Hindawi Publishing Corporation
Mathematical Problems in Engineering
Volume 2013, Article ID 612182, 11 pages
http://dx.doi.org/10.1155/2013/612182

Research Article

B3: Fuzzy-Based Data Center Load Optimization in Cloud Computing

M. Jaiganesh and A. Vincent Antony Kumar

Department of Information Technology, PSNA College of Engineering and Technology, Dindigul, Tamil Nadu 624 622, India

Correspondence should be addressed to M. Jaiganesh; jaidevlingam@gmail.com

Received 31 December 2012; Revised 24 February 2013; Accepted 25 February 2013

Academic Editor: Ming Li

Cloud computing started a new era in getting variety of information puddles through various internet connections by any connective devices. It provides pay and use method for grasping the services by the clients. Data center is a sophisticated high definition server, which runs applications virtually in cloud computing. It moves the application, services, and data to a large data center. Data center provides more service level, which covers maximum of users. In order to find the overall load efficiency, the utilization service in data center is a definite task. Hence, we propose a novel method to find the efficiency of the data center in cloud computing. The goal is to optimize date center utilization in terms of three big factors—Bandwidth, Memory, and Central Processing Unit (CPU) cycle. We constructed a fuzzy expert system model to obtain maximum Data Center Load Efficiency (DCLE) in cloud computing environments. The advantage of the proposed system lies in DCLE computing. While computing, it allows regular evaluation of services to any number of clients. This approach indicates that the current cloud needs an order of magnitude in data center management to be used in next generation computing.

1. Introduction

Cloud computing is an evolving paradigm to access assortment of data pool via internet by using connective devices such as Personal Digital Assistant (PDA), work station, and mobile [1–4]. It is a utility-based computing, which has the capability to deliver services over the internet. It provides on-demand access without any human intervention. The standard deployment object that is used in cloud computing is Virtual Machines (VM). It enhances flexibility and enables data center to be dynamic in nature. The techniques of dividing a physical computer into several partly or completely isolated machines are known as virtualization [5, 6]. A collection of data is stored in a centralized pool called Data Center (DC) [7–9]. Cloud computing is the art of managing tasks and applications by altering the software, platform, and infrastructure and by organizing third party data centers known as Cloud Service Providers (CSP) such as Yahoo!, Amazon, Google, and VMware [2, 10]. Data center is deployed as an individual server room which is hosted within the organization. It runs several applications on a single server. In cloud computing, the data center provides more services, which covers maximum numbers of users [11]. So, cloud service providers are prepared in better tolerance to manage and update the data centers. Cloud computing provides myriad of services [12]. Therefore, the data center is too costly to build and manage. The challenges of data centers are the following.

(i) *Irrefutable Cost:* Construction of low cost data center is unaffordable for a single compound. Cloud computing built a centralized data center which requires increasing cost in servers and storage.

(ii) *Workload Utilization:* Cloud computing needs new servers to be installed in data centers. Virtualization has enabled many applications to run on a single server or couple of servers. Some key factors of utilization are storage, power, cooling, response time, capacity, and efficiency.

(iii) *Optimization of Services:* Numerous data centers applications provide variety of services. So, finding overall load efficiency and utilization of services is a complex task associated with data center.Due to enormous applications running on it, optimization of data center service is a major challenge.

The major difficulty in a data center is to deploy that producers are expected to have better knowledge in monitoring data centers so that they are able to find the service utilization issues by managing the data center load configurations [13–16]. In [17], presented a data center utilization scenario to monitor and analyze cloud system, the utilization of client specification bounds such as bandwidth, memory and CPU utilization.

Fuzzy Logic was introduced by Zadah [18–20]. It is a problem-solving system methodology that lends itself to survive systems ranging from simple to sophisticated to survive. It is used in embedded, networked, distributed systems. Fuzzy set is a common set that has collection of elements measuring improbability in the set. It has varying degrees of membership in the set. A typical function of a crisp set allocates a value of either 1 or 0 to each individual in the universal set. The function can be comprehensive in such a manner that the values are assigned to the elements of the universal set. Huge values represent upper degrees of set membership, and it is called membership function and the set is identified as fuzzy set. The most usually employed range of values of membership function is the unit interval $[0, 1]$. Each membership function plots elements of a given universal set X, and it is always a crisp set, into real numbers in $[0, 1]$. The membership function of fuzzy set A is denoted by μ_A; that is, $\mu_A : X \rightarrow [0, 1]$. Each fuzzy set is completely and uniquely defined by one particular membership function, and it may also be used as labels of the associated fuzzy sets [21]. Each element of fuzzy set is mapped to a universal membership value by using function theoretic form [22]. It is having an element in the universal set X, is a member of fuzzy set A, and then this mapping is given by $\mu_A(X) \in [0, 1]$, where $\mu_A(X)$ is called grade of membership.

In this work, extensive use of Fuzzy logic has been deployed to find the data center load efficiency. Here, we used crisp value of input as real numbers, and in the next analysis, we intend to go in for Fuzzy Fractal Dimensions [23, 24]. Data center load efficiency is the key object. Here, the fuzzy fractal dimension is denoted by the pair of Bandwidth (BW) and Memory of the CPU fields [25, 26]. Here, BW is the numerical value of the fractal dimension of bandwidth, and M_0 is the membership function of bandwidth, namely, the memory and CPU. It is mainly because the Memory and CPU are dependent on the bandwidth. The unevenness of the dynamically changing resource requirements and the emerging demand pattern can be compared to the different geometric objects [27, 28]. Hence, here, we apply fuzzy rules to differentiate the different patterns and cluster them. Fractal geometry can be used to classify different objects based on their roughness [24, 26, 29]. In this case, the focus is based only on the smoother objects where the limitation of the fractal value is only up to one, and if it is closer to one, then that means maximum utilization of the memory and CPU resources. If memory is M_1 and CPU cycle is C_1, then the data center load efficiency is *DC1*. Thus, based on the input parameter, the output object efficiency is predicted using the simple fuzzy rules. Only disadvantage here is that based on the parameters the total number of rules increases causing problem due to dimensionality. Based on this model which has been created, the future values of the demand for CPU and memory can be predicted leading to accurately assess the efficiency of the data center in the varying situations.

1.1. Background. In recent times, more attention is shown on the framework of cloud computing and the performance evaluation. Iosup et al. [30] have done "performance analysis of cloud computing services" approach for supporting efficiency of cloud computing. In their model, they analyze the performance of Many Task Computing (MTC) workloads. They have proposed a comparison on performance characteristics and cost models. Moreno-Vozmediano et al. [31] have deployed a computing cluster on the top of many task computing applications. In this subsequent work, cluster loads have been used for resources from different clouds to construct high availability strategies. These are used for proving viability to perform scalability of resources and performances for large scale cluster infrastructure. Dutreilh et al. [32] have considered the recent research to construct a data center management framework for atomic resource allocation in virtual applications. They evaluated in two ways, namely, threshold-based and reinforcement learning methods to dynamically scale resources.

Yazir [33] presented a virtualization tool that provides this gap by applying ideas from computational geometry. It proved valuable assistance in providing quick and easy preliminary performance analyses. Data processing management is difficult to get as many machines as an application needs. The large scale jobs are distributed on different machines as parallel running processes. The control and coordination of these processes is complex with time dependent. Cloud Architectures [34] have solved such difficulties. Cloud administrators usually worry about hardware procuring (when they run out of capacity) and better infrastructure utilization (when they have excess and idle capacity). The lower network bandwidth and the inherent lower hardware dependability force enterprises to reorganize cloud application architecture [35]. From the data center challenges and methodologies, the two key questions arise. How are the efficiency of data centers and performance of cloud computing calculated? What are the key factors to decide the efficiency of data center in cloud computing? This paper answers these questions. The contributions of the paper are outlined as follows.

