

Analysis of Trust Score of CSPS by Comparing Service Broker Policies and Load Balancing Policies using Cloud Analyst and Fuzzy Inference System

Prabu Ragavendiran. S. D
M.E., (Phd.,)
Associate Professor,
Department Of CSE,
Builders Engineering College,
Nathakadaiyur, Kangayam,
Tirupur, Tamilnadu.

Sowmiya N
M.E.,
Assistant Professor,
Department Of CSE,
Builders Engineering College,
Nathakadaiyur, Kangayam,
Tirupur, Tamilnadu.

Santhiya P,
M.E CSE – PG Student,
Department Of CSE,
Builders Engineering College,
Nathakadaiyur, Kangayam,
Tirupur, Tamilnadu.

Abstract:- Trust is one of the main component in cloud computing, when it comes to relationship with customers and service providers. To minimize the security risk and malicious attacks, trust in the cloud between cloud providers and users is very important. To motivate the awareness about the ranking of different Cloud Service providers, a model is proposed to compute the trust component of different Cloud Service Providers. In cloud computing, Fuzzy logic is an effective technique to evaluate the trustworthiness of service providers. This trust computing model is derived from CSMIC 's Service Measurement model. In this model three Service Broker Policies and three Load Balancing policies are considered to compute the trust score of cloud Service providers. The real cloud environment is simulated through Cloud Analyst, a visual GUI tool which runs on Cloud Sim. The major simulation parameters which reflect key performance indicators are input into fuzzy inference system (FIS) to evaluate the trust score which in turn used to rank the cloud service providers. Nine different categories are analysed based on three service broker policies and the three Load Balancing policies. Based on the trust score ranking of cloud service providers for all these nine categories are studied.

Key words: Cloud Service Providers, CSMIC, Cloud Analyst, FIS, Service Broker policy, Load Balancing.

I. INTRODUCTION

Cloud computing has gained tremendous momentum in past few years as the use of computers in our day-to-day life has increased. Some of the inspiring characteristics of cloud computing such as cost economics, instant scalability are the main reasons which this rapid growth. The success of cloud computing depends on how the cloud users believe their service providers. The integrity, security and reliability of their data have much influence among the users to select a service provider. The cost and availability also play a vital role in electing a provider.

In order to ensure the satisfaction of cloud consumers, all the above factors must be supported in an integrated manner and one of the classical measures used to represent this is Trust. Trust is a virtual entity, rather than a physical entity which may be approximated and assessed by

various means. In our cloud environment we are in a position to ensure the necessary key performance attributes expected from a provider for a selected cloud service are properly addressed and this idea paved a way to assess the trust score for every service provider using the fulfilment of quality of above attributes.

In order to evaluate the trust score of the provider for ranking purpose, the performance indicators are taken into account based on the Key Performance Indicator attributes with reference to Service Measurement Index(SMI) of Cloud Service Measurement Index Consortium (CSMIC). The CSMIC is a consortium which define the QoS attributes in the specified framework with a method for calculating a relative index, which is further used for comparing various cloud services and providers.

The evaluation of the trust ranking for service providers is carried out in two phases. In phase one KPIs are grouped under Performance, Cost, Agility, Time and Security are obtained as inputs, derived from a simulation environment using Cloud Analyst, a visual tool which runs on Cloud Sim. In second stage these obtained parameter values are again input to fuzzy inference system in the next level to obtain the trust values of providers. Based on the trust values, providers are ranked. Mamdani FIS is used in the second phase.

The same procedure is applied for obtaining ranks by using the load balancing process in cloud computing in two different levels. In the first level which is presented by CloudAppServiceBroker in Cloud Analyst simulator, a model of service brokers handles traffic routing between user bases and datacenters. The three default and common routing policies which are provided in Cloud Analyst simulator are: "Closest Datacenter", "Optimize Response Time" and "Reconfigure Dynamic with Load". The second level which is introduced in CloudAnalyst by VMLoadBalancer component is responsible for modeling the load balance policy used by datacenters when serving allocation requests. There are three usual "Round Robin", "Throttled" and "Equally Spared Current Execution Load"

load balancing algorithms in each datacenter provided by simulator. By different combination of these three VM load balancing algorithms and datacenter broker Policies, nine different results are available which will be analyzed in the rest of this paper based on different evaluation parameters such as Performance, Cost, Agility, Time and security.

The cost, time and trust value varies for service providers in above different schemes and reflect in different order of ranking. The simulation was done using Cloud Analyst tool of Cloud Sim and also the fuzzy inference system is used to capture the output of Simulation environment as inputs and provide trust score at the end, which is used for ranking the providers.

II. RELATED WORKS

A work published by Florian Skotplik, et al [1] referred how fuzzy logic was used to provide trust in cloud computing in fuzzy logic system. A paper published by XiaodongSun, et al [2] explains direct and recommended trust measurements based on fuzzy set theory. The model is very useful in improve robustness, fault tolerance and security in cloud services.

A publication from Garg et al [3] on "SMI Cloud" proposed the skeleton to measure the performance of CSPs and rank them, which lead to healthy technical and business competition among providers to fulfil their Quality of Service (QoS) and satisfy Service level Agreements. The proposed idea systematically evaluates all quality attributes of a cloud service as proposed by CSMIC and rank them based on those attributes.

The model proposed by Supriya M. Venkataramana, et al [5] uses fuzzy logic inference system to evaluate cloud providers through parameters like Performance, Security, Elasticity, Time and Cost. Hamdy M. Mousa, et al [6] expanded the above work by using Mamdani fuzzy inference engine at all levels of fuzzy system and added new key performance indicators in computing parameters and improved the efficiency.

A work done by Sonia Lamba , Dharmendra Kumar[6] mainly focus on comparing Load Balancing policies and Service Broker policies and reducing overhead, reducing the migration time and improving performance etc. The several strategies not efficient for scheduling and load balancing resource allocation techniques leading to increased operational cost.

Our proposed work extends the above works by adding more KPIs towards computing parameters and also we test the entire comparison scenario for 3 different service broker policy and 3 different Load Balancing policies in order to view the changes in trust scores and ranks accordingly.

