1.33 and 1.00).

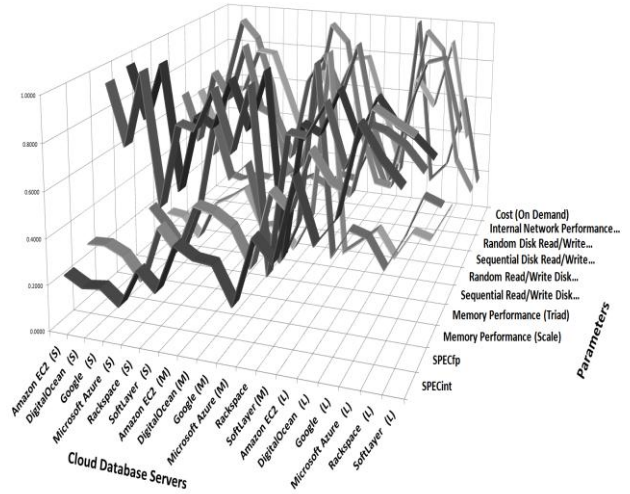

Figure 1. Cloud database servers result on benchmark parameters (Normalized).

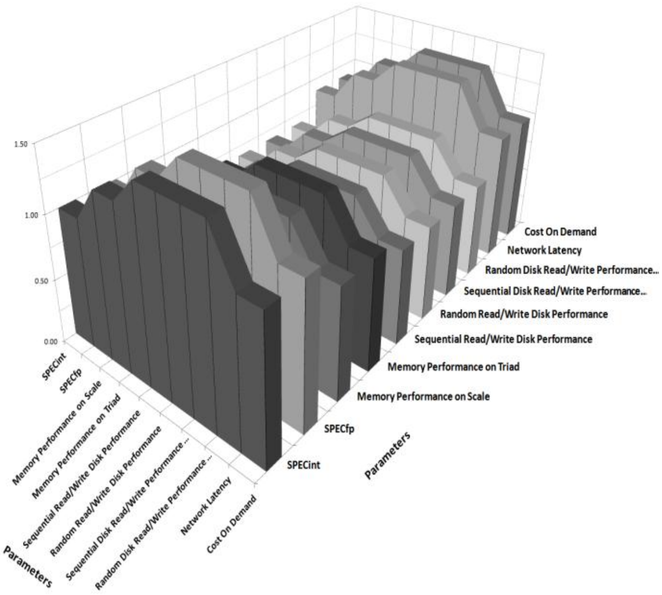

Figure 2. Relative importance of benchmark parameters.

Figure 3 illustrates the weights evaluated for various benchmark parameters. SPECint (SP_{int}) and SPECfp (SP_{fp}) have the maximum weight (SP_{int}= SP_{fp}=0.1250). In comparison, Sequential Read/Write Disk Performance (SRW_{dp}), Random Read/Write Disk Performance (RRW_{dp}), Sequential Disk Read/Write Performance Consistency (SRW_{pc}), Random Disk Read/Write Performance Consistency (RRW_{pc}) have the minimum weight (SRW_{dp}=RRW_{dp}=SRW_{pc}=RRW_{pc}=0.0833). In other words SP_{int} and SP_{fp} are considered to be most significant in contrast to SRW_{dp}, RRW_{dp}, $SRW_{pc,}$ and RW_{pc} which are considered least significant benchmark parameters in current study.

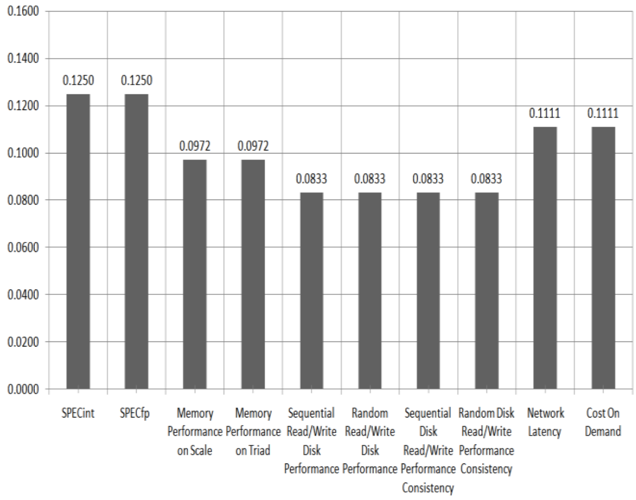

Figure 3. Relative normalized weights on benchmark parameters.

