Auto-Scalability in Cloud: A Surveyof Energy and Sla Efficient Virtual Machine Consolidation

[1]A.Richard William, Dr.J.Senthilkumar[2]

[1]*Asst. Prof. CSE, Jayalakshmi Institute of Technology*
[2]*Professor IT, Sona College of Technologies*

Abstract

In cloud computing, the modern cloud data centers are hosting a variety of advanced applications and the IT infrastructure over the recent years because of the demand for computational power infrastructure which are widely used by some of the applications increasing rapidly. Due to the enormous amount of electrical energy consumed by the huge cloud data centers, the operating cost and the emission of carbon dioxide (Co2) produces the high value as a result. In order to reduce the energy consumption and to increase the physical resource utilization in data centers, the most effective way used is a dynamic consolidation of virtual machines (VMs). The main purpose of this paper is to provide a novel method which is used in dynamic virtual machine consolidation. This proposed novel method has outperformed the existing policies in terms of energy consumption, SLA violation and VM migration time by surveying the determination of under loaded hosts, determination of overloaded hosts, and selection of VM and placement of the migrating VMs.

Keywords-- *cloud computing, consolidation, energy consumption, SLA violation*

I. INTRODUCTION

Cloud computing is a kind of distributed computing that refers to both the applications delivered as services over the Internet and the hardware and systems software in the data centers that provide the services such as Infrastructure as a Service (IaaS), Platform as a service (PaaS), Software as a Service (SaaS). It provides the ability to quickly meet the business demands and one can get all the benefits of their application, data, and storage requirement without investing in the infrastructure.

Cloud platforms have to plan and provide resources in a faster manner so as to satisfy huge amounts of tasks. The major goal is to make sure that the requirements of users are being fulfilled properly with less power consumption and cost. Hence several mechanisms are implemented to characterize and forecast workload for a period of seconds or minutes[2]. Based on the future workload prediction, performance of each VM (Virtual Machine) is determined in advance and resources are scaled accordingly. The estimate includes the fraction of capacity to be assigned to each VM and the number of requests effectively served which ensures cost minimization and service quality. This provision is widely used for real-time control functions, capacity planning, resource allocation and datacenter energy saving to predict the effects of adding and removing resources in cloud computing environment. With the help of effective prediction of workload, system administrators might take necessary actions to prevent the system from damage caused by high load.

The future workload can be predicted with the help of various machine learning techniques[2]. Machine learning is a form of artificial intelligence in which an application can learn from processing real data using algorithms to enhance predictability and make necessary arrangements for unexpected outcomes. Companies such as Google, Amazon, Microsoft integrates machine learning algorithms with their cloud services for the ability to predict the future for both tactical and strategic purposes. Developers can build learning capabilities into their own applications with the help of machine learning techniques. It integrates many distinct approaches such as reinforcement learning, probability theory, combinatorial optimization, control theory.

Prediction of workload is very essential for better performance of the system. Depending on the predicted workload, the resources are to be scaled properly. Generally, there are two types of methods of scaling namely horizontal scaling and vertical scaling[3]. When the system finds a higher utilization exceeding the upper threshold value, the horizontal scaling or the vertical scaling can be executed.

Horizontal scaling deals with the adjustment of VM instances and provides a larger scale resource. It takes few minutes to boot a VM. Horizontal scaling is suitable for applications that have a clustered framework in which a master node will distribute requests among the worker nodes which are represented as VM in cloud environment. The reconfiguration cost varies among applications and this kind of scaling is suitable for enterprise clouds. Vertical scaling deals with changing the partition of resources inside a VM and it can scale resources in a few milliseconds. Most of the hypervisors go for on-line VM resizing without shutting down the VM. Live migration increases the scope of vertical scaling because a scaled VM can be provided with additional resources by migrating other VMs in the server. Vertical scaling is widely used for dynamic consolidation in datacenters.

The individual benefits of horizontal scaling and vertical scaling may enhance the performance of a system but are limited in certain situations. Horizontal scaling takes a while to complete and can scale the application at higher cost. Vertical scaling has lower resource and configuration costs which may impact the performance of the applications running at higher utilization rate. Hence an auto-scaling approach is designed which is the combination of both horizontal scaling and vertical scaling to find a optimal scaling strategy[1]. Auto-scaling supports cloud computing providers for providing access to hardware which can be allocated and de-allocated at any time.