III. ABOUT CSMIC

The Cloud Services Measurement Initiative Consortium (CSMIC) developed to address the need for industry-wide, globally accepted measures for calculating the benefits and risks of cloud-computing services.

In order to measure the cloud services CSMIC is a group which has development kind of measurement framework called SMI. This mainly focuses on

characteristics like customer's security and privacy, performance, assistance, financial, usability and accountability.

The drive to develop the consortium was prompted by a desire to help develop industry standards for measurement of services, and for innovative and multidisciplinary problem-solving skills to tackle industry challenges.

The major product of the consortium's efforts is the Service Measurement Index (SMI), a set of business-relevant Key Performance Indicators (KPI's) that provide a standardized method for measuring and comparing a business service. From procurement and ongoing service levels, to business viability and security, the SMI framework will provide a holistic view into the entire customer experience for cloud service providers in these primary areas: Accountability, Agility, Assurance of Service, Cost, Security and Privacy, and Usability (functionality and performance).

The SMI is a hierarchical framework. The top level divides the measurement space into 7Categories.EachCategory is further refined by 4or more attributes. Then within each Attribute a set of KPI's are defined that describe the data to be collected foreach measure/metric. Some of these KPI's will be service specific while others will apply to all services (BPaaS, IaaS, PaaS and SaaS)

IV. TRUST MODEL ARCHITECTURE

The Trust model is derived from Direct Trust computation model which includes Inter and Intra domain Direct Trust components. Our aim is to build confidence among users, which may be reflected in Trust Score of the providers.

The architecture consists of two major components User Bundles and Data centers of service providers, which encompass of physical and virtual computing units. In all nine cases, the configuration of User bundles is kept constant whereas Data centre configurations are different in order to compare the performance of individual providers. Data centres are dynamically configured using a resource pool of datacenters.

1. SERVICE BROKER POLICIES:

SBP allows us to build data centers around the applications rather than the network. All the parts of data centre like IP, network, storage and applications helps data centers to distribute information at multiple locations.

SBP plays an important role in:

- Connect among the geographical distances according to application and business nature
- Resource sharing like, data migration, computation migration and process migration will be improved.
- It has transparency in presenting hidden details
- Improvement of data protection like, data loss and corruption
- Ability to perform as a remote replication for distribution of contents

- Protection and Security concerns like confidentiality, reliability to unauthorized individuals.
- Protect many business applications like, disaster recovery infrastructure, real time disaster recovery solutions.

The steps are

- 1) Select the region based on the user request.
- 2) Calculate the number of Data Centers in selected region.
- 3) If there is single Data Center then send the request to that specific Data Center.

Major Three service broker policies in CloudAnalyst:

A. Closest Datacenter Policy:

The datacenter with least proximity from the user is selected. Proximity in term of least network latency. If more

than one Datacenters having same proximity then it will select datacenter randomly to balance the load.

B. Optimize Response Time Policy:

First it identifies the closest datacenter using previous policy but when Closest Datacenter's performance (considers response time) starts degrading it estimates current response time for each datacenter then searches for the datacenter which having least estimated response time. But there may be 50:50 chance for the selection of closest and fastest datacenter. (again here random selection).

C. Dynamically reconfigurable routing with load balancing:

Here, routing logic is similar and extension of Closest Datacenter Policy. Based on the load it has one more responsibility of scaling the application. It also increases or decreases the no. of VMs accordingly.