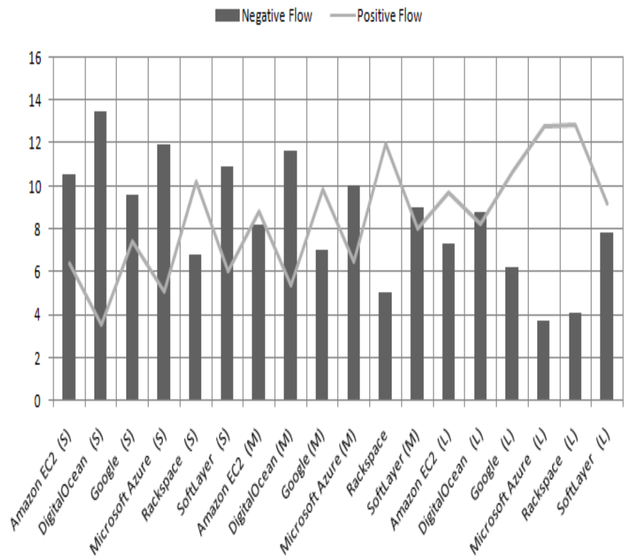

Figure 4. Positive and negative outranking flow.

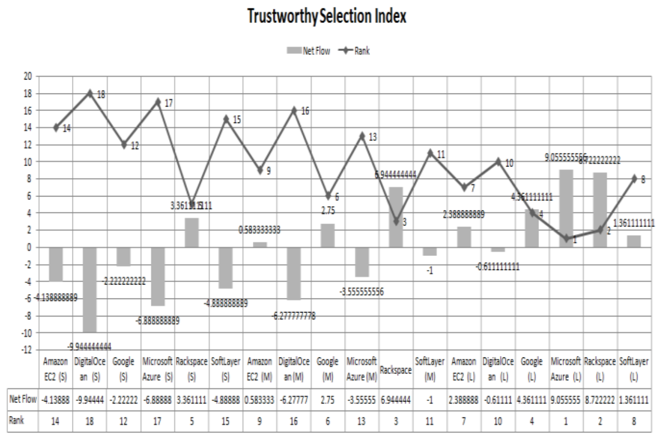

Figure 5. Net flow (CDSs trustworthy selection index).

Figure 4 illustrates the positive outranking flow and negative outranking flow as evaluated by improved PROMETHEE method as per benchmark data shown in Figure 1 and weights calculated in Figure 3. The positive outflow quantifies how a given CDS is globally preferred to all the other CDSs and the negative outflow quantifies how a given CDS is being globally un-preferred by all the other CDSs.

Figure 5 illustrates CDSs trustworthy selection index (Net flow) evaluated by improved PROMETHEE method as per benchmark data.

From Figure 5, it is evident that $CDS_{Microsoft\ Azure(L)}$ ($\emptyset$=9.055555556) is evaluated to be most trustworthy service provider. In contrast $CDS_{DigitalOcean(S)}$ ($\emptyset$=-9.944444444) is evaluated to be least trustworthy service provider. It also illustrates the selection ranks of CDSs based on trustworthiness evaluated by improved PROMETHEE method.

The proposed technique is quantitative and more logical compared to subjective technique which many a times lead to inaccurate results.

6. Summary and Conclusions

Trust on CSPs is the need of the hour for rapid adaptation and growth of cloud computing. CCs need to have trust on CSPs to migrate their security critical information, data and resources to cloud. Though, there have been numerous efforts to form trust between service providers and clients by providing data, storage and network security, but no efforts have been attempted on selection techniques based on PROMETHEE method. In this paper an attempt has been made to design and demonstrate a selection technique based on improved PROMETHEE method. The proposed technique generates CDS selection index based on benchmark parameters. To validate the proposed technique, a case study has been demonstrated on dataset extracted from Cloud Harmony. The proposed approach is a major step towards trustworthy CDS selection based on benchmark parameters. Results indicate that the approach is workable and can be employed for selection of trustworthy CDSs in real cloud environment.