(1) We compute the DCLE be (η) for load-based data center management in cloud computing. (η) is a valuable perception for cloud service providers to monitor, manage, and mitigate the cloud computing services and justify the ability to hire a single virtual client or thousands of virtual clients.

(2) We construct it as a fuzzy expert system model to find the DCLE with basic three factors like Bandwidth, Memory, and CPU cycles to validate the steps of our

model which is possible through tangible implementation and assessment.

This paper is organized as follows. Section 2, gives the problem identification. Section 3, the deals with problem formulation, preliminaries, and definitions. Section 4 presents finding of data center load efficiency using fuzzy modeling, Section 5, provides the performance analyzes and experiment results. Section 6 gives the conclusion of the paper.

2. Problem Identification

The objective of this work is to assess the data center load efficiency, when more number of clients and several requests are running on the same server. The typical web application used in cloud computing has the potential capacity constraints such as bandwidth into the load balancer, CPU cycle, and memory of the load balancer [36, 37]. The ability of the load balancer depends upon (i) bandwidth between the load balancer with application server [38, 39]; (ii) CPU cycle and memory of the application server; (iii) bandwidth between application server and network storage devices; (iv) data storage and Disk I/O of database server [40]. The following major three factors play a vital role in cloud computing:

(1) Bandwidth,

(2) Memory,

(3) Central Processing UnitCycle (CPU Cycle).

2.1. Bandwidth. In corporate motto, the cloud computing is operationally exhaustive and obviously parallel. In any software that runs on entire virtual client, it should be communicative. It is not giving operational transaction and bandwidth assurance. The cloud service provider [28] can offer a bandwidth, which is found through their network connections of data center with internal as well as in public internet. The data centers can provide consistency and service delivery efficiently. It includes the guaranteed amount of bandwidth that every client should get [41, 42]. The number of service tends to grow, and cloud service provider increases the cloud information rate which also brings increase in their bandwidth [43–45]. Based on High Performance Computing (HPC), challenging results exist in [44, 46]. Figure 1 depicts the bandwidth utilization of High Performance Cluster Computing (HPCC) for GoGrid cloud computing platforms. Here, bandwidth is calculated for HPCC performance prediction. The volume of services on the cloud computing keeps on growing and tends to more bandwidth [24, 26, 47, 48]. The bandwidth utilization and the data center load are directly proportional to each other; that is, when the bandwidth utility in cloud increases, the data center load also increases, and vice versa. Hence, the bandwidth utilization is considered as one among the big three factors for providing a good cloud service to the customers.

2.2. Memory. It is a major difficulty for storage and delivery of services in cloud computing. It is purely depending upon the application or task used by the client. In cloud computing, the applications and the files are permanently stored in data center by the access of third party clients and users. Amazon's Simple Storage Service (S3) (e.g.). In cloud survey [49], Figure 2 shows the memory usage of Amazon EC2 platforms m1.small to c1.xlarge. In dynamic nature of data centers [46], the database management system requires more amount of memory for processing the services. The memory should be elastic in nature, such that applications are being performed. Memory is comparatively low while running SaaS applications. So, the memory elasticity and memory visualization are manageable see; [50, 51]. In cloud computing

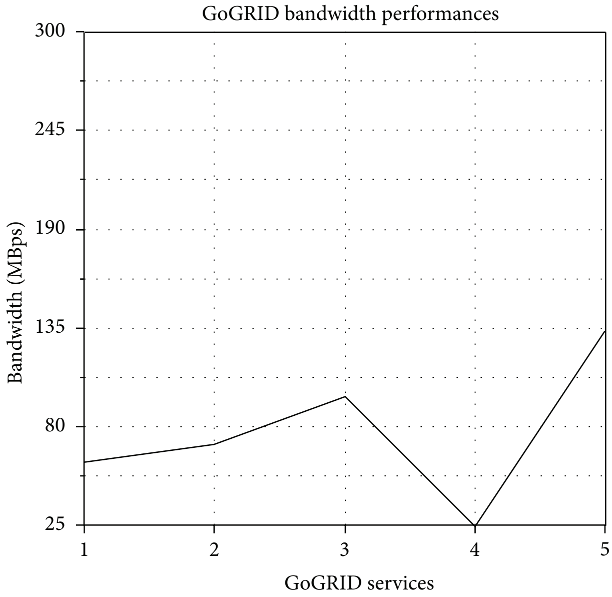


Figure 1: GoGRID bandwidth utilization.

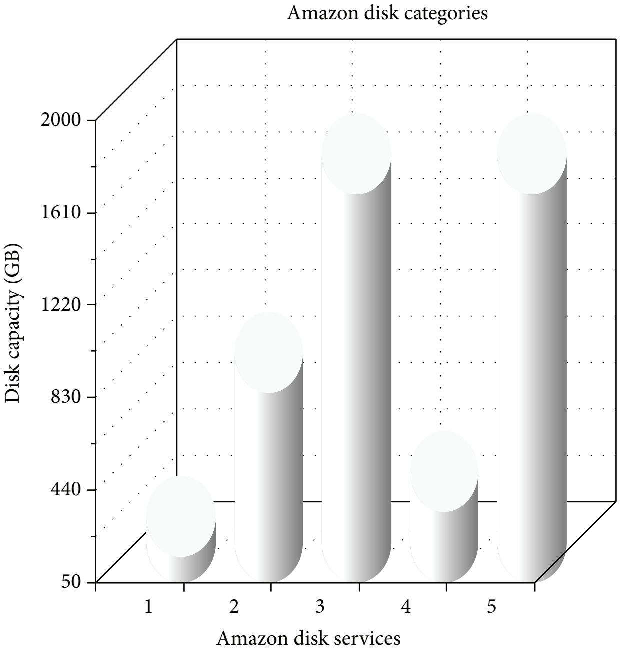


Figure 2: Amazon EC2 Instances-Memory utilization.

many of CPU's transaction is done in a single data center. So, memory is able to tolerate the CPU transactions and service performance calculations. Because of this aforementioned facts, The memory is another important factor to construct DCLE.

2.3. Central Processing Unit Cycle (CPU Cycle). Third, cloud computing needs core of processors present in a single fragment and providing high concurrent throughput for services with parallel operation. In cloud computing, utilization of CPU is an important factor. An input supplied factor to a processor's computing power is its clock speed. It is an approximation to the division of clock speeds that actually take place for a given processor design. In addition, the advent of new processors affects purchase of existing processors. Data center applications need large amount of memory not at all having CPUS responsible for processing. According to this situation, CPU with efficient performance called work station is installed. In cloud computing, the same work station is termed as data center. In the real world, memory is limited and not infinite. Then, we only prefer CPU cycle to be the one of the prime factor to decide DCLE. The database applications are deployed on mainframe computer or server with huge capacity. In [46], the grid workload archive traces along with CPU utilization. The cloud computing system will need some of 100's CPU's for multiprocessing architectures. It starts from CPU ranges from 64 to 128. We identified that previous three big factors play a major role in computing of DCLE. We present these big three factors to obtain an optimized value of maximized data center efficiency. It is done through a valid problem solving control system using fuzzy modeling.