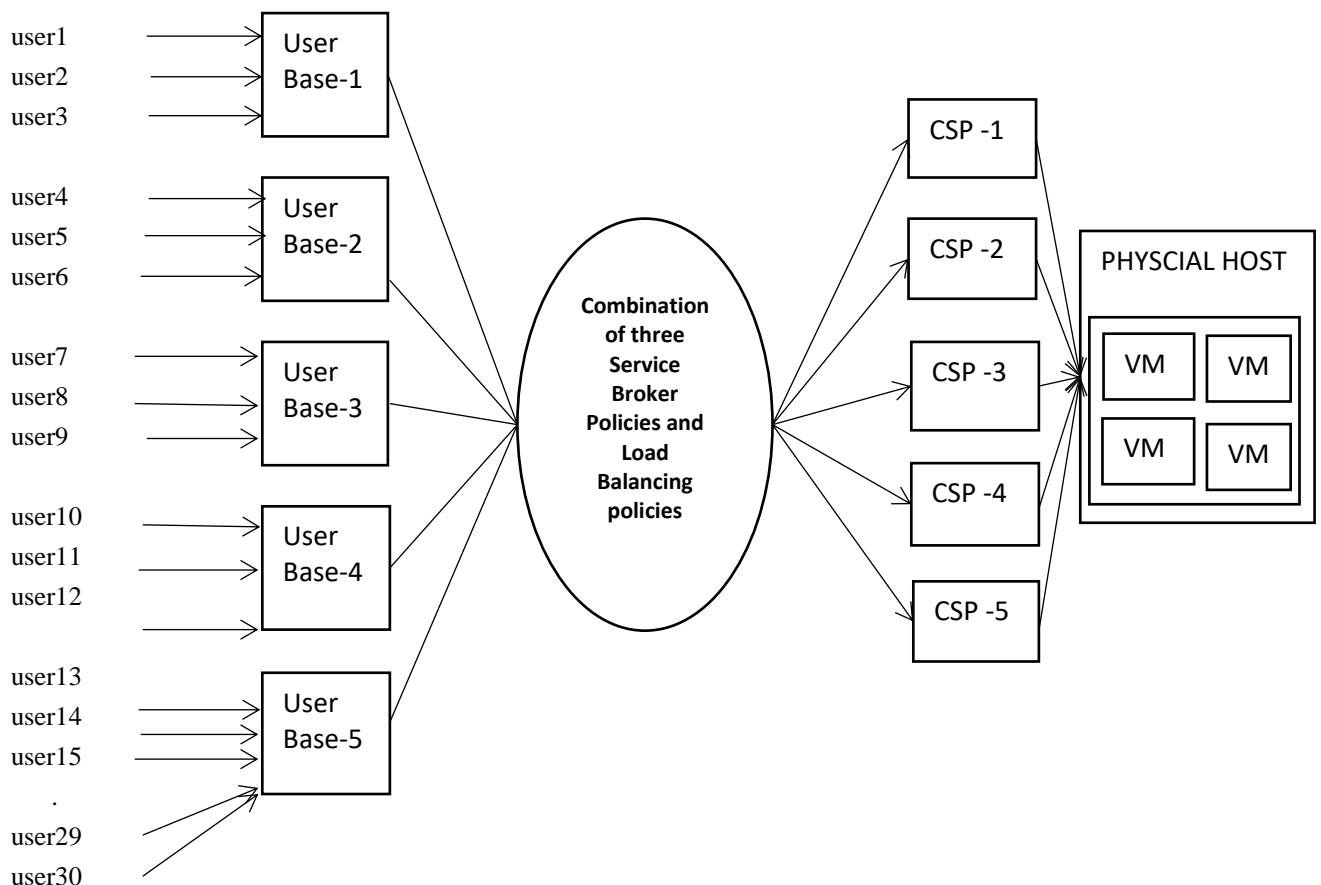


Fig.1 The Block Diagram of the proposed Service Broker Policy and Load Balancing policies Based Trust Computing Procedure

2. LOAD BALANCING IN CLOUD COMPUTING

- Load balancing in the cloud computing environment has an important impact on the performance.

- Good load balancing makes cloud computing more efficient and improves user satisfaction.
- Load balancing is a new technique which provides high resource time and effective resource utilization by assigning the total load among the

various cloud nodes side by side and solves the problem of overutilization and underutilization of virtual machines.

- Load balancing resolve problem of overloading and focuses on maximum throughput, optimizing resource utilization and minimize response time. Load balancing is the pre-requirements for maximizing the cloud performance and utilizing the resources efficiently.
- The load balancing is an efficient concept in cloud computing which helps in utilizing the resources optimally.
- To minimize the consumption of resources, the load to be distributed over the nodes in cloud-based architecture, thereby each resource does the equal amount of work at any point of time that is performed by a load balancer which determines various request allocation to different servers.
- The two major tasks in Load balancing, one is the resource allocation or resource provisioning and other is scheduling in distributed environment.
- To distribute the workload across multiple nodes over the network links to achieve optimal resource utilization, minimum data processing time, minimum average response time.

To avoid overload, the three algorithm are explained below:

- A) Round Robin Algorithm (RR)
- B) Equally Spread Current Execution Algorithm (ESCE)
- C) Throttled Load Balancing Algorithm (TLB)

A) Round Robin Algorithm (RR):

This is the simplest algorithm, uses the concept of time quantum or slices. Here, time is divided into multiple slices and each node given a particular time quantum and within the time quantum the node will perform its operations.

- First, the DataCenterController assign the request to a list of VM's on a rotating basis.
- The first request is allocated to a VM randomly from the group and then DataCenterController assigns the subsequent requests in a circular order.
- The VM is moved to the end of the list when the virtual machine is assigned the request.
- In the RRLB, the *Weighted Round Robin Allocation* concept is used for better allocation in which one can assign a weight to each VM.
- If one VM is capable of handling twice as much load as the other, the powerful server gets a weight of 2.
- In such cases, DataCenterController will assign the two requests to the powerful VM for each request assigned to a weaker one.
- Round Robin Algorithm selects the load randomly, and then leads to a situation where some nodes are heavily loaded and some are lightly loaded.

- However, this algorithm is very simple but there is an additional load on the scheduler to decide the size of quantum.
- It leads to longer average waiting time, higher context switches, higher turnaround time and low throughput.

B) Equally Spread Current Execution Algorithm (ESCE):

In this algorithm, the Load Balancer uphold with an index table of VM's and the number of requests currently allocated to the VM's. In this Algorithm, a communication exist between the load Balancer and the DataCenterController for updating the index table leading to an overhead which leads to delay in providing response to the arrived requests.

Steps involved in this algorithm:

- Initially, all VM's have 0 allocations.
- When a request to allocate a new VM from the DataCenterController arrives, it parses the index table and identifies the least loaded VM.
- If there are more than one, the first identified is selected. The Load Balancer returns the VM ID to the DataCenterController.
- The DataCenterController sends the request to the VM identified by that ID.
- The DataCenterController notifies the Load Balancer of the new allocation.
- The Load Balancer updates the allocation table thereby, incrementing the allocation count of that VM.
- When the VM finish processing request and DataCenterController receives the response cloudlet, it notifies the Load Balancer of the VM de-allocation.
- By decrementing the allocation count for the VM one by one, the Load Balancer updates the allocation table.

C) Throttled Load Balancing Algorithm (TLB):

In this algorithm, the Load Balancer maintains an index table of VM's as well as their states (Available/Busy). The following steps involved in this algorithm:

- When a request to allocate a new VM from the DataCenterController arrives, it parses the index table from top until the first available VM is found.
- If VM is found, the Load Balancer returns the VM ID to the DataCenterController.
- The DataCenterController send the request to the VM identified by that ID.
- The DataCenterController notifies the Load Balancer of the new allocation.
- Accordingly, the Load Balancer updates the allocation table by incrementing.
- When the VM finishes processing the request and the DataCenterController receives the response cloudlet, it notifies the Load Balancer of the VM de-allocation.

V. ABOUT CLOUDSIM

In Cloud service provisioning establishment and access to the infrastructure is costly in real and we need simulation approaches which provide enormous advantages to Cloud service customers to evaluate their services often

in a compact environment with no cost. In addition, they can be able to fine-tune the performance issue before deploying on the actual Cloud platform.

CloudSim provide a framework which is autonomous and extensible for simulation which provide great results in terms of simulation, and testing of Cloud computing application services and platforms. By using CloudSim, researchers may concentrate on application specific design issues instead of concerning on bottom level details of infrastructures and services. It used simulation of large computing in data centers and simulation of virtualized hosts, with flexible strategies for provisioning physical host machines to virtual machines.