References

[1] Beth T., Borcherding M., and Klein B., "Valuation of Trust in Open Networks," *in Proceedings of European Symposium on Research in Computer Security*, Brighton, pp. 1-18, 1994.

[2] Brans J., Vincke P., and Mareschal B., "How to Select and How to Rank Projects: The PROMETHEE Method," *European Journal of Operational Research*, vol. 24, no. 2, pp. 228-238, 1986.

[3] Chandran K., Valarmathy S., and Kirubakaran S., "Designing A Fuzzy-Logic Based Trust and Reputation Model for Secure Resource Allocation in Cloud Computing," *The International Arab Journal of Information Technology*, vol. 13, no. 1, pp. 30-37, 2016.

[4] Chaowen C., Chen L., and Yuqiao W., "A Subjective Trust Model Based on Two-Dimensional Measurement," *in Proceedings of International Conference on Computer Engineering and* Technology, Xi'an, pp. 37-41, 2009.

[5] Doelitzscher F., Fischer C., Moskal D., Reich C., Knahl M., and Clarke N., "Validating Cloud Infrastructure Changes by Cloud Audits," *in Proceedings of IEEE 8th World Congress Services*, Honolulu, pp. 377-384, 2012.

[6] Fan W., Yang S., and Pei J., "A Novel Two Stage Model for Cloud Service Trustworthiness Evaluation," *Expert Systems*, vol. 31, no. 2, pp. 136-153, 2014.

[7] Filali F. and Yagoubi B., "A General Trust Management Framework for Provider Selection in Cloud Environment," *in Proceedings of Advances in Databases and Information Systems*, Poitiers, pp. 446-457, 2015.

[8] Firdhous M., Ghazali O., and Hassan S., "Trust and Trust Management in Cloud Computing-A Survey," InterNetWorks Research Group, Technical Report, 2011.

[9] Foster I., Zhao Y., Raicu L., and Lu S., "Cloud Computing and Grid Computing 360-Degree Compared," *in Proceedings of Grid Computing Environments Workshop*, Austin, pp. 1-10, 2008.

[10] Gambetta D., *Trust: Making and Breaking of Cooperative Relations*, Blackwell, 1990.

[11] Garg S., Versteeg S., and Buyya R., "A Framework For Ranking Of Cloud Computing Services," *Future Generation Computer Systems*, vol. 29, no. 4, pp. 1012-1023, 2013.

[12] Ghosh N., Ghosh S., and Das S., "Selcsp: A Framework to Facilitate Selection of Cloud Service Providers," *IEEE Transactions on Cloud*

Computing, vol. 3, no. 1, pp. 66-79, 2015.

[13] Gowrigolla B., Sathyalakshmi S., and Masillamani R., "Design and Auditing Of Cloud Computing Security," *in Proceedings of 5th International Conference on Information and Automation for Sustainability*, Colombo, pp. 292-297, 2010.

[14] Guo Q., Sun D., Chang G., Sun L., and Wang X., "Modeling and Evaluation of Trust in Cloud Computing Environments," *in Proceedings of 3rd International Conference on Advanced Computer Control*, Harbin, pp. 112-116, 2011.

[15] https://cloudharmony.com/reports/state-of-the-cloud-compute-report, Last Visited, 2015.

[16] Jøsang A., "An Algebra for Assessing Trust in Certification Chains," *in Proceedings of the Network and Distributed Systems Security Symposium*, San Dieg, pp. 1-10, 1999.

[17] Jøsang A., Ismail R., and Boyd C., "A Survey of Trust and Reputation Systems for Online Service Provision," *Decision Support Systems*, vol. 43, no. 2, pp. 618-644, 2007.

[18] Kamvar S., Schlosser M., and Garcia-Molina H., "EigenRep: Reputation Management in Peer-To-Peer Networks," *in Proceedings of 12th International World Wide Web Conference*, Hungary, 2003.