3. Problem Formulation

The proposed model is formulated as knowledge base fuzzy expert system modeling [52, 53]. We propose a novel approach that has been tightening in data center to find the new perception called Data Center Load Efficiency (DCLE). This factor is predicted in network load configuration region. DCLE is depicted as three important fundamental factors. The factors are Bandwidth (BW), Memory (MEM), and CPU cycle or Speed (CPU) of data center. This knowledge of finding DCLE is mentioned in terms of fuzzy inference rules which connect antecedents with consequences. A few definitions will be provided to demonstrate this perception model.

3.1. Preliminaries

Definition 1 (approximate reasoning). Fuzzy set corresponding to the linguistic values defined as A_1, B_1. We include a reasoning as multiconditional in the form

$$\begin{aligned}&\text{Rule 1: IF } x \text{ is } A_1, \quad \text{THEN } y \text{ is } B_1,\\ &\text{Rule 2: IF } x \text{ is } A_2, \quad \text{THEN } y \text{ is } B_2,\\ &\quad\vdots\\ &\text{Rule } n\text{: IF } x \text{ is } A_n, \quad \text{THEN } y \text{ is } B_n,\\ &\text{Fact: } x \text{ is } \overline{A},\\ &\quad\vdots\\ &\text{Conclusion: } Y \ is\ \overline{B}.\end{aligned} \tag{1}$$

Given n *If - then* Rule, rule 1 through n and a fact "x is $\overline{A}$". We conclude that "y is $\overline{B}$" where $\overline{A}$, $A_j \in f(x)$, $\overline{B}$, $B_j \in f(y)$ for all $j \in N_n$ and X, Y are sets of variables of x and y.

Definition 2 (fuzzy implication). In general, fuzzy implication ∂ is defined as the function of the form

$$\partial : [0,1] * [0,1] \longrightarrow [0,1]. \tag{2}$$

It gives any of possible true values a, b of given fuzzy propositions p, q, respectively, define the true value $\partial(a,b)$ of the conditional proposition called IF Then rules like "IF p, then q". This is called classical implication of $p \rightarrow q$, from the restricted domain $\{0,1\}$ to the full domain $[0,1]$ of true values in fuzzy logic deriving "∂" in classical formula being

$$\partial(a,b) = \overline{a} \vee b. \tag{3}$$

for all $a, b \in \{0,1\}$. We interpret disjunction and negation as a fuzzy union and fuzzy complement, and then ∂ in classical logic is to employ the formula

$$\partial(a,b) = \max\{x \in \{0,1\} \mid a \wedge x \leq b\}. \tag{4}$$

Moreover, equation (4) may also be rewritten, due to law of absorption of negation in classical logic, as either

$$\partial(a,b) = \overline{a} \vee (a \wedge b). \tag{5}$$

Definition 3 (relation "R"). The fuzzy relation R employed in reasoning is obtained from the given *if- then* rules in (2). For each rule j in (2), we determine a relation. R_j by the formula,

$$R_j(x,y) = \min\left[A_j(x), B_j(y)\right]. \tag{6}$$

for all $x \in X$, $y \in Y$, then R is defined by the unions of relations R_j for all rule in Definition 1 gives

$$R = \bigcup_{j \in N_n} R_j. \tag{7}$$

In this paper, consider the problem as disjunctive in nature. So, the interpretation of the rules in disjunctive can be returned as

$$B' = \bigcup_{j \in N_n} A' \cdot R_j. \tag{8}$$

In general, R_j may be determined by a suitable fuzzy implication mentioned in Definition 2 as

$$R_j(x,y) = \partial\left(A_j(x), B_j(y)\right), \tag{9}$$

a general counterpart of (2).

Definition 4 (fuzzy proposition). The proposition is measured in its ranges and true values. It depends on the matter of degree. So, each fuzzy proposition is uttered by a number in the element interval $[0, 1]$. We consider our model as conditional, and unqualified propositions, Propositions "P," of this type are expressed by the canonical form

$$P : \text{IF } x = A, \text{ THEN } y = B, \tag{10}$$

where x and y are variables whose values are in set X and Y, respectively. Finally, A and B are fuzzy sets on X and Y, respectively. The propositions may also be viewed as

$$\langle x, y \rangle \text{ is } R, \tag{11}$$

where R is a fuzzy set on $X * Y$ that is determined for each $x \in X$, $y \in Y$ by formula

$$R(x, y) = \partial [A(x), B(y)], \tag{12}$$

where ∂ denotes a binary operation on $[0, 1]$ representing a suitable fuzzy implication.

Definition 5 (compositional rule inference). Consider variables x and y that take values from sets X and Y, respectively, and assume that for all $x \in X$ and all $y \in Y$, the variables are related by a function $y = f(x)$, and x is in a given set A and y in a given set B is given by

$$B = \{y \in Y \mid \langle x, y \rangle \in R\}. \tag{13}$$

Similarly, since $x \in A$, we can infer that $y \in B$, where

$$B = \{y \in Y \mid \langle x, y \rangle \in R, x \in A\}. \tag{14}$$

Examine that this inference may be expressed equally well in terms of characteristics functions X_A, X_B, X_R of sets A, B, R, respectively, by the equation

$$X_B(y) = \sup_{x \in X} [X_A(x), X_B(x, y)], \tag{15}$$

for all $y \in Y$. Let us proceed now one step further and assume that R is fuzzy relation on $X * Y$ and $\overline{A}$, $\overline{B}$ are fuzzy sets on X and Y, respectively. Then, if R and $\overline{A}$ are given,

$$\overline{B} = \sup_{x \in X} [\overline{A}(x), R(x, y)], \tag{16}$$

for all $y \in Y$ which is the generalization of (7) obtained by replacing the characteristics functions in (7) with corresponding membership functions. We prefer this equation as generalization called compositional rule of inference to facilitate approximate reasoning.

4. Cloud Data Center Efficiency Prediction Using Fuzzy Expert System

Fuzzy controller is working as a feedback system by repeating the cycles to all and attaining a desired output. To establish the fuzzy controller modeling, first we have to define the input and output variables. Data center management is progressed by the DCLE (η) which is calculated among three factors Bandwidth (BW), Memory (MEM), and CPU Cycles (CPU). In our assumption, these three factors are considered as input variables and data center load efficiency as output variable. The solution is judged by data center management as control problem in nature. To define the load efficiency of data center is a single output variable of cloud environment. This system consists of three modules:

(i) fuzzification and defuzzification,

(ii) fuzzy inference engine,

(iii) fuzzy rule base.

First observations are done of all input and output variables, which mention conditions of the data center management control process. Then, these observations are converted into appropriate fuzzy set to propose observation uncertainties called fuzzification. To define the data center load efficiency η of a single variable inspite of bandwidth, memory, and CPU cycles, we consider the combinations of any two input variables d, $\overline{d}$ to be considered as bandwidth, CPU cycle, or memory. By utilizing these values, the fuzzy controller produces a control variable η that is DCLE. Linguistic variables and their notations are depicted in Table 1.

Table 1: Fuzzy linguistic values and notations.

Linguistic variables	Notation
Bandwidth(BW)	
Low	L
Medium	M
High	H
Memory (MEM)	
Small	S
Medium	M
Large	LA
CPU Utilization (CPU)	
Low	L
Medium	M
High	H
Data Center Load Efficiency (DCLE)	
Minimum	MN
Moderate	MD
Maximum	MX

4.1. Step 1. It is a process of identifying input/output variables and to assign a meaningful linguistic states and their ranges. To prefer exact linguistic states for each variable and pose them by corresponding fuzzy sets, these linguistic states are proposed as fuzzy sets (or) fuzzy numbers. Consider that the ranges of input variables d belongs to $[-a, a]$, $\overline{d}$ belongs to $[-b, b]$ and the range of output variable η belongs to $[-c, c]$. The linguistic input variables are Bandwidth, and Memory, CPU cycle, and output variable is Data Center Load Efficiency (DCLE). The ranges of the each input variables are having three linguistic states as shown in Figures 3 and 4. Also the output variable has three linguistic states.