VI. CLOUDANALYST

CloudSim helps to enable smooth modelling, simulation, and practicing on cloud computing infrastructure. It can be used as a platform to model data centers, service brokers, and scheduling and allocation policies of large- scale cloud platforms. CloudAnalyst [6] is built directly on top of the CloudSim toolkit.

a) *Main components of CloudAnalyst and the function of each component:*

1. **GUI Package** - is responsible for the GUI, for serving as the frontend controller for the application, and for managing screen transitions and other user interface activities.
2. **Region** - Six regions correspond to six continents in the world.
3. **User Base** - This component models a group of users and generates traffic that represents the users.
4. **Data Center** - encapsulates a set of computing hosts or servers that are either heterogeneous or homogeneous in nature depending on their hardware configurations.
5. **Data Center Controller** - controls data center activities.
6. **Cloudlet** - specifies a set of user requests. It contains the application ID, name of the user base as originator for routing back the responses, size of request execution commands, and input and output files.
7. **Internet**. This component models the Internet and implements the traffic routing behavior.
8. **Internet Characteristics** - component is used to define the characteristics of the Internet applied during simulation.
9. **VM Load Balancer** - component models the load balance policy used by data centers when the serving allocation is requested.

User Base and Data Center Regions

Code	0	1	2	3	4	5
Region	N. America	S. America	Europe	Asia	Africa	Oceania



Fig.2 Geographical regions in the world distributed in Cloud Analyst

VII. SIMULATION SETUP

We setup five user bases representing six main regions of the world represented with different configurations and which remain constant across all scenarios whereas the various Cloud service providers configuration varies. The service provider configurations are created using the pool of Data centers at various regions of the world. Five different simulations are run and each

produces an output report detailing the Total cost, Minimum cost, Maximum cost, Response time, Data centre processing time, Data Center Minimum time, Data Center Maximum time and user base request service times. The setup of service providers and 5 user bundles are analysed for all the three schemes to verify the changes in the trust score which reflect changes in the ranking.

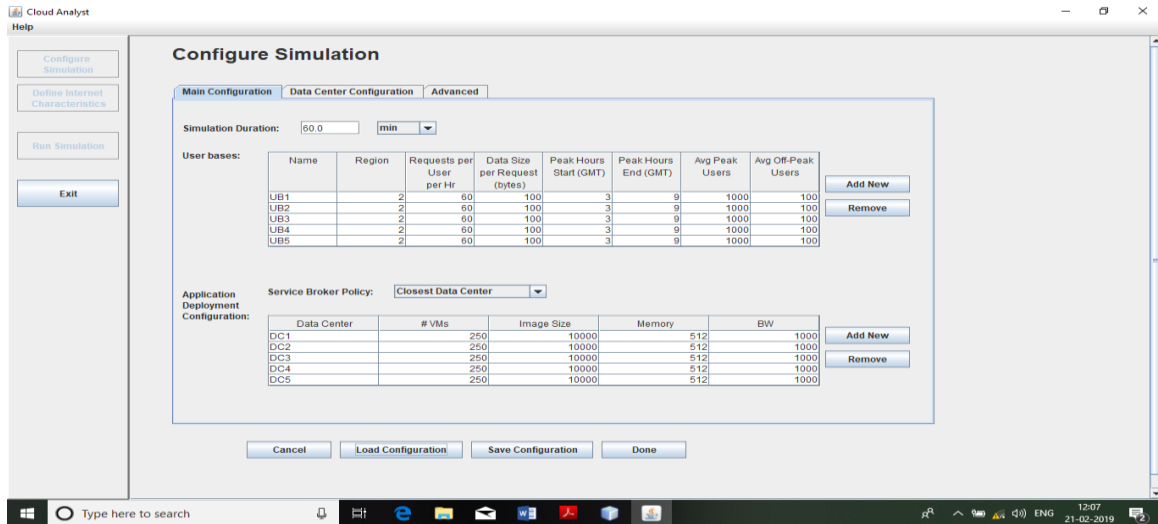


Fig.3 Configuring of Service Broker Policy in Cloud Analyst Running Simulation

Simulation in progress during running and after complete running are given in following figures. In the figures UB represents User Bundle/User Base and DC represents Data Centers.

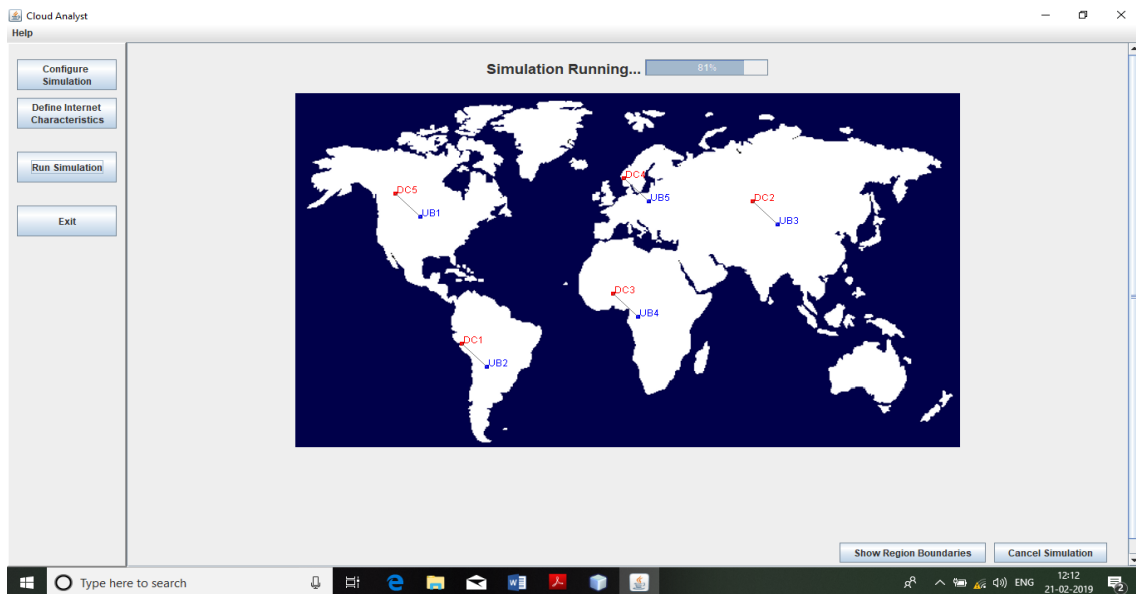


Fig.4 Running status of simulation using service broker policy strategy



Fig.5 Results after the completion of the simulation

a) Fuzzy logic implementation

Fuzzy inference has been widely used to solve and control reasoning problems in uncertain scenarios due to its ability to handle inaccurate inputs. The Three Major components of FIS are :

1. **Inference Engine**, which defines fuzzy logic operators and defuzzifier used in the inference process.
2. **Membership Functions**, defines the degree of fuzzy element belongs to the corresponding fuzzy se. It

maps crisp values to membership degrees between 0 and 1

3. **Rule base**, is a set of “If-then” rules which defines the inference model. The rule structure enables antecedent and consequent fuzzy prepositions connected with “AND” or “OR” operators.

