International Journal of Computer Applications (0975 – 8887)
Volume 29– No.10, September 2011

A Resource Pool Management Model using Fuzzy Logic Decision Making

Mya Sithu
University of Computer Studies, Yangon
Myanmar

Ni Lar Thein
University of Computer Studies, Yangon
Myanmar

ABSTRACT
In Virtualized Data Center (VDC), fixed allocation of virtual machines (VMs)' resources usually make overloading or underloading condition as the workloads of the application they run have dynamic features. Underloading leads unutilized resource usages and SLA deviation is also appeared because of the overloading. Therefore, dynamic allocation of VM resource usage is necessary in order to get a load balanced environment. Resource usage prediction of virtual machines is needed to be done for dynamic VM resource provisioning. For VMs, dynamic resource provisioning is available only when the host has enough resources to complement their demands. But in the condition of host overloading, VM migration should be made in order to keep a load balanced condition. VM migration depends on a variety of criteria and efficient decision support is required. In this paper, neuro-fuzzy is applied for resource usage prediction and combination of AHP and Fuzzy-TOPSIS is used for decision support of VM migration. VM resource prediction and decision support model are evaluated by the resource pools data. Simulation and mathematical evaluation confirms that such an integrated solution ensures acceptable accuracy.

General Terms
Virtualization, Resource Management

Keywords
Load balancing, Resource Management, AHP, Fuzzy-TOPSIS, Neuro-Fuzzy

1. INTRODUCTION
Virtualized data center (VDC) contains physical and virtual servers which serve a variety of services including web services, file services etc. The advantages of VDC is enabling application isolation since malicious or greedy applications cannot impact other applications co-located on the same physical server. Perhaps the biggest advantage of employing virtualization is the ability to flexibly remaps physical resources to virtual servers in order to handle workload dynamics.

Server resources in a data center are multiplexed across multiple applications and each server runs one or more applications [4]. These applications are usually business critical applications with Quality-of-Service (QoS) requirements. The resource allocation needs to not only guarantee that a virtual container always has enough resources to meet its application's performance goals, but also prevent over provisioning in order to reduce cost and allow the concurrent hosting of more applications. Static allocation approaches that consider a fixed set of applications and resources cannot be used because of changing workload mixes, and solutions that only consider behavior of individual applications fail to capture the competition for shared resources by virtualized containers [5]. For these reasons, dynamic resource allocation for VMs should be conducted to prevent SLA violations and unoptimized resource utilization.

Furthermore, each application sees dynamic workload fluctuations caused by incremental growth, time-of-day effects, and flash crowds [4]. In that case, although dynamic resource provisioning can solve the over provisioning and under provisioning of VM resources, the host must have enough resources in order to solve the demands of high workloads. Host may also be overloaded when it gets demands of additional resources from its virtual machines concurrently. Such condition can be handled by one of the biggest advantage of the ability of virtualization, namely, live migration. When virtual machines are needed to be migrated from one server to another intending load balancing, the migration controller needs to choose the proper VM to be migrated and destination server at a time. For a proper selection, the decision maker may need a large amount of data to be analyzed and many factors to be considered. Apart from the overloading cases, there may also be underloading situation when the average utilization of a host in the data center reaches below defined threshold. Migration can also be conducted for the case of underloading in which all VMs from the underloaded server are moved to appropriate destination servers and then it is shut down.

In this paper, a resource management model for virtualized data center is presented. Required virtual machine's resource usage prediction is performed with neuro-Fuzzy and decision support is made with the combination of AHP and fuzzy-TOPSIS. To our knowledge, resource usage prediction is conducted by several methods. But in our proposed model, neuro-Fuzzy system is chosen as it is capable of reasoning and learning in an uncertain and imprecise environment. Resource usage prediction can be performed via two methods, namely, resource-resource mapping and workload-resource mapping. In the workload-resource mapping approach, profiles of data center workloads and CPU usages are used as input and output pairs. In the resource-resource mapping approach, time series profiles of CPU usages are used for input and output data pairs. Data pairs are clustered according to their feature at first. Based on the clustered groups, the number of fuzzy membership functions and the number of rules are defined. With the concept of the neuro-fuzzy logic, the membership functions and associated rules are regarded as the layers of the neural network and then weights of the connections of the layers are adjusted based on the errors in the training phase. Workload-resource mapping can be used in the environment where application workloads can be monitored and characterized. When the former method is not applicable because of different types of application and workload natures, the latter can be used.

In decision support of virtual machine migration, four criteria are used for VM selection and six criteria are used for target server selection. Several methods exist for multi-criteria decision making (MCDM) problems. Among them, AHP and fuzzy TOPSIS are chosen and presented in this paper. AHP is used to check the consistency of the weights of the criteria and TOPSIS is used for alternative ranking. Instead of original TOPSIS, fuzzy-TOPSIS is chosen in this model as it can deal with decision maker ambiguities, uncertainties and vagueness, which cannot be handled by crisp values. From the numerical evaluation, we can see that the combination of AHP and fuzzy-TOPSIS can help the data center administrator for decisions of virtual machine migrations when they are selecting the proper virtual machine and target server.