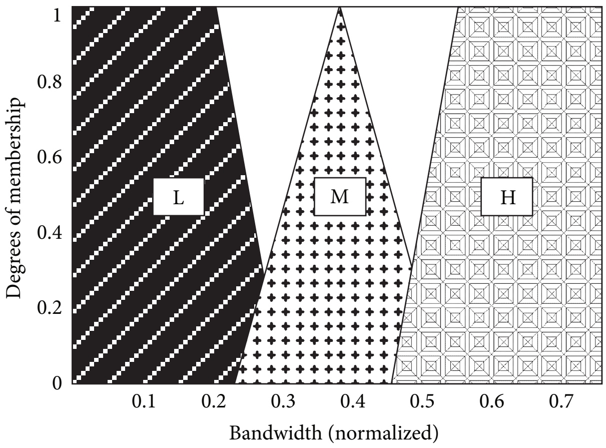


Figure 3: Fuzzy trapezoid view of bandwidth.

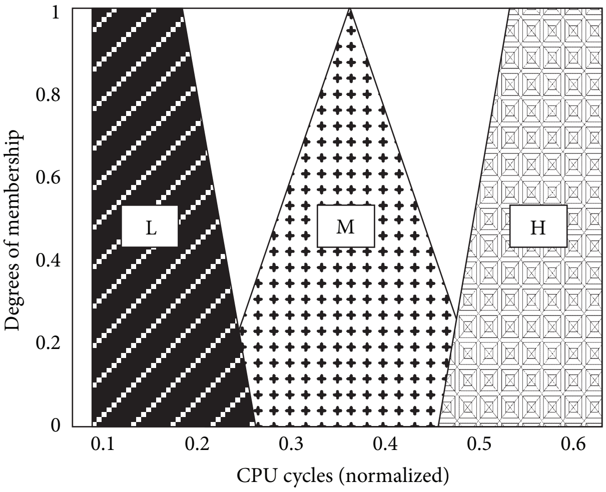


Figure 4: Fuzzy trapezoid view of CPU cycles.

4.2. Step 2. In this step, we introduce a fuzzification function for each input variable to propose the associate observation uncertainness. To find grades of membership of linguistic values of linear variable corresponding to an input number or fuzzy number, it is used to calculate and interpret observations of input variable, each expressed as a real number.

Consider a fuzzification function of the form

$$f_d : [-a, a] \longrightarrow R, \tag{17}$$

where R denotes the set of all fuzzy numbers and $f_d(x_0)$ is a fuzzy number chosen by f_d as approximation of the measurement $d = x_0$.

We introduced trapezoidal shape as membership function to define $f_d(x_0)$. It is showing the two control variables and their trapezoidal view to represent fuzzy numbers. We illustrate fuzzification by showing the membership function for Bandwidth and Memory together with a trapezoid view of variables depicted in Figure 5.

4.3. Step 3. Fuzzy inference system can be generated as relevant fuzzy inference rules by fuzzy associated memory called FAM square. They can be conveniently represented by Figures 6, 7, and 8 as a FAM square.

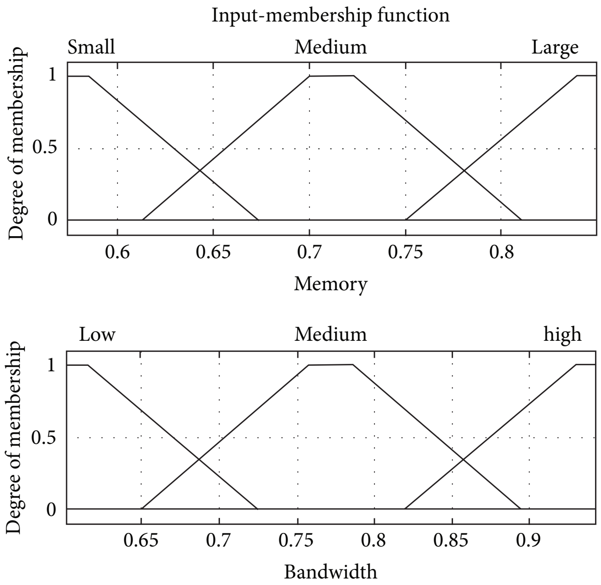


Figure 5: Input membership function of bandwidth, and memory (Normalized).

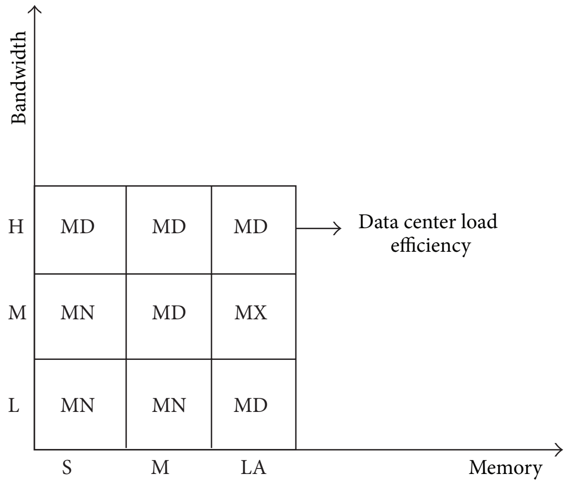


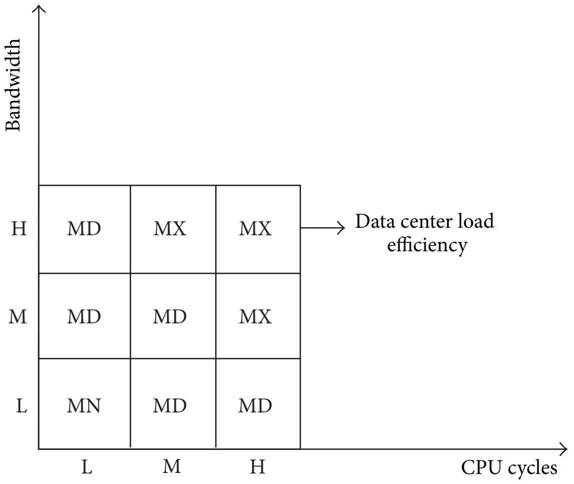
Figure 6: FAM square-rule 1.

In our approach, d, $\overline{d}$ are inputs, η is output variable, and then

$$\text{IF } d = A,\ \overline{d} = B,\ \text{THEN } \eta = C, \tag{18}$$

where A, B, C are fuzzy numbers chosen from the set of numbers and their linguistic states. The possible rule generated for each input and output variable is 3; so, $3^2 = 9$, and totally we have 36 rules. To find the fuzzy rules practically, we need a set of input-output data of the following:

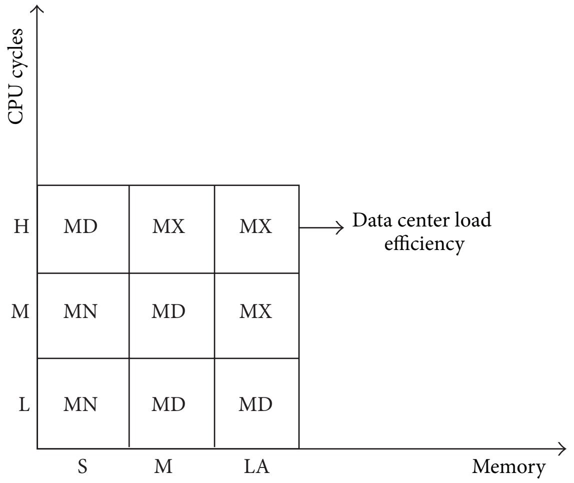
$$X \left\{ \left\langle x_k, y_k, z_k \right\rangle \mid k \in K \right\}, \tag{19}$$



Bandwidth			
H	MD	MX	MX
M	MD	MD	MX
L	MN	MD	MD
	L	M	H

CPU cycles
Data center load efficiency

Figure 7: FAM square-rule 2.