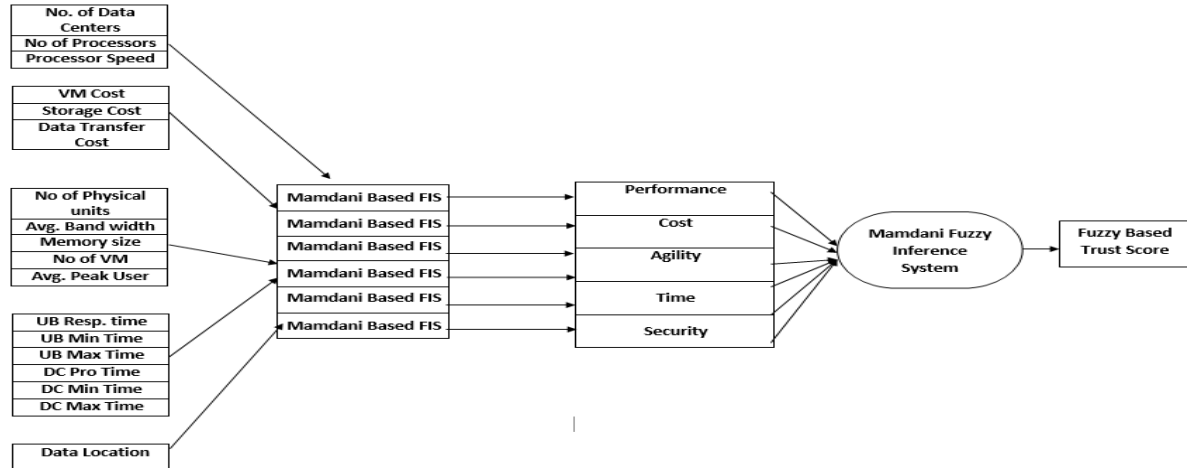


Fig.6 Two stage Hierarchical Fuzzy Inference system for Cloud Trust Score Computation

b) The inference process involves five major steps:

1. **Fuzzification**: Input crisp values into the membership functions to obtain corresponding membership degrees of each input variable regarding specific fuzzy set.
2. **Applying Fuzzy operations**: Obtain the membership degree of the antecedent using “AND” and “OR” operators.
3. **Implication**: Obtain the fuzzy set of each rule using the defined aggregation operator.
4. **Aggregation**: aggregate output fuzzy sets of all rules using the defined aggregation operator.
5. **Defuzzification**: transform the aggregated fuzzy set into a crisp value using the defined defuzzification algorithm

Fuzzy model is used to extend the mathematics ontology in certain method with fuzziness for making an intelligent decision. The proposed fuzzy method uses three fuzzy sets low (L), medium (M) and high (H) to characterize the fuzzy value for each input, which are Performance, Cost, Agility, Time and Security. In the above parameters, Cost and Time are considered as parameters expected to be lower which are inversely proportional to get high trust score, while Performance, Agility and Security need to be in higher side, which are directly proportional to achieve a high trust score.

VIII. PARAMETERS MODEL

a) Performance:

Performance has 3 input attributes namely Number of Data Centers, Number of Processors and Processor speed. All three have 3 member functions each. The output ‘Performance’ has 3 membership functions low,

medium and high. The output obtained for all service providers from the FIS are used for comparison.

b) Cost:

Cost has 4 input attributes namely Memory Cost, Storage Cost, Total Virtual Machine Cost, Total Data Transfer Cost. All four have 3 member functions each. The output ‘Cost’ has 3 membership functions low, medium and high. The output obtained for all service providers from the FIS are used for comparison.

c) Agility:

Agility has 6 input attributes namely Number of Physical units , Bandwidth, Memory Size, No. Of Virtual Machines, Request per user per Hour and Average Peak Users. All six have 3 member functions each. The output ‘Agility’ has 3 membership functions low, medium and high. The output obtained for all service providers from the FIS are used for comparison.

d) Time:

Time has 6 input attributes namely User Base Response Time, User Base Minimum Time, User Base Maximum Time, Data Center Base Processing Time, Data Center Minimum Time and Data Center Maximum Time. All six have 3 member functions each. The output ‘Time’ has 3 membership functions low, medium and high. The output obtained for all service providers from the FIS are used for comparison.

e) Security:

Security has one input attributes namely Number of Data Locations. This has 3 member functions each. The output ‘Security’ has 3 membership functions low, medium and high. The output obtained for all service providers from the FIS are used for comparison.

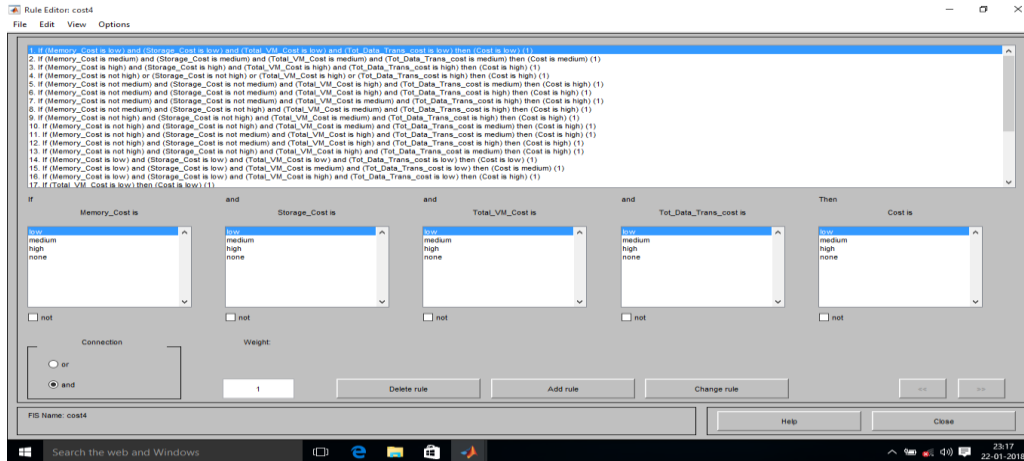


Fig.7 Rule Editor on FIS for Trust Computation with parameters

The rule viewer is used to dynamically adjust the input parameter values instantly, and view the changes in the output variable. We may easily modify and aware of input parameters to achieve the maximum output level.