The rest of the paper is structured as follows. Section 2 presents the related work of this paper. Section 3 is about the proposed resource management model in virtualized data center. Section 4 will explain about virtual machine resource prediction system. It will be followed by decision support for live migration in section 5. Section 6 concludes the paper and references are shown in Section 7.

2. RELATED WORK

All This section presents the related work that is considered for resource usage predictions of virtual machines and virtual machine migrations. Sandpiper [11] automates the task of monitoring and detecting hotspots, determining a new mapping of physical to virtual resources and initiating the necessary migrations. But Sandpiper did not take into account the complicated and uncertain relationship between the system's parameters. P.Paddala [10] presents their system as automated control of multiple virtualized resources. Auto control is a resource control system that automatically adapts to dynamic workload changes to achieve application service level objectives (SLO). The model estimator captures the complex relationship between application performance and resource allocations, while the MIMO controller allocates the right amount of multiple virtualized resources to achieve application SLOs. In online model estimator, they used adaptive modeling approach to capture the complex behavior of enterprise applications.

Jing Xu [5] proposed a two level resource management system to dynamically allocate resources to individual virtual containers. It uses local controllers at the virtual container level and a global controller at the resource pool level. They used fuzzy logic and migration decision is not considered. In [12], VMware's Distributed Resource Scheduler solves the CPU and memory pressure by performing load balancing dynamically. But VMware's DRS cannot utilize application logs to have better placement decisions. Moreover, DRS is only efficient for homogeneous virtualized environment.

In [8], M.Tarighi presents a method to migrate VMs between cluster nodes using TOPSIS algorithm to find the most loaded server. In their system, they combine the TOPSIS algorithm and fuzzy theory. In their first level of implementation, they order the physical servers and decide which the most overloaded ones are. Then the second level of operation starts by ordering the virtual machines. After completing the ordering of two steps, then migration decisions are made to move the most overloaded virtual machines from the most overloaded physical machines to the least overloaded machine. But they did not consider the consistency checking of the weights of the criteria.

3. PROPOSED RESOURCE MANAGEMENT MODEL

The proposed model is intended for the resource management of virtualized data center and operates as shown in Figure 1. Although the model is intended for the real world virtualized data center, simulation and mathematical evaluations here in this paper are based on a single resource pool which contains five physical servers with the aim of simplicity. As initial allocation of virtual machines to resources is another interesting challenge area, virtual machines in this resource pool are assumed to be already assigned to the physical servers. For VM allocation in physical hosts, fixed allocation, in other words, worst case provisioning is not good because most of the time resources are allocates unnecessary. Therefore, with the purpose of good resource utilization, dynamic allocation should be used to handle the resource provisioning according to the dynamic fluctuated pattern of workloads. Therefore, the proposed model monitors VM resource usages in the resource pool at every five minutes interval, estimates the resource usage of the next time interval and make re-provisioning. At a host, the VMs demand required resource from the host it is existed on according to their workload. As all the VMs on a single host are running different applications that have different workload patterns, it will be difficult for the host they are existed on when all of its VMs demands concurrently. There may occur an unbalanced situation that one host is overloading and another host is underloading. Load unbalancing and SLA deviation can be overcome by migrating VMs from the overloaded to underloaded one. With the help of the benefit of virtualization technology, the virtual machines can be migrated without user interruption. For both of the cases of preventing SLA deviation and aiming of high resource utilization, underload and overload thresholds are needed to be regarded at a physical machine. In this proposed model, 70% is regarded as the overload threshold and 40% is as the underload threshold. These threshold levels are defined based on the history profiles of workload patterns of this resource pool. In the underloading case, if a host workload is under 40%, it seems that resource utilization is not as good as already expected and if there are other available hosts that can accept all of its virtual machines, they should be moved to these servers and it should be shut down with the purpose of power safety. Although the live migration has no user interruption, it should be aware of migration overhead and possibility of occurring ping-pong effect because of unnecessary migration. To limit such unnecessary overheads, the proposed model calculates which destination server is the most optimal one for the migrated server. To know the loads of the servers, the proposed decision support model monitors the status information of all virtual and physical servers in a resource pool periodically and makes migration decision. In that way, a load balanced environment can be built. The VM resource usage prediction system will sit on every virtual machine and VM resource provisioning is made according to its prediction. As for decision support system, it will monitor system every five minutes and ranks physical machines and virtual machines. Migration decision will be made if it finds one of the virtual or physical machines is overloading or underloading.