CPU cycles			
H	MD	MX	MX
M	MN	MD	MX
L	MN	MD	MD
	S	M	LA

Memory
Data center load efficiency

Figure 8: FAM square-rule 3.

where z_k is a attained value of output variable η for given value x_k and y_k of the input variable d and $\overline{d}$ respectively, K is an appropriate index set.

Let $A(x_k)$, $B(y_k)$, $C(z_k)$ denote the largest membership grades. Then the degree of relevance can be expressed by

$$i_1\left[i_2,\left(A\left(x_k\right),B\left(y_k\right)\right),C\left(z_k\right)\right], \tag{20}$$

where $i_1,\ i_2$ are t-norms.

Note. A function i: $[0,1]^2 \rightarrow [0,1]$ such that for all $a, b, d \in [0,1]$, $i(a,1) = a$; $b \leq d$ implies $i(a,b) \leq (a,d)$; $i(a,b) = i(b,a)$; $i(a,i(b,d)) = i(i(a,b),d)$.

The function is usually also continuous such that $i(a,a) \leq a$ for all $a \in [0,1]$.

4.4. *Step 4.*

The observation of input variable must be periodically matched with fuzzy inference rules to make inference in terms of output variables.

We choose composite inference logic mentioned in Definition 5 to define our variables. We convert the given fuzzy inference rules represented in (18) which are equivalent to simple fuzzy conditional proposition of the form

$$\text{IF } \left\langle d,\overline{d}\right\rangle \text{ is } A\times B,\ \text{THEN } \eta \text{ is } C, \tag{21}$$

where

$$[A\times B]\left(x,y\right)=\min\left[A\left(x\right),B\left(y\right)\right] \tag{22}$$

for all $x \in [-a,a]$ and $y \in [-b,b]$.

The output variable DCLE η becomes the problem of approximate reasoning with composite inference fuzzy proposition mentioned in Definitions 4 and 5, respectively. The fuzzy rule base consists of "n" fuzzy inference values; then,

$$\begin{aligned}
&\text{Rule 1: IF } \left(d,\overline{d}\right) \text{ is } A_1\times B_1,\ \text{Then } \eta \text{ is } C_1,\\
&\text{Rule 2: IF } \left(d,\overline{d}\right) \text{ is } A_2\times B_2,\ \text{Then } \eta \text{ is } C_2,\\
&\qquad\vdots\\
&\text{Rule } n\text{: IF } \left(d,\overline{d}\right) \text{ is } A_n\times B_n,\ \text{Then } \eta \text{ is } C_n,\\
&\text{Fact: } \left(d,\overline{d}\right) \text{ is } f_d\left(x_0\right)\times f_{\overline{d}}\left(y_0\right),\\
&\qquad\vdots\\
&\text{Conclusion: } \eta \text{ is } C.
\end{aligned} \tag{23}$$

The symbols A_j, B_j, C_j $(j = 1,2,\ldots n)$ denote fuzzy sets that represent the linguistic states of variables d, $\overline{d}$, η, respectively.

The rule is explained in terms of relation R_j, which is mentioned in Definition 2.

The rules are considered as disjunctive in nature. We derive (17) to conclude that the output variable η is defined by the fuzzy set as

$$C=\bigcup_j\left[f_d\left(x_0\right)\times f_{\overline{d}}\left(y_0\right)\right],o^iR_j, \tag{24}$$

where o^i is the sup-i composition for a t-norm i. The choice of the t-norm is a matter similar to the choice of fuzzy sets for given linguistic labels.

4.5. *Step 5.*

The process of computing single fuzzy number from C is called defuzzification. The fuzzy output variable is also a linguistic variable, whose values have been assigning grades of membership. In the last step, we find a single number compatible with membership function in Data Center Load Efficiency (DCLE) called output membership function depicted in Figure 9. This number will be the output from this final step in defuzzification process. There are several methods for calculating a single defuzzified number. We used a centroid method to convert the output values of inference engine as a crisp numbers expressed as fuzzy set. We

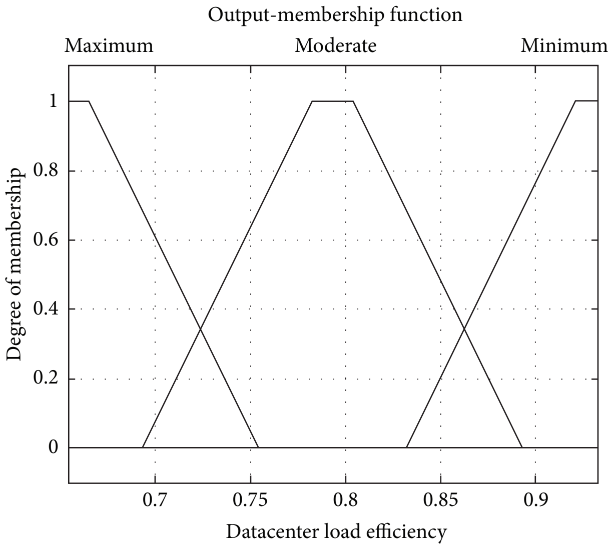


Figure 9: Output membership function DCLE.

calculated the output variable with centroid method which can be expressed as

$$x^* = \frac{\int_a^b \mu_A(x)\, x\, dx}{\int_a^b \mu_A(x)\, dx}. \quad (25)$$

Let $\mu_A(x)$ be the corresponding grade of membership in the aggregated membership function, and let

(1) $X_{\min}$ be the minimum x value attain the minimum of data center load efficiency η;

(2) X_{mod} the moderate x value attain the moderate of data center load efficiency η;

(3) $X_{\max}$ the maximum x value attain the maximum of data center load efficiency η.

X^* is defuzzified output as a real number value.

5. Performance Analysis

We now asses the performance of the proposed cloud data center efficiency using the Fuzzy Expert system model to show that they are load efficient. We will focus on the load efficiency of the data center in all the factors like bandwidth, memory, and CPU Cycles. The experiment is conducted using MATLAB Version 7.8 with an Intel Core 2 Processor running at 1.86 GHz, 2048 MB of RAM. Among the three key variables, namely, bandwidth, memory, and CPU cycles, the first step of the simulation focuses on fuzzification by converting them into input membership functions. This is performed using the tool called as membership function editor provided in the MATLAB. Each variable in the experiment is quantified into small, medium, and large for memory; low, medium, and high for bandwidth and CPU cycles. The input variables are segregated because the comparison of the variables becomes effective and it helps in providing better results. The If-Then rules of the experiment are formulated using rule editor.