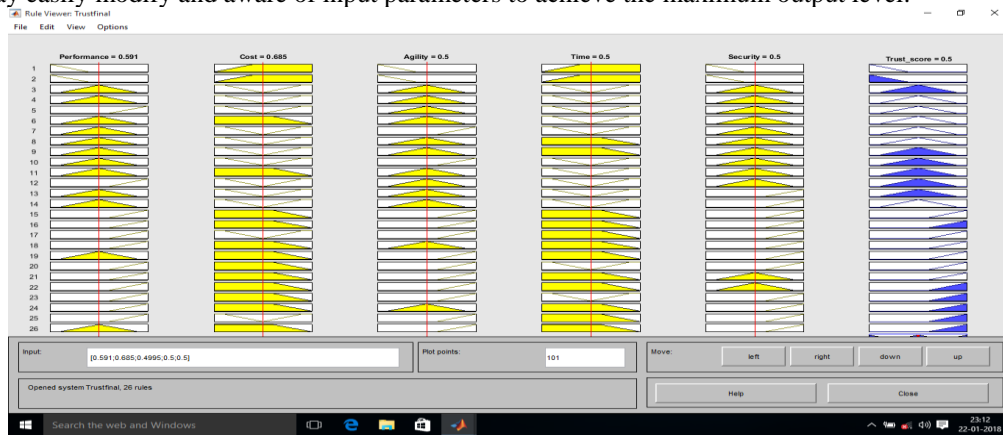


Fig.8 Rule Editor on FIS for Trust Computation

IX. RESULTS:

The outcome compares the ranking among service providers among 9 combinations such as

1. Closest Data Center-Round Robin(CD-RR)
2. Closest Data Center- Equally Spared Current Execution Load (CD-ESEL)
3. Closest Data Center- Throttled(CD-T)
4. Optimise Response Time- Round Robin(ORT-RR)
5. Optimise Response Time- Equally Spared Current Execution Load(ORT-ESEL)
6. Optimise Response Time- Throttled(ORT-T)

7. Dynamic Allocation- Round Robin(DA-RR)
8. Dynamic Allocation- Equally Spared Current Execution Load(DA-ESEL)
9. Dynamic Allocation- Throttled(DA-T)

Trust Values of Cloud Service Providers and their Ranking based on their trust scores are obtained for all five strategies are given below. The below table gives the ranking of service providers based on trust score under the strategy of SBP-Dynamic allocation with three load balancing combinations.

Table 1 Trust score for SBP-Dynamic Allocation with three Load Balancing policies

CSPs	PERFORMANCE	COST	AGILITY	TIME	SECURITY	TRUST SCORE
DYNAMIC ALLOCATION- ROUND ROBIN						
CSP 1	0.1386	0.1300	0.1300	0.1488	0.1300	0.2550
CSP 2	0.1572	0.3834	0.1445	0.1488	0.1405	0.5
CSP 3	0.4310	0.5	0.1692	0.1300	0.1628	0.5
CSP 4	0.5	0.4194	0.5	0.1300	0.1405	0.5113
CSP 5	0.8700	0.1530	0.5	0.5	0.1300	0.5270
DYNAMIC ALLOCATION-EQUALLY SPREAD CURRENT EXECUTION LOAD						
CSP 1	0.1386	0.1300	0.1300	0.1300	0.1300	0.2550
CSP 2	0.1572	0.3834	0.1445	0.1488	0.1405	0.5
CSP 3	0.4310	0.5	0.1692	0.1300	0.1628	0.5
CSP 4	0.5	0.4194	0.5	0.1300	0.1405	0.5109
CSP 5	0.8700	0.1530	0.5	0.5	0.1300	0.5270
DYNAMIC ALLOCATION- THROTTLED						
CSP 1	0.1386	0.1300	0.1300	0.1488	0.1300	0.2779
CSP 2	0.1572	0.3834	0.1445	0.1488	0.1340	0.5
CSP 3	0.4310	0.5	0.1692	0.1300	0.1556	0.5
CSP 4	0.5	0.4179	0.5	0.1300	0.1432	0.5113
CSP 5	0.8700	0.1530	0.5	0.5	0.1556	0.5372

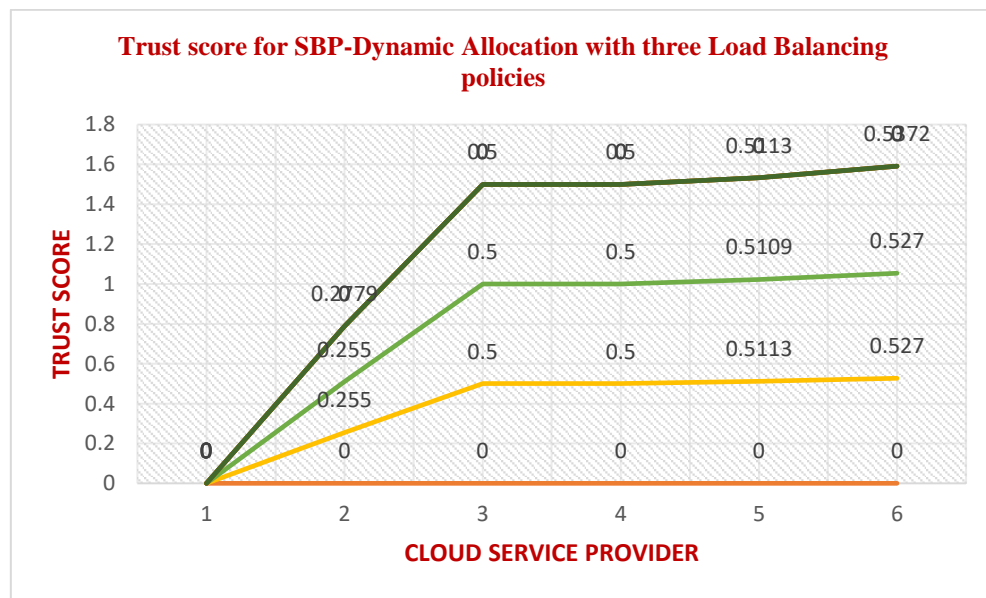


Fig.9 Trust score for SBP-Dynamic Allocation with three Load Balancing policies

The below table gives the ranking of service providers based on trust score under the strategy of three Service Broker Policies with three load balancing combinations. Finally, average trust score of each combination are obtained for analysing purpose.