[19] Ko R., Jagadpramana P., Mowbray M., Pearson S., Kirchberg M., Liang Q., and Lee B., "TrustCloud: A Framework for Accountability and Trust in Cloud Computing," *in Proceedings of IEEE World Congress Services*, Washington, pp. 584-588, 2011.

[20] Liu Q., Wang G., and Wu J., "Consistency as a Service: Auditing Cloud Consistency," *IEEE Transactions on Network and Service Management*, vol. 11, no. 1, pp. 25-35, 2014.

[21] Mell P. and Grance T., "The NIST Definition of Cloud Computing," *Communications of the ACM*, vol. 53, no. 6, 2011.

[22] Noor T. and Sheng Q., "Credibility-Based Trust Management for Services in Cloud Environments," *in Proceedings of Service-Oriented Computing*, Berlin, pp. 328-343, 2011.

[23] Noor T., Sheng Q., Zeadally S., and Yu J., "Trust Management of Services in Cloud Environments: Obstacles and Solutions," *ACM Computing Surveys*, vol. 46, no. 1, 2013.

[24] Ramanathan R., "A Note on the Use of the Analytic Hierarchy Process For Environmental Impact Assessment," *Journal of Environmental Management*, vol. 63, no. 1, pp. 27-35, 2001.

[25] Rao R. and Patel B., "Decision Making In The Manufacturing Environment Using An Improved PROMETHEE Method," *International Journal of Production Research*, vol. 48, no. 16, pp. 4665-4682, 2010.

[26] Rao R., *Decision Making In The Manufacturing Environment: Using Graph Theory And Fuzzy Multiple Attribute Decision Making Methods*, Springer Science and Business Media, 2007.

[27] Ron S. and Aliko P., "Service Level Agreements," *Internet NG Project*, 2001.

[28] Saaty T., *Theory and Applications of the Analytic Network Process: Decision Making With Benefits, Opportunities, Costs, and Risks*, RWS Publications, 2005.

[29] Shetty S., "Auditing and Analysis of Network Traffic in Cloud Environment," *in Proceedings of IEEE 9th World Congress Services*, Santa Clara, pp. 260-267, 2013.

[30] Sotomayor B., Montero R., Liorente I., and Foster I., "Virtual Infrastructure Management in Private and Hybrid Clouds," *IEEE Internet Computing*, vol. 13, no. 5, pp. 14-22, 2009.

[31] Subashini S. and Kavitha V., "A Survey on Security Issues in Service Delivery Models of Cloud Computing," *Journal of Network and Computer Applications*, vol. 34, no. 1, pp. 1-11, 2011.

[32] Taillandier P. and Stinckwich S., "Using the PROMETHEE Multi-Criteria Decision Making Method To Define New Exploration Strategies For Rescue Robots," *in Proceedings of International Symposium on Safety Security and Rescue Robotics*, Kyoto, pp. 321-326, 2011.

[33] Wang C., Wang Y., Liu C., and Wang X., "An Audit-based Trustworthiness Verification Scheme for Monitoring the Integrity of Cloud Servers," *Journal of Computational Information Systems*, vol. 10, no. 23, pp. 9923-9937, 2014.

[34] Wu L. and Buyya R., "Service Level Agreement (SLA) In Utility Computing Systems," *Performance and Dependability in Service Computing: Concepts, Techniques and Research Directions*, vol. 1, pp. 1-25, 2010.

[35] Zhang B., Huang Z., and Xiang Y., "A Novel Multiple-Level Trust Management Framework for Wireless Sensor Networks," *Computer Networks*, vol. 72, pp. 45-61, 2014.

Jagpreet Sidhu received his Masters and Ph.D degree from Panjab University, India. He is currently working as Assistant Professor in Jaypee University of Information Technology, Waknaghat, India. He has published many papers in international journals and conferences. His research interests include cloud computing and trust issues in distributed systems.

Sarbjeet Singh is currently working as Professor in UIET, Panjab University, India. He received his Masters and Ph.D degree from Thapar University, India. His research interests include security and trust related issues in cloud computing. He is Life Member of Computer Society of India and Indian Society for Technical Education.