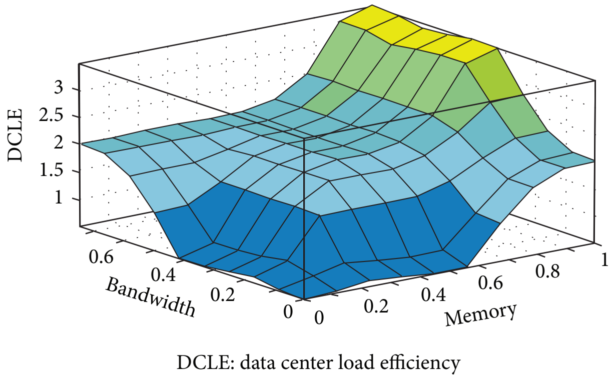


Figure 10: Fuzzy 3D view of bandwidth and memory versus DCLE.

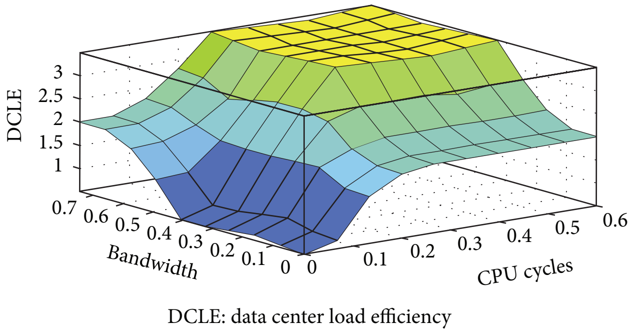
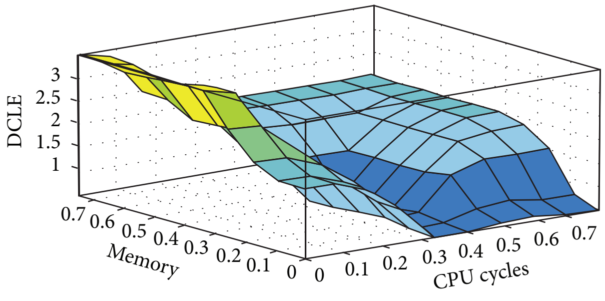


Figure 11: Fuzzy 3D view of bandwidth and CPU cycles versus DCLE.

We performed our required operation in FIS editor which handles the high level issues. The membership function editor which defines the shapes of all membership function is associated with each variable and rule editor for editing the list of rules. The surface viewer plots an output surface map for the system. The input vectors of the fuzzy inference engine as calculated by the simple attribute function are 0.812, 0.872, and 0.884, and the unique output generated by the Mamdani method is 0.959. All the rules have been depicted as 3D graphs called surface viewer in Figures 10, 11, and 12. Through Figure 10, we infer that when the bandwidth and memory linearly increase, the load efficiency of the data center increases at the same time, when they decrease, it brings down the efficiency of the data center linearly. In Figure 11, the Bandwidth and the CPU cycles are compared with the efficiency of the data center load. When the bandwidth and the CPU are higher, the efficiency of the data center is also higher and vice versa. In Figure 12, memory and CPU cycles are compared with the DCLE. The results indicate that when the memory and the CPU cycles are higher the DCLE is also higher, and lower in the opposite case. However, the experiments suggest that our system is more accurate in predicting the efficiency of a data center than a human expert. Here, DCLE is used as a prime factor in determining the overall system utilization and assessment of



DCLE: data center load efficiency

Figure 12: Fuzzy 3D view of memory and CPU cycles versus DCLE.

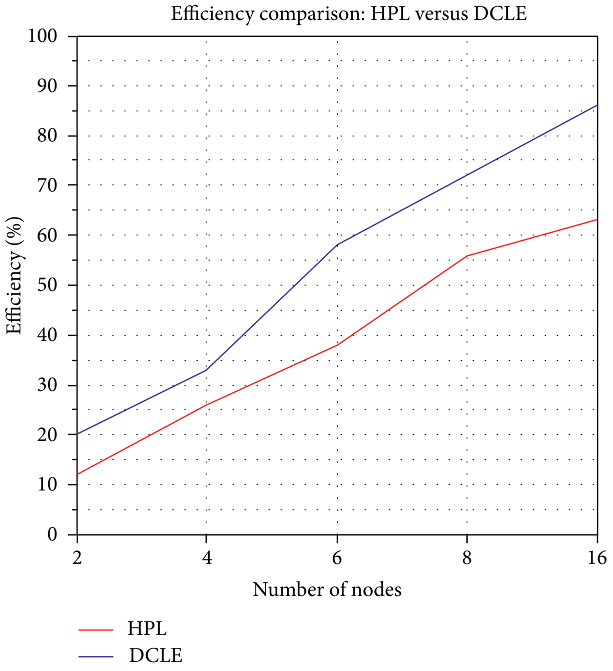


Figure 13: Efficiency Comparison.

the system efficiency. The results proved the increase in the number of services in the data center leading to increase in the complexity of the calculation in the DCLE. We list the features of our system Figure 13 and also make a comparison of our scheme with HPL performances (LINPACK Scheme) [44]. It was observed that they performed the experiment using the virtual clusters for GoGrid cloud service provider instances according to the varying number of nodes and percentage of efficiency. The efficiency is varied from 60 to 70. In this experiment, they consider bandwidth, memory, and processing cycles. It was observed that when the bandwidth, memory, and the CPU Cycles ranges were higher for the instances, this resulted in the increase in efficiency of the GOGrid instances. Whereas even when any of the three big factors were reduced, it impacted on the efficiency of the HPL system. The three big factors have been used to study the data center load efficiency, and it was observed the attribute values of the three factors when increased resulted in higher efficiency of any cluster or virtual systems. It is clearly evident that the simulation results are 20 percentage higher in comparison to the results offered by HPL systems.

6. Conclusion

The most important task in the successful service of the internet is access through maximum data center load efficiency. In this paper, we examined the load efficiency of data center, which is essentially needed for the cloud computing systems. This system is designed according to the service layers of cloud computing, cloud service provider estimating the strategy. Data center maintains a chart to monitor the big three factors suggested in this work. The advantage of the proposed system lies in DCLE computing. While computing, it allows regular evaluation of services to any number of clients. This work is extended in the way of providing resource adaptation and trustworthiness of cloud computing environment.

References

[1] C. Modi, D. Patel, B. Borisaniya, H. Patel, A. Patel, and M. Rajarajan, "A survey of intrusion detection techniques in cloud," *Journal of Network and Computer Applications*, vol. 36, no. 1, pp. 42–57, 2013.

[2] F. M. Aymerich, G. Fenu, and S. Surcis, "An approach to a cloud computing network," in *Proceedings of the 1st International Conference on the Applications of Digital Information and Web Technologies (ICADIWT '08)*, pp. 113–118, Ostrava, Czech Republic, August 2008.

[3] Enterasys Secure Networks, *Data Center Networking-Managing Virtualized Environment*, Enterasys Networks, Salem, NH, USA, 2011.

[4] I. Foster, Y. Zhao, I. Raicu, and S. Lu, "Cloud computing and grid computing 360-degree compared," in *Proceedings of the Grid Computing Environments Workshop (GCE '08)*, pp. 1–10, Austin, Tex, USA, November 2008.

[5] M. Jaiganesh and A. Vincent Antony Kumar, "JNLP based secure software as a service in cloud computing," *Communications in Computer and Information Science*, vol. 283, pp. 495–504, 2012.

[6] M. Armbrust, A. Fox, R. Griffth et al., "Above the clouds: a berkeley view of cloud computing," Tech. Rep. UCB/EESCS-2009-28, EECS Department; University of California, Berkeley, Calif, USA, 2009.

[7] M. Beckert, B. A. Ellison, and S. Krishnapura, "Intel it data center solutions: strategies to improve efficiency," IT@ Intel White Paper, Intel Information Technology, Intel Corporation, Santa Clara, Calif, USA, 2009, 1–11.