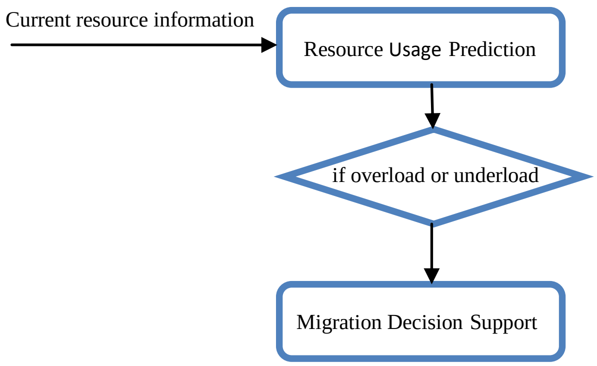

Figure 1. Proposed Resource Management Model

4 VIRTUAL MACHINE RESOURCE PREDICTION SYSTEM

Virtual machine resource prediction system can be performed via two methods, namely, workload-resource and resource-resource mapping. In workload-resource mapping, the proposed model predicts the next time interval resource usage of the virtual machines according to the nature of the workload on that virtual machine. Although the proposed model is available for all dimensions of resources, in this paper, the relationship between workload and CPU usage is only considered for simplicity. By learning the nature of the relationship between the workload and CPU usage, it is found that the CPU usage is always getting higher whenever the workload is increasing. Rules are initially generated by clustering the data pairs of one week log history.

In second method, instead of multiple dynamic resources, CPU resources are used for simplicity. In this model, next time interval of CPU resource usage is predicted based on the first three consecutive intervals of CPU usages as shown in Figure 2. Both methods are modeled using neuro-fuzzy system.

4.1 Neuro-Fuzzy

The neuro-fuzzy controller uses the neural network learning techniques to tune the member-ship functions while keeping the semantics of the fuzzy logic controller intact. Neural networks offer the possibility of solving the problem of tuning. Although a neural network is able to learn from the given data, the trained neural network is generally understood as a black box. Neither it is possible to extract structural information from the trained neural network nor can we integrate special information into the neural network in order to simplify the learning procedure. On the other hand, a fuzzy logic controller is designed to work with the structured knowledge in the form of rules and nearly everything in the fuzzy system remains highly transparent and easily interpretable. However, there exists no formal framework for the choice of various design parameters and optimization of these parameters generally is done by trial and error. A combination of neural network and fuzzy logic offers the possibility of solving tuning problems and design difficulties of fuzzy logic [9]. The resulting network will be more transparent and can be easily recognized in the form of fuzzy logic control rules or semantics. This neuro-fuzzy system combines the well established advantages of both the methods and avoids the drawbacks of both [9]. For those reasons, we chose neuro-fuzzy system to use in our proposed resource provisioning model for virtual machine controllers.

In both methods, firstly, the data pairs are clustered to study and generate the fuzzy rules needed to control the input without compromising the quality of control. After dividing the data space into fuzzy clusters, each representing one specific part of the system behavior. After projecting the clusters onto the input space, the antecedent parts of the fuzzy rules can be found. The consequent parts of the rules can then be simple functions. In this way, one cluster corresponds to one rule of the TSK model [1]. The TSK model is composed of the IF-THEN rules of the following form.

$$R_{(r)}: \quad if\ x_1\ is\ A_r^1\ and\ A_r^2\ and\ \ldots and\ x_m\ is\ A_r^m$$

$$then\ y_r\ is\ fr(x) \qquad (1)$$

$$where\ f_r(x) = \alpha_r^0 + \alpha_r^1 x_1 + \cdots + \alpha_r^m x_m \qquad (2)$$

in which $(r_m = 1, \ldots, n)$ and $(x_j (1 \leq j \leq m)$ are the input variables, y_r is the output variable, A_r^m are fuzzy sets, and f_r(x) is a linear function [1]. Gaussian membership functions are chosen for this model.

After initializing an initial fuzzy inference structure (FIS), combination of the least-squares method and the back propagation gradient descent method are used to tune the membership functions and rules of the initial FIS. For each cluster, a TSK rule is constructed in the rule layer. Epoch numbers '10', error goal '0', step size increase rate '1.1' and step size decrease rate '0.9' are chosen for the tuning purpose. Absolute error '0' is achieved for this type of approach.

4.2 Two Approaches of Fuzzy-Prediction

In this section, workload-resource mapping and resource-resource mapping, two approaches of fuzzy prediction are explained.

4.2.1 Workload-Resource Mapping

To simulate workload-resource mapping, the nature of the workload history of the web server in the resource pool is learned. One week workload history of that server workload is taken as input, clustered, trained and rules are generated. Based on these rules, real time input data are taken as inputs to evaluate whether it can predict correctly the next time interval usage or not. In normal condition, the server has similar frequency pattern in their workloads. For example, in a business company, Monday morning and Friday evening usually has higher load than the rest of the time. But in the lunch time of weekdays, lower frequency patterns are found and there is no load on weekends. Because of this cyclic frequency nature of workloads under normal condition, neuro-fuzzy sytem is used to train and produce rules. But under unusual conditions, the system will face a very different frequency pattern. For example, the web server of the world weather forecasting center has the high CPU usage when disaster occurs in a part of the world. At that condition, the trained pattern that based on normal frequency is not sufficient for the server. To cope with this condition, 24 hour interval is chosen as a training interval. After every 24 hours, the system takes the history of previous 24 hours load profile and re-trains it again.