Table 2 Comparison of ranks obtained from all nine different combination

CLOUD SERVICE PROVIDER	CD-RR	CD-ESEL	CD-T	ORT-RR	ORT-ESEL	ORT-T	DA-RR	DA-ESEL	DA-T
CSP 1	0.2550	0.2550	0.2550	0.2779	0.2781	0.2550	0.2550	0.2550	0.2779
CSP 2	0.5	0.5	0.5	0.5	0.5	0.5	0.5	0.5	0.5
CSP 3	0.5	0.5	0.5	0.5	0.5	0.5	0.5	0.5	0.5
CSP 4	0.5113	0.5113	0.5113	0.5113	0.5113	0.5113	0.5113	0.5109	0.5113
CSP 5	0.5270	0.5270	0.5270	0.5270	0.5270	0.5270	0.5270	0.5270	0.5372
AVERAGE SCORE	0.45866	0.45866	0.45866	0.46364	0.46328	0.45866	0.45866	0.45858	0.46528
RANK				2	3				1

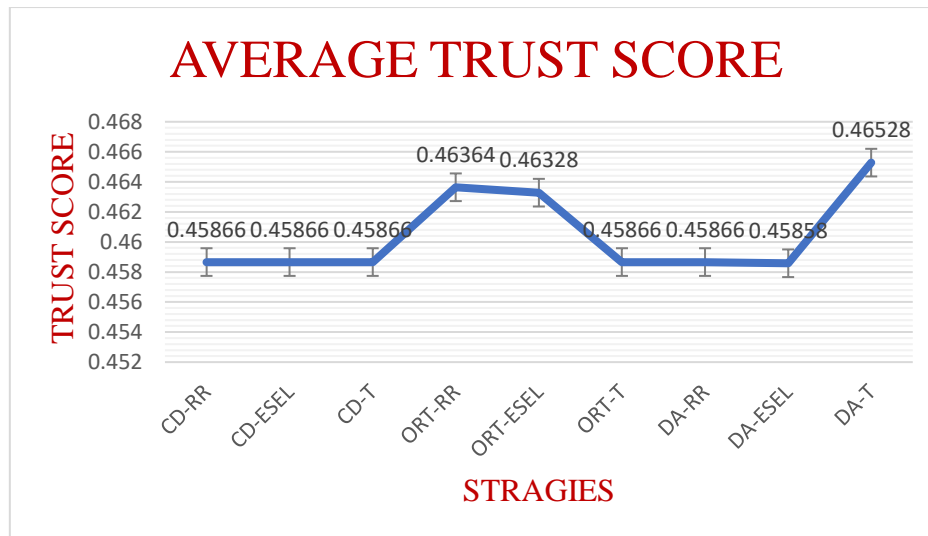


Fig.9 Average trust score

X. CONCLUSION AND DISCUSSION

REFERENCES

It is observed that there is no specific change in the trust score of first 3 combination of closest datacenter service broker policy with 3 load balancing policies. The rank 3 and 4 among five providers are similar in all combination of SBP and Load balancing policies. Among the five Cloud service provider, CSP 5 remains the highest trust score (0.5372) in case of Dynamic Allocation-Throttled combination (case study 9).

Among the nine case studies, we obtained highest average trust score in case 9(Dynamic Allocation-Throttled), case 4(Optimise response time-Round Robin) and then, case 5(Optimise response time-Equally spread current execution load). We obtain lowest average trust score in case 1 that is with SBP-Closest datacenter strategy.

Hence, based on highest average trust score and highest individual trust score, case 9, that is SBP-Dynamic Allocation-Throttled, stand first. Hence

- i. The user may strongly prefer CSP 5 which results in high individual trust score, 0.5372.
- ii. The scenario SBP-Dynamic Allocation along with Throttled combination may be preferred which results in high average trust score, 0.46528.

Also, based on lowest average trust score, case 8, that is SBP-dynamic allocation-Equally spread current Execution load, stand last. Hence

- i. The user may strongly avoid CSP 1 which results in low individual trust score in all strategies.
- ii. The scenario SBP- dynamic allocation- Equally spread current Execution load combination may be avoided by the provider to organize cloud service which resulted in low average trust score, 0.45858.