[8] P. Stryer, "Understanding data centers and cloud computing," Expert Reference Series of White Papers, Global Knowledge Training LLC, Cary, NC, USA, 2010, 1–7.

[9] R. Buyya, "Market-oriented cloud computing: vision, hype, and reality of delivering computing as the 5th utility," in *Proceedings of the 9th IEEE/ACM International Symposium on Cluster Computing and the Grid (CCGRID '09)*, pp. 1–13, Shanghai, China, May 2009.

[10] J. Varia, "Cloud architectures," White Paper, Amazon Web Services, Amazon Company, 2008, 1–14.

[11] L. Wang, J. Zhan, W. Shi, Y. Liang, and L. Yuan, "In cloud, do MTC or HTC service providers benefit from the economies of scale?" in *Proceedings of the 2nd ACM Workshop on Many-Task Computing on Grids and Supercomputers (MTAGS '09)*, pp. 7–11, November 2009.

[12] M. Jaiganesh and A. Vincent Antony Kumar, "ACDP: prediction of application cloud data center proficiency using fuzzy modeling," in *Proceedings of the International Conference on Modeling, Optimization and Computing, Procedia Engineering*, vol. 38, pp. 3005–3018, Elsevier Publications, 2012.

[13] B. J. S. Chee and C. Franklin, *Cloud Computing—Technologies and Strategies of the Ubiquitous Data Center*, CRC Press, Boca Raton, Fla, USA, 2010.

[14] C. Cattani, S. Chen, and G. Aldashev, "Information and modeling in complexity," *Mathematical Problems in Engineering*, vol. 2012, Article ID 868413, 3 pages, 2012.

[15] X. Jie, M. Li, W. Zhao, and S. Y. Chen, "Bound maxima as a traffic feature under DDOS flood attacks," *Mathematical Problems in Engineering*, vol. 2012, Article ID 419319, 20 pages, 2012.

[16] Y. Zheng, S.-Y. Chen, Y. Lin, and W.-L. Wang, "Bio-inspired optimization of sustainable energy systems: a review," *Mathematical Problems in Engineering*, vol. 2013, Article ID 354523, 12 pages, 2013.

[17] M. Kutare, G. Eisenhauer, C. Wang, K. Schwan, V. Talwar, and M. Wolf, "Monalytics: online monitoring and analytics for managing large scale data centers," in *Proceedings of the 7th IEEE/ACM International Conference on Autonomic Computing and Communications (ICAC '10)*, pp. 141–150, Washington, DC, USA, June 2010.

[18] P. Lu, S. Chen, and Y. Zheng, "Artificial intelligence in civil engineering," *Mathematical Problems in Engineering*, vol. 2012, Article ID 145974, 22 pages, 2012.

[19] L. A. Zadeh, "Fuzzy sets and systems," *International Journal of General Systems*, pp. 129–138, 1990.

[20] L. A. Zadeh, "Outline of new approach to the analysis of complex systems and decision processes," *IEEE Transactions on Systems, Man and Cybernetics*, vol. SMC-3, pp. 28–44, 1973.

[21] A. K. Lohani, R. Kumar, and R. D. Singh, "Hydrological time series modeling: a comparison between adaptive neuro-fuzzy, neural network and autoregressive techniques," *Journal of Hydrology*, vol. 442-443, pp. 23–35, 2012.

[22] E. H. Mamdani and S. Assilian, "Experiment in linguistic synthesis with a fuzzy logic controller," *International Journal of Man-Machine Studies*, vol. 7, no. 1, pp. 1–13, 1975.

[23] B. B. Mandelbrot, *Gaussian Self-Affinity and Fractals*, Springer, New York, NY, USA, 2001.

[24] M. Li and W. Zhao, "Abstract description of internet traffic of generalized Cauchy type," *Mathematical Problems in Engineering*, Article ID 821215, 18 pages, 2012.

[25] D. Veitch, N. Hohn, and P. Abry, "Multifractality in TCP/IP traffic: the case against," *Computer Networks*, vol. 48, no. 3, pp. 293–313, 2005.

[26] M. Li and W. Zhao, "Visiting power laws in cyber-physical networking systems," *Mathematical Problems in Engineering*, vol. 2012, Article ID 302786, 13 pages, 2012.

[27] P. Abry, R. Baraniuk, P. Flandrin, R. Riedi, and D. Veitch, "Multiscale nature of network traffic," *IEEE Signal Processing Magazine*, vol. 19, no. 3, pp. 28–46, 2002.

[28] M. Li and W. Zhao, "Representation of a stochastic traffic bound," *IEEE Transactions on Parallel and Distributed Systems*, vol. 21, no. 9, pp. 1368–1372, 2010.

[29] G. Terdik and T. Gyires, "Lévy flights and fractal modeling of internet traffic," *IEEE/ACM Transactions on Networking*, vol. 17, no. 1, pp. 120–129, 2009.

[30] A. Iosup, S. Ostermann, N. Yigitbasi, R. Prodan, T. Fahringer, and D. Epema, "Performance analysis of cloud computing services for many-tasks scientific computing," *IEEE Transactions on Parallel and Distributed Systems*, vol. 22, no. 6, pp. 931–945, 2011.

[31] R. Moreno-Vozmediano, R. S. Montero, and I. M. Llorente, "Multicloud deployment of computing clusters for loosely coupled MTC applications," *IEEE Transactions on Parallel and Distributed Systems*, vol. 22, no. 6, pp. 924–930, 2011.

[32] X. Dutreilh, N. Rivierre, A. Moreau, J. Malenfant, and I. Truck, "From data center resource allocation to control theory and back," in *Proceedings of the 3rd IEEE International Conference on Cloud Computing (CLOUD '10)*, pp. 410–417, Miami, Fla, USA, July 2010.

[33] Y. O. Yazir, "Visual assessment of cloud resource consolidation managers using convex hulls," in *Proceedings of the IEEE Eighth World Congress on Services*, pp. 293–300, Honolulu, Hawaii, USA, June 2012.

[34] M. Litoiu, M. Woodside, J. Wong, J. Ng, and G. Iszlai, "A business driven cloud optimization architecture," in *Proceedings of the 25th Annual ACM Symposium on Applied Computing (SAC '10)*, pp. 380–385, ACM, New York, NY, USA, March 2010.

[35] S. W. Liao, T. H. Hung, D. Nguyen, H. Zhou, C. Chou, and C. Tu, "Prefetch optimizations on large-scale applications via parameter value prediction," in *Proceedings of the 23rd International Conference on Supercomputing (ICS '09)*, pp. 519–520, Yorktown Heights, NY, USA, June 2009.

[36] A. Patel, M. Taghavi, K. Bakhtiyari, and J. C. Jnior, "An intrusion detection and prevention system in cloud computing: a systematic," *Journal of Network and Computer Applications*, vol. 36, no. 1, pp. 25–41, 2013.

[37] G. Reese, *Cloud Application Architectures: Building Applications and Infrastructure in the Cloud*, O'Reilly Media Publications, Gravenstein Highway North Sebastopol, Calif, USA, 2009.

[38] M. Li and W. Zhao, "Asymptotic identity in min-plus algebra: a report on CPNS," *Computational and Mathematical Methods in Medicine*, Article ID 154038, 11 pages, 2012.

[39] M. Malekzadeh, A. A. A. Ghani, and S. Subramaniam, "A new security model to prevent denial-of-service attacks and violation of availability in wireless networks," *International Journal of Communication Systems*, vol. 25, no. 7, pp. 903–925, 2012.