4.2.2 Resource-Resource Mapping

In the second proposed approach, resource-resource mapping, required information about the resource usage is easy to obtain by monitoring system level metrics. Future resource needs are determined on the basics of observations of past resource usage using neuro-fuzzy system. The operations of resource- resource mapping is similar to workload- resource mapping. The fuzzy rules representing a mapping from input space to output space are generated from the monitored data and stored in the rule base. The learned fuzzy rules are processed by the fuzzy inference system to predict future resource demands based on the current system observations. At sampling time t, the latest m measurements u(t), u(t −1),..., u(t −m+1) are taken as the inputs and determine the resource needs at future times u(t +1), u(t +2),...,u(t +n) as the outputs (m and n are the number of inputs and outputs for a fuzzy rule, respectively). In this approach, the data pairs which are taken as input to neuro-Fuzzy system contain four columns. The first three columns, first three consecutive CPU usages are regarded as inputs and the fourth column, the fourth time interval of CPU resource usage is regarded as output. The inputs are first three consecutive intervals and the output is the fourth time interval of CPU usage. If any two of records have same data in first three columns and different data in the last column, it means that these records are conflicting. Because of the conflicting records in the data set, ambiguities occur when the system try to make predictions. If the data set does not contain any conflicting records, less difference between desired value and actual value can be achieved.

5 DECISION SUPPORT OF VIRTUAL MACHINE MIGRATION

For both of the cases of overloading and underloading in the host, virtual machine live migration is needed to perform in order to address SLA deviation and load unbalancing. In this proposed model, the combination of AHP and Fuzzy-TOPSIS is used to make decision support of these migrations. If more criteria are considered in decision making, the more computational overhead it can take. But, on the other hand, insufficient number of criteria can cause inefficient decisions. Therefore, the decision maker is needed to balance the trade-off between number of criteria and the computational overhead. Six criteria are used for physical server ranking and four criteria are used for virtual machine ranking. For mathematical evaluation, resource information is collected from a resource pool and it can be seen that the proposed model can give accurate decision. AHP is used to check the consistency of the criteria to know whether they are consistent or not. For example, if criteria A is 3 times more preferable than criteria B and criteria B is 2 times more preferable than criteria C, then it can be assumed that criteria A is 6 times preferable than criteria C. If it is not, inconsistencies occur. If they are not consistent, it will be affected on the rankings of fuzzy-TOPSIS as the contents of priority vectors are used in the calculation of weighted fuzzy matrix.

5.1 AHP

AHP is a process for developing a numerical score to rank each decision alternative based on how well each alternative meets the decision maker's criteria. With the AHP, the objectives, criteria and alternatives are arranged in a hierarchical structure similar to a family tree. A hierarchy has at least three levels: overall goal of the problem at the top, multiple criteria that define alternatives in the middle, and decision alternatives at the bottom [7].

5.2 Fuzzy-TOPSIS

The TOPSIS method is a technique for order preference by similarity to ideal solution [2]. The best alternative is the one that is closest to the ideal solution and farthest from the negative ideal solution. Suppose a MCDM problem with m alternatives, A1... Am, and n decision criteria/attributes, C1... Cn. Each alternative is evaluated with respect to m criteria/attributes. All the values/ratings assigned to the alternatives with respect to each criterion form a decision matrix denoted by X = (xij) $_{nm}$. W = (w1... wn) be the relative weight vector for the criteria [6]. General TOPSIS process with six activities is listed below [3].

Activity 1

Establish a decision matrix for the ranking. The structure of the matrix can be expressed as follows:

$$D = \begin{bmatrix} f_{11} & \cdots & f_{1n} \\ \vdots & \ddots & \vdots \\ f_{m1} & \cdots & f_{mn} \end{bmatrix} \quad (1)$$

Activity 2

Calculate the normalized decision matrix $R(= [r_{ij}])$. The normalized value r_{ij} is calculated as:

$$r_{ij} = \frac{f_{ij}}{\sqrt{\sum_{j=1}^{n} f_{ij}^2}} \quad (2)$$

where j = 1…n; i = 1…m.

Activity 3

Calculate the weighted normalized decision matrix by multiplying the normalized decision matrix by its associated weights. The weighted normalized value V is calculated as:

$$v_{ij} = w_{ij} r_{ij} \quad (3)$$

where w_j represents the weight of the jth attribute or criterion.