- [1] Supriya M. Venkataramana L.J, K Sangeetha and G K Patra, "Estimating Trust Value for Cloud Service Providers using Fuzzy Logic", International Journal of Computer Applications, Volume 48 – No.19 , June 2012.
- [2] Hamdy M. Mousa, Gemal F. Elhady, "Trust Model Development for Cloud Environment using Fuzzy Mamdani and Simulators", International Journal of Computers and Technology, Volume 13, No. 11, September 2014, pp. 5142-5153.
- [3] Garg, S.K., Versteeg, S and Buyya, R. "SMICloud: A Framework for Comparing and Ranking Cloud Services", Fourth International Conference on Utility and Cloud Computing, Australia, December 2011, pp.210-218.
- [4] Chenhao Qu and Rajkumar Bayya, "A cloud trust evaluation system using hierarchical fuzzy inference system for service selection", 2014 IEEE 28th International Conference on Advanced Information Networking and Applications, 2014 pp 850-857.
- [5] Qu, Chenhao, and Rajkumar Buyya. "A Cloud Trust Evaluation System Using Hierarchical Fuzzy Inference System for Service Selection", 2014 IEEE 28th International Conference on Advanced Information Networking and Applications, 2014.
- [6] Chenhao Qu and Rajkumar Bayya, "A cloud trust evaluation system using hierarchical fuzzy inference system for service selection", 2014 IEEE 28th International Conference on Advanced Information Networking and Applications, 2014 pp 850-857.
- [7] Sonia Lamba , Dharmendra Kumar "A Comparative Study on Load Balancing Algorithms with Different Service Broker Policies in Cloud Computing" (IJCSIT) International Journal of Computer Science and Information Technologies, Vol. 5 (4) , 2014, 5671-5677.
- [8] Wickremasinghe, B, Calheiros R.N and Buyya, R. "CloudAnalyst: A CloudSim-Based Visual Modeller for Analyzing Cloud Computing Environments and Applications" 24th International Conference on Advanced Information Networking and Applications, Australia, April 2010, pp. 446-452.
- [9] Saurabh Kumar Garg. "SMICloud: A Framework for Comparing and Ranking Cloud Services" , 2011 Fourth IEEE International Conference on Utility and Cloud Computing, 12/2011.
- [10] Cloud Service Measurement Index Consortium (CSMIC). "Service Measurement Index Version 1.0" (PDF), USA, September 2011
- [11] Selvaraj, Alagumani, and Subashini Sundararajan. "Evidence-Based Trust Evaluation System for Cloud Services Using Fuzzy Logic" , International Journal of Fuzzy Systems, 2016.
- [12] C. Qu and R. Buyya "A Cloud Trust Evaluation System using Hierarchical Fuzzy Inference System for Service Selection" IEEE 28th International Conference on Advanced Information Networking and Applications, 2014.
- [13] Reena Panwar and Bhawna Mallick "A Comparative Study of Load Balancing Algorithms in Cloud Computing" International Journal of Computer Applications (0975 – 8887) Volume 117 – No. 24, May 2015.

- [14] Neha Singla "Load balancing of user processes among virtual machines in cloud computing environment" ISSN (PRINT): 2393-8374, (ONLINE): 2394-0697, VOLUME-2, ISSUE-5, 2015.
- [15] H. M. Alabool and A. K. Mahmood, "Trust-based service selection in public cloud computing using fuzzy modified vikor method," *Australian Journal of Basic and Applied Sciences*, vol. 7, no. 9, pp. 211–220, 2013.
- [16] http://www.mathworks.com/help/pdf_doc/fuzzy/fuzzy.pdf Fuzzy Logic Toolbox™ User's Guide.
- [17] Kim Won, "Cloud Computing : Today and Tomorrow", 2009.
- [18] Mohammed Alhamad, Tharam Dillon and Elizabeth Chang. "A Trust-Evaluation Metric for Cloud Applications", *International Journal of Machine Learning and Computing*, Volume 1, Number 4, October 2011, pp. 416-421.
- [19] Mohamed Firdhous, Osman Ghazali and Suhaidi Hassan, "Trust Management in Cloud Computing: A Critical Review". *International Journal on Advances in ICT for Emerging Regions*, 2011, 04 (02): 24–36.
- [20] R. Buyya et al., "Cloud computing and emerging IT platforms: Vision, hype, and reality for delivering computing as the 5th utility," *Future Generation Computer Systems*, vol. 25, no. 6, pp. 599–616, 2009.
- [21] Peter Mell and Timothy Grance. "The NIST Definition of Cloud Computing (Draft) Recommendations of the National Institute of Standards and Technology", NIST Special Publication 800-145, September 2011, 84 pages.
- [22] Tripathi, A and Mishra, A. "Cloud Computing Security Considerations", *International Conference on Signal Processing, Communications and Computing*, India, September 2011, pp. 1-5.
- [23] Mohamed Firdhous, Osman Ghazali, and Suhaidi Hassan. "Trust and Trust Management in Cloud Computing – A Survey", *Internet Networks Research Group, University Utara Malaysia, Technical Report*, February 2011.
- [24] *International Journal of Computer Applications (0975 – 888) Volume 48– No.19, June 2012*
- [25] Cloud Security Alliance, "Top Threats to Cloud Computing V1.0", March 2010.
- [26] *International Journal of Computer Applications (0975 – 888) Volume 48– No.19, June 2012*
- [27] Cloud Security Alliance, "Top Threats to Cloud Computing V1.0", March 2010.
- [28] Jensen M, Schwenk J, Gruschka N and Iacono, L.L. "On Technical Security Issues in Cloud Computing", *International Conference on Cloud Computing*, Germany, September 2009, pp. 109-116.
- [29] Siani Pearson, "Privacy, Security and Trust in Cloud Computing". *HP Laboratories, Springer*, June 2012.
- [30] S. K. Garg, S. Versteeg, and R. Buyya, "A framework for ranking of cloud computing services," *Future Generation Computer Systems*, vol. 29, no. 4, pp. 1012–1023, 2013.
- [31] Somesh Kumar Prajapati, Suvamoy Changder and Anirban Sarkar, "Trust Management Model For Cloud Computing Environment". *Proceedings of the International Conference on Computing, Communication and Advanced Network -ICCCAN 2013*.
- [32] Reena Panwar, Bhawna Mallick, Ph.D, "A Comparative Study of Load Balancing Algorithms in Cloud Computing" *International Journal of Computer Applications (0975 – 8887) Volume 117 – No. 24, May 2015*.
- [33] Veerawali Behal, Anil Kumar "Comparative Study of Load Balancing Algorithms in Cloud Environment using Cloud Analyst" *International Journal of Computer Applications (0975 – 8887) Volume 97– No.1, July 2014*.
- [34] <https://www.cmu.edu/>
- [35] <http://www.buyya.com/>
- [36] www.cloudbus.org/cloudsim/
- [37] <https://cirworld.com/index.php/ijct/article/view/2784>
- [38] <http://www.ijcaonline.org/>
- [39] <https://www.iaop.org/>
- [40] <https://espace.curtin.edu.au/>
- [41] <http://www.redorbit.com/>
- [42] <http://www.i-scholar.in/>
- [43] <https://www.irjet.net/>
- [44] <https://www.ijies.org/>
- [45] <https://www.coursehero.com/>