The rest of the paper is organized as follows. Section II presents the various machine learning algorithms for forecasting the workload. Section III deals with design of an auto-scaling approach based on the predicted workload. Finally, the paper is concluded by a summary.

II. WORKLOAD PREDICTION

Different machine learning algorithms such as linear regression, neural networks and support vector machine can be used to predict future workload which can be used to improve the performance of the system.

A. Neural Network

Generally, Neural Network can be defined as a computer network system, which is designed based on the fundamental concepts of human brain and nervous system. It consists of interconnected layers and the first layer is the input layer where inputs are presented. This input layer is connected to the next layer namely the hidden layer. The actual computation is performed in the hidden layer. The hidden layer is connected with the output layer which produces the output. A simple Neural Network is shown in Fig. 1.

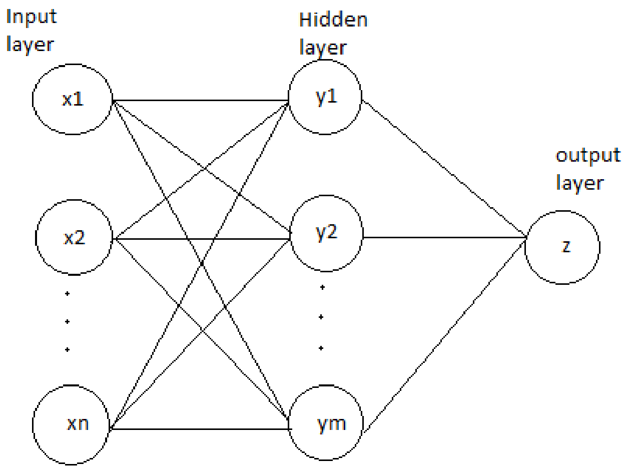

Fig. 1 Neural Network

Multilayer neural network perceptions are used to predict the future workload of applications. Perception is an algorithm that process the given inputs and makes its predictions. Perceptions are trained using back propagation algorithm in a supervised manner. The back propagation algorithm consists of a forward pass and a backward pass. The synapse weights are considered as static in forward pass and dynamic in backward pass. Dynamic updation of weights is based on the error correction rule and finally it minimizes the error[4].

In back propagation algorithm[5], the previous workload values are presented as input to the neural network. The weights of the edges from input layer to hidden layer and from hidden layer to output layer are set in a random manner. Once the output is produced from the neural network, it is compared with the desired output and error is being computed. This error is then given back to the neural network and the weights are adjusted to decrease the error value with each iteration so that the neural model will produce the desired output.

B. Linear Regression Model

The linear regression model is an approach to study the relationship between a dependent variable and an independent variable[6]. The linear regression model is of the generic form(1).

$$Y_i = a + bX_i \qquad (1)$$

Where Y_i is the dependent variable for observation i, X_i is the independent variable for observation i and a and b are coefficients. For future workload prediction, Y is the workload and X is the time. The coefficients are calculated with the help of linear regression equation solved based on previous workloads such as Y_{i-1}, Y_{i-2}, Y_{i-3} and so on. These values may change with various previous workloads. The workload trend is linear and it is shown in the Fig. 2.

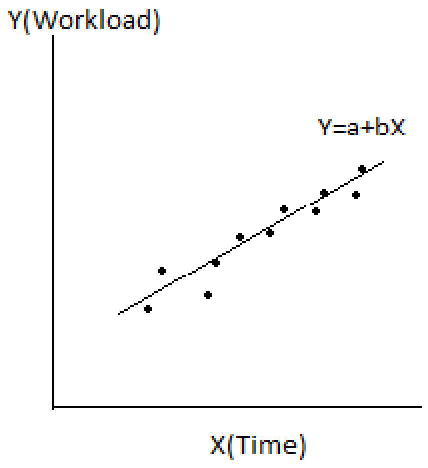

Fig. 2 Linear Regression Model

The values of the coefficients a and b can be solved using Cramer's Rule as shown in the Equation (2) and (3).

$$a = \frac{\sum X_i^2 \sum Y_i - \sum X_i \sum X_i Y_i}{n\sum X_i^2 - (\sum X_i)^2} \qquad (2)$$

$$b = \frac{n\sum X_i Y_i - \sum X_i \sum Y_i}{n\sum X_i^2 - (\sum X_i)^2} \qquad (3)$$

The known previous workloads can be substituted into Equation (2)and (3) to calculate the coefficients and then the workload is solved at the next interval using Equation (1).