Activity 4

Determine the PIS and NIS, respectively:

$$V^+ = \{v_1^+, \ldots, v_n^+\} = \{(Max\ v_{ij} \mid j \in J), (Min\ v_{ij} \mid j \in J')\quad (4)$$

where J is associated with the positive criteria and J' is associated with the Negative criteria.

Activity 5

Calculate the separation measures, using the m dimensional Euclidean distance. The separation measure D_i^+ of each alternative from the PIS is given as:

$$D_i^+ = \sqrt{\sum_{j=1}^{n} (v_{ij} - v_j^+)^2}, \text{ i=1,….m} \quad (5)$$

Similarly, the separation measure D_i^- of each alternative from the NIS is as follows:

$$D_i^- = \sqrt{\sum_{j=1}^{n} (v_{ij} - v_j^-)^2}, i = 1, \ldots m \quad (6)$$

Activity 6

Calculate the relative closeness to the idea solution and rank the alternatives in descending order. The relative closeness of the alternative A_i with respect to PIS V^+ can be expressed as:

$$\bar{C}_i = \frac{D_i^-}{D_i^+ + D_i^-} \quad (7)$$

where the index value of $\bar{C}_i$ lies between 0 and 1. The larger the index value, the better the performance of the alternatives.

The use of fuzzy set theory allows the decision-makers to incorporate unquantifiable information, incomplete information, non-obtainable information and partially ignorant facts into decision model. As a result, fuzzy TOPSIS and its extensions are developed to solve ranking and justification problems [7].

6 Numerical Evaluation of Decision Support

To make a numerical evaluation, we used the state information of the resource pool of a virtualized data center that is presented in [8]. This resource pool has five physical hosts and each of them has some virtual machines. As Table 1 indicates it has total 12 virtual machines. According to the state information as shown in Table 2, we can see that physical machine 3 is the most overloaded server. For the sake of load balancing, virtual machines of PM3 are needed to check to know which one should be migrated away.

Table 1. Physical machines and associated virtual machines

Physical Machine	Virtual Machine
PM1	VM1,VM2
PM2	VM3
PM3	VM4,VM5,VM6,VM7
PM4	VM8,VM9,VM10
PM5	VM11,VM12

In the first stage of the proposed model, alternative VMs of the overloaded server and the criteria are determined and the decision hierarchy is formed. AHP model is structured such that the objective is in the first level, criteria are in the second level and alternative virtual machines are on the third level. Four criteria are determined according to the data center's administrator experience and they are shown in Table 3.

Table 2. Physical machines and their state information

Physical Machine	CPU %	RAM%	NET %	CPU Clock Speed	RAM Capacity	Network Band-width
PM1	15	32	13	2	2	100
PM2	0	2	0	1.2	1	1000
PM3	81	60	40	1.8	2	100
PM4	70	49	85	3.2	1	1000
PM5	53	70	16	2.4	6	100

Table 3. Virtual machines and their state information

VirtualMachine	CPU%	RAM%	NET%	RAM Usage
VM1	15	23	13	0.6
VM2	0	0	0	0
VM3	60	67	58	0.4
VM4	54	56	72	1

6.1 VM Selection

In this resource pool, most of the servers are running CPU intensive applications. Therefore, CPU usage percentage is the most preferable criteria for the data center's administrator.

6.1.1 Consistency Checking in VM Selection

The pairwise comparison matrix of the criteria is shown in Table 4. Priority vector and their consistency check are shown in Table 5, 6 and 7. The weights of the criteria are needed to be consistent. For example, if criteria A is 3 times more preferable than criteria B and criteria B is 2 times more preferable than criteria C, then it can be assumed that criteria A is 6 times preferable than criteria C. If it is not, inconsistencies occur. If they are not consistent, it will be affected on the rankings of fuzzy-TOPSIS as the contents of priority vectors are used in the calculation of weighted fuzzy matrix.

Table 4. Pairwise comparison matrix

	C1	C2	C3	C4
C1	1	3	5	2
C2	0.5	1	2	0.5
C3	0.2	0.5	1	0.5
C4	0.5	2	2	1

Table 5. Weighted pairwise comparison matrix

	C1	C2	C3	C4
C1	0.455	0.462	0.5	0.5
C2	0.227	0.154	0.2	0.125
C3	0.0909	0.077	0.1	0.125
C4	0.227	0.308	0.2	0.25

Table 6. Normalized principal Eigen vector

Criteria	Eigen Values
C1	0.479
C2	0.177
C3	0.098
C4	0.246

Table 7. Values of $\lambda_{max}, CI, RI, CR$

$\lambda_{max} = 4.168$ ($Eigen\ Value$)
CI = 0.0561 (Consistency Index)
RI = 0.9 (Random Index)
CR = 0.062<0.1 (Consistency Ratio)

6.1.2 Finding Proper VM in VM Selection

Weights of the criteria chosen for VM selection are consistent as shown in Table 7. After then, ranks of the virtual machines are determined by using fuzzy TOPSIS method. The triangular fuzzy numbers related with these variables are shown in Table 8.