[40] The parallel Workloads archive Team, "The parallel Workloads Archive logs January," 2009.

[41] M. Li and W. Zhao, "A model to partly but reliably distinguish DDOS flood traffic from aggregated one," *Mathematical Problems in Engineering*, vol. 2012, Article ID 860569, 12 pages, 2012.

[42] M. Mirahmadi and A. Shami, "Traffic-prediction-assisted dynamic bandwidth assignment for hybrid optical wireless networks," *Computer Networks*, vol. 56, no. 1, pp. 244–259, 2012.

[43] H. S. Kim and N. B. Shroff, "The notion of end-to-end capacity and its application to the estimation of end-to-end network delays," *Computer Networks*, vol. 48, no. 3, pp. 475–488, 2005.

[44] The HPCC Team, "HPC challenge results," 2009.

[45] H. Liu and D. Orban, "GridBatch: cloud computing for large-scale data-intensive batch applications," in *Proceedings of the 8th IEEE International Symposium on Cluster Computing and the Grid (CCGRID '08)*, pp. 295–305, Lyon, France, May 2008.

[46] A. Iosup, H. Li, M. Jan et al., “The grid workloads archive,” *Future Generation Computer Systems*, vol. 24, no. 7, pp. 672–686, 2008.

[47] A. Rizk and M. Fidler, “Non-asymptotic end-to-end performance bounds for networks with long range dependent fBm cross traffic,” *Computer Networks*, vol. 56, no. 1, pp. 127–141, 2012.

[48] X. Jie, M. Li, and W. Zhao SY Chen, “Bound maxima as a traffic feature under DDOS flood attacks,” *Mathematical Problems in Engineering*, vol. 2012, Article ID 419319, 20 pages, 2012.

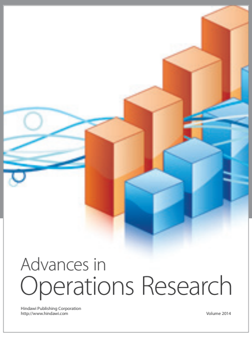
[49] M. Maurer, I. Brandic, and R. Sakellariou, “Self-adaptive and resource-efficient SLA enactment for cloud computing infrastructures,” in *Proceedings of the IEEE 5th International Conference on Cloud Computing (CLOUD)*, pp. 368–3375, 2012.

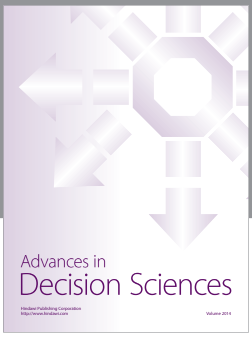
[50] M. Armbrust, A. Fox, R. Griffth et al., “Above the clouds: a berkeley view of cloud computing,” Tech. Rep. UCB/EESCS-2009-28, EECS Department, University of California, Berkeley, Calif, USA, 2009.

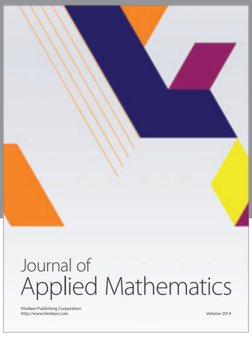
[51] L. Youseff, M. Butrico, and D. Da Silva, “Toward a unified ontology of cloud computing,” in *Proceedings of the Grid Computing Environments Workshop (GCE ’08)*, Austin, Tex, USA, November 2008.

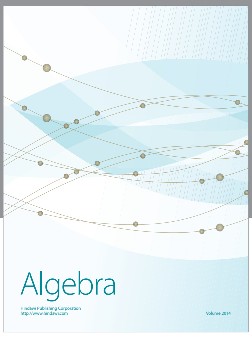
[52] M. Saleem Khan, “Fuzzy time control modeling of discrete event systems,” in *Proceedings of the World Congress on Engineering and Computer Science, International Association of Engineers (IAENG ’08)*, pp. 683–688, 2008.

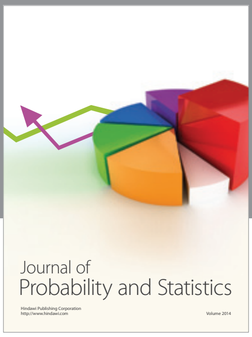
[53] W. Duch, R. Adamczak, and K. Gra bczewski, “A new methodology of extraction, optimization and application of crisp and fuzzy logical rules,” *IEEE Transactions on Neural Networks*, vol. 12, no. 2, pp. 277–306, 2001.


Advances in
Operations Research

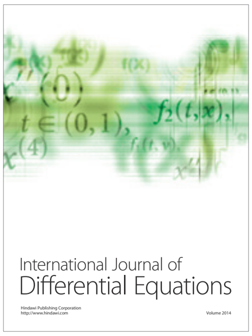

Advances in
Decision Sciences


Journal of
Applied Mathematics

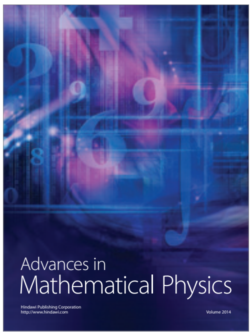

Algebra

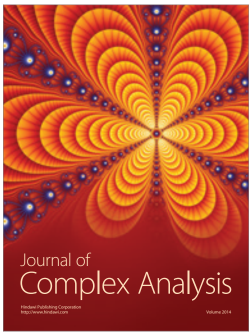

Journal of
Probability and Statistics

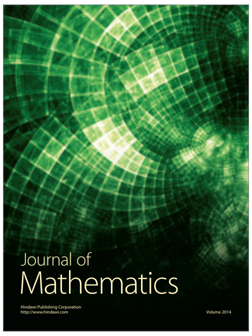
The Scientific
World Journal

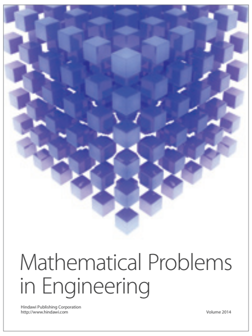

International Journal of
Differential Equations

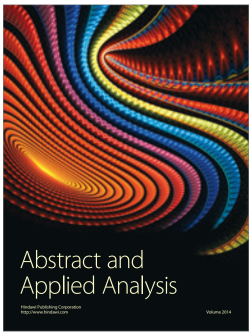
International Journal of
Combinatorics

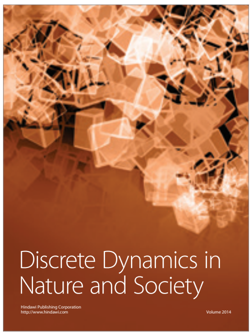

Advances in
Mathematical Physics

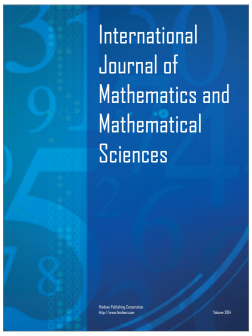

Journal of
Complex Analysis


Journal of
Mathematics


Mathematical Problems
in Engineering

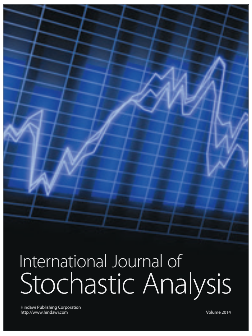

Abstract and
Applied Analysis


Discrete Dynamics in
Nature and Society


International
Journal of
Mathematics and
Mathematical
Sciences

Journal of
Discrete Mathematics

Journal of
Function Spaces


International Journal of
Stochastic Analysis

Journal of
Optimization