Table 8. Linguistics values and fuzzy numbers

Fuzzy Membership Regions	
Very Low (VL)	(0,0,0.2)
Low (L)	(0.1,0.2,0.3)
Mol Low (ML)	(0.3,0.4,0.5)
Medium (M)	(0.4,0.5,0.6)
Mol High (MH)	(0.5,0.6,0.7)
High (H)	(0.6,0.7,0.9)
Very High (VH)	(0.8,1,1)

Established Fuzzy evaluation matrix of alternatives VM by linguistic variables is shown in Table 9. Table 10 consists of the triangular fuzzy numbers which are equivalent of Linguistic variables. After the fuzzy evaluation matrix was determined, fuzzy weighted decision table is obtained. Using the criteria weights calculated by AHP, the weighted evaluation matrix is established with Eq. (3). According to the values of weighted matrix, it is seen that the elements are belonged to the closed interval [0, 1]. Thus, we can define the fuzzy positive solution (FIPS, A^*) and the fuzzy negative-ideal solution (FINS, A^-) as $\breve{v}_i^*$= (1,1,1) and $\breve{v}_i^-$= (0,0,0) for benefit criterion, and $\breve{v}_i^*$= (0,0,0) and $\breve{v}_i^-$=(1,1,1) for cost criterion. In this case, criteria 1, 2 and 3 are defined as benefit criteria and criterion 4 is regarded as cost criteria. Criterion 4, RAM usage by each VM is defines as cost criterion because virtual machine which has the lowest RAM usage is the optimal virtual machine that should be migrated. The more memory pages are used by the VM, the more migration time it will take.

Table 9. Fuzzy evaluation matrix by linguistic variables

	C1	C2	C3	C4
VM1	Low	Mol Low	Low	High
VM2	Very Low	Very Low	Very Low	Very Low
VM3	High	High	Mol High	Medium
VM4	Mol High	Mol High	High	Very High

The distance of each alternative from D^* and D^- can be calculated using Eq. (5) and Eq. (6).Then similarities to the ideal solution are calculated using Eq. (7).

Table 10. Fuzzy evaluation matrix by triangular fuzzy numbers

	C1	C2	C3	C4
VM1	(0.1,0.2,0.3)	(0.3,0.4,0.5)	(0.1,0.2,0.3)	(0.6,0.7,0.9)
VM2	(0,0,0.2)	(0,0,0.2)	(0,0,0.2)	(0,0,0.2)
VM3	(0.6,0.7,0.9)	(0.6,0.7,0.9)	(0.5,0.6,0.7)	(0.4,0.5,0.6)
VM4	(0.5,0.6,0.7)	(0.5,0.6,0.7)	(0.6,0.7,0.9)	(0.8,1,1)

The CC_j values are summarized in Table 11. Based on the CC_j values, the ranking of the alternatives are made in the descending order VM3, VM4, VM1 and VM2. If the weight of the criteria has not consistency, the appropriate VM that should be migrated can change.

Table 11. Fuzzy-TOPSIS result for D_j^*, D_j^- and CC_j

Alternatives	D^*_j	D^-_j	CC_j
VM1	2.997	1.018	0.253
VM2	2.979	1.071	0.264
VM3	2.588	1.424	0.355
VM4	2.767	1.241	0.309

Table 12. Weighted ranking

Rank	Weighted CC_j	Weighted Rank
1	0.355	VM3
2	0.309	VM4
3	0.264	VM2
4	0.253	VM1

6.2 Target Server Selection

After deciding which virtual machine is needed to migrate, then the proposed model finds the appropriate target server for the migrated virtual machine by measuring with the defined criteria. The same steps of calculation are taken in target server selection. Six criteria are used for this case as shown in Table 13. In this case, three criteria, CPU usage percentage, RAM usage percentage and NET usage percentage are defined as benefit criteria and CPU cycle, RAM capacity, Net BW of the physical host are defined as cost criteria.

Table 13. Criteria for target server selection

Criteria	Definition	Type
C1	CPU%	Benefit
C2	RAM%	Benefit
C3	NET%	Benefit
C4	CPU Clock Speed	Cost
C5	RAM Capacity	Cost
C6	Net Bandwidth	Cost

In this phase, the lowest rank is regarded as the least loaded server. Therefore, it is chosen as the destined server for migration. The last three criteria are used as cost criteria because the physical machine with the higher value of them is capable for higher load. Most loaded server is in the top rank and the least loaded server is defined as the optimal server for migrated VMs. Step by step calculations by using AHP are shown in Table 14, 15 and 16 .

6.2.1 Consistency Checking in Target Server Selection

Table 14. The pairwise comparison matrix for criteria

	C1	C2	C3	C4	C5	C6
C1	1	3	5	1	3	5
C2	0.33	1	2	0.33	1	2
C3	0.2	0.5	1	0.2	0.5	1
C4	1	3	5	1	3	5
C5	0.33	1	2	0.33	1	2
C6	0.2	0.5	1	0.2	0.5	1

Table 15. Weighted normalized pairwise comparison criteria

	C1	C2	C3	C4	C5	C6
C1	0.0327	0.333	0.313	0.0327	0.333	0.313
C2	0.108	0.111	0.125	0.108	0.111	0.125
C3	0.065	0.056	0.063	0.065	0.056	0.063
C4	0.0327	0.333	0.313	0.0327	0.333	0.313
C5	0.108	0.111	0.125	0.108	0.111	0.125
C6	0.065	0.056	0.063	0.065	0.056	0.063

Table 16. Values of $\lambda_{max}, CI, RI, CR$

$\lambda_{max} = 6.0049$ ($Eigen\ Value$)
CI = 0.002 (Consistency Index)
RI = 1.24 (Random Index)
CR = 0.002<0.1 (Consistency Ratio)

Consistency ratio of the pairwise comparison matrix is calculated as 0.002 < 0.1. So the weights are shown to be consistent and they are used in the selection process.

6.2.2 Finding Proper Target Server

Table 17. Fuzzy evaluation matrix by linguistic variables

	C1	C2	C3	C4	C5	C6
PM1	L	ML	L	M	M	M
PM2	VL	VL	VL	M	M	H
PM3	VH	H	MH	M	M	M
PM4	H	MH	VH	VH	ML	H
PM5	MH	H	Low	VH	VH	M

When Fuzzy-TOPSIS is used for the target server selection according to the collected data, the final result is shown that Physical host 3 is the most overloaded server and Physical 2 is the least loaded server. Therefore, in this case Physical host 2 is chosen as the destined server. Step by step calculations by using Fuzzy-TOPSIS are shown in Table 17, 18, and 19.

Table 18. Fuzzy-TOPSIS result for D_j^*, D_j^- and CC_j

Alternatives	D^*_j	D^-_j	CC_j
PM1	3.163	2.848	0.474
PM2	3.235	2.794	0.463
PM3	2.845	3.164	0.527
PM4	3.034	2.976	0.495
PM5	3.161	2.849	0.475

Table 19. Weighted ranking

	Weighted CCj	Weighted Ranks
1	0.527	PM3
2	0.495	PM5
3	0.475	PM5
4	0.474	PM1
5	0.463	PM2

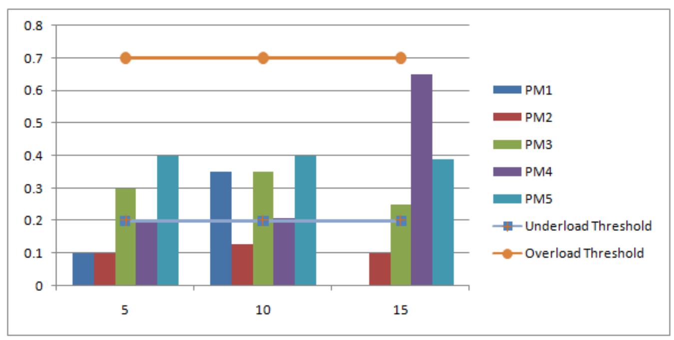

Figure 2. Underloading occurs at PM2

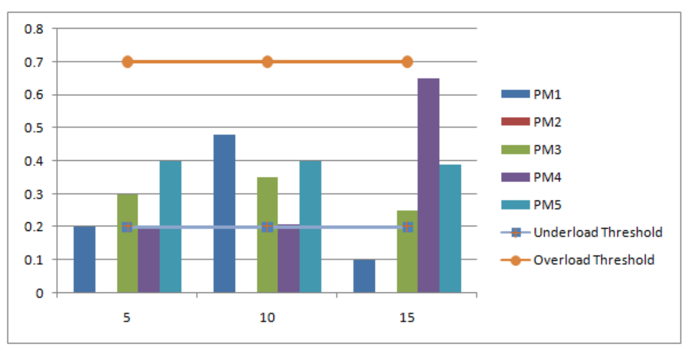

Figure 3. Load Balanced Condition Supported by Proposed Resource Management Model

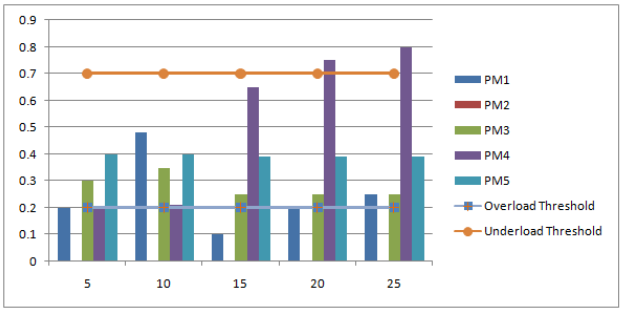

Figure 4. Overloading occurs at PM4

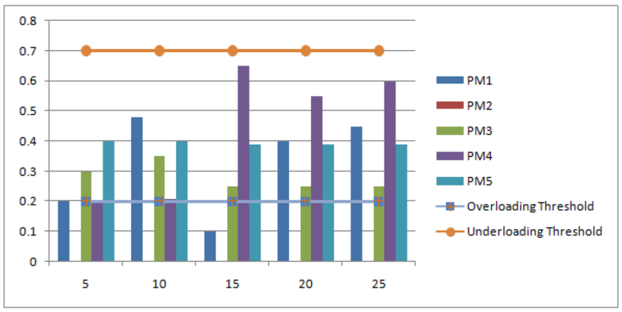

Figure 5. Load Balanced Condition Supported by Proposed Resource Management Model

7. CONCLUSION AND FUTURE WORK

This paper presents resource management model containing virtual machine resource usage prediction and decision support of virtual machine migration which use neuro-fuzzy and fuzzy-TOPSIS respectively. The proposed resource prediction model is able to provide much lower resource allocation costs than worst-case provisioning. To enable to accurately estimate the resource demands of virtual machines, two fuzzy logic based methods, workload-resource mapping and resource-resource mapping are proposed to guide resource allocation. Specifically, the work-load-resource method characterizes the mapping from the work-loads and the corresponding resource requirements, while the fuzzy prediction builds a mapping from current resource usage to future resource needs. In order to adapt to the system changes and reflect the most recent system conditions, re-training and rule addition is performed. In the case of resource shortage or poor utilization of host, reassignment of a virtual machine to another physical host is required. The migration decision involves many parameters that are interrelated in the changes in some parameters affect the others. In the case of virtual machine decision support system, virtual machine and physical server selection are performed in a fuzzy environment and uncertain linguistic value of variables. Fuzzy-TOPSIS is a viable method for the proposed problem and is suitable for the use of linguistic variables. When the decision making condition is vague and inaccurate, then this method is the preferred technique. Moreover, to know whether the criteria used for decision making is consistent or not, AHP is applied before the ranking process. The present study explored the use of the combination of AHP and Fuzzy TOPSIS in finding the most critical machines and the least one.

Future papers will consider the cost of virtual machine migration and the profit model of the data center for the case of the time when there is limited additional resource for the demanding virtual machines. The proposed model does not include the migration cost (i.e., the cost is assumed to be zero). This assumption is reasonable in the scenario where the data center's physical resources reside in one location and they have adequate network bandwidth so that the migration could take place in less than a few seconds. However, in the case of scale-out data centers where the migrations may happen between different locations, the migration delay could be significant. The cost due to the resource overhead and reallocation delay should be incorporated to consider.

8. REFERENCES

[1] B.Kosko.Neural Networks and Fuzzy Systems. First Edition, Prentice Hall of India, 1997.

[2] C.L. Hwang and K. Yoon. Multiple attribute decision making: Methods and applications. Berlin: Springer. 1981.

[3] D. L. Olson, "Comparison of weights in TOPSIS models", Mathematical and Computer Modeling, 40, (7) 21-727, 2004.

[4] I.Mevag.Towards Automatic Management and Live Migration of Virtual Machines. Thesis of Oslo University, May 2007.

[5] J.Xu, M.Zhao, J.Fortes, R.Carpenter and M.Yousif.Autonomic Resource Management in virtualized data centers using fuzzy logic-based approaches. In Proc. IEEE, July 2008.

[6] K.Begnum and M.Disney. Decision Support for Virtual Machine Re-Provisioning in Production Environments. 21st Large Installation System Administration Conference, LISA' 07.

[7] M.Dagdviren, S.Yavuz and N.Kilinc. Weapon Selection using AHP and TOPSIS Methods Under Fuzzy Environments. In Journal of Elsevier, Expert Systems with Applications, 2009. pp.8143-8151.

[8] M.Tarighi, S.A.Motamedi, S.Sharifian. A New Model for Virtual Machine Migration in Virtualized Cluster Server Based on Fuzzy Decision Making. In Journal of Telecommu nications, Volume 1, Issue 1, February 2010.

[9] P. J. Werbos, "Neuro control and fuzzy logic: connections and design," Int. J. Approximate Reasoning, Vol. 6, Feb. 1992, pp. 185-220.

[10] P.Padala, X.Zhu, M.Uysal and Z. Wang. Automated Control of Multiple Virtualized Resources. In Proc. EuroSys'09 ACM, April 2009.

[11] T.Wood, P.Shenoy,A.Venkataramai, and M.Yousif.Black Box and Gray Box Strategies for Virtual Machine Migration . Computer Networks Journal Special Issue on Virtualized Data Center 2009.

[12] VMWare,Inc.DRS Performance and Best Practices, 2008.