C. Support Vector Machine

Support Vector Machine (SVM) is one of the supervised machine learning algorithms which is mainly used for pattern recognition, object classification, and regression analysis in case of time series prediction. Generally, the major aim of time series prediction is to calculate the future value based on current and past data sample values. Support Vector Regression (SVR) is the methodology which is used to estimate a function using observed data and it helps to train the SVM. The goal of SVR is to find a function that has at most ε deviation from the actual target for all training data with as much flatness as possible[7].

With the training data(x_i ,y_i) (i=1...l), where x is an n-dimensional input with x ϵ R^n and the output is y ϵ R, the linear regression model can be written in the form(4). (4)

$$f(x) = \langle w, x \rangle + b, w, x \in R^n, b \in R$$

where f(x) is the target function and <.,.>represents the dot product in R^n.To achieve flatness mentioned by [9], we minimize w i.e. $||w||^2 = \langle w,w \rangle$ and it can be written as a convex optimization problem.

$$\text{Minimize } \frac{1}{2}||w||^2$$

$$\text{subject to } \begin{cases} yi - \langle w, xi \rangle - b \leq \varepsilon \\ \langle w, xi \rangle + b - \text{yi} \leq \varepsilon \end{cases} \qquad (5)$$

Equation (5) assumes that there is always a function f that approximates all pairs of (x ,i) with ε precision. But this may not be attainable and hence [9] introduces slack variables γi, $\gamma i*$ to manage infeasible constraints, with Equation (5) leading to

Minimize

$$\frac{1}{2}\| \text{w} \|^2 + \text{C}\sum_{\text{i}=1}^{\text{n}}(\gamma_{\text{i}} + \gamma_{\text{i}}*)$$

$$\text{subject to } \begin{cases} \text{yi} - \langle w, xi \rangle - \text{b} \leq \varepsilon + \gamma\text{i} \\ \langle w, xi \rangle + b - \text{yi} \leq \varepsilon + \gamma\text{i}* \\ \gamma\text{i}, \gamma\text{i}* \geq 0 \end{cases}$$

(6)

The constant C>0 will determine the trade-off between the flatness of f and the amount up to which the deviations larger than ε are tolerated. Equation (6) can be re-constructed and solved so that the optimal Lagrange multipliers $\alpha and \alpha*$ can be given with w and b in the Equation (7) and (8).

$$w = \sum_{i=1}^{n}(\alpha - \alpha^*)x_i \qquad (7)$$

$$b = -\frac{1}{2} < w, (x_r + x_s) \quad (8)$$

where$x_r and x_s$ are support vectors, hence inserting (7) and (8) into (4) gives

$$f(x) = \sum_{i=1}^{n} (\alpha - \alpha^*) < x_i, x > +b \quad (9)$$

This approach is extended for nonlinear functions and this can be done by replacing $x_i$with (x_i), where a feature space that linearizes the relation between xi and yi [8].Thus, Equation (9) can be re-written as:

$$f(x) = \sum_{i=1}^{n} (\alpha - \alpha^*) K < x_i, x > +b$$

(10)

where$K$<x_i,x> = <$\varphi(x_i)$, $\varphi(x)$ is called as kernel function.

III.AUTO-SCALING APPROACH

Generally, scaling is a process to add resources when the system detects a higher system utilization which exceeds the upper threshold or to remove resources when there occurs a lower system utilization. The auto-scaling approach mainly consists of pre-scaling and real-time scaling and this kind of scalability can be achieved with the help of vertical scaling and horizontal scaling methods.

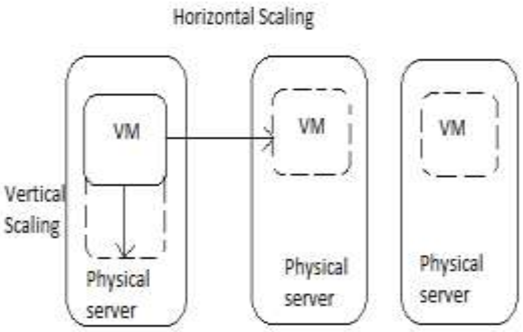

Fig. 3 Vertical and Horizontal scaling

Vertical scaling includes two types of scaling techniques namely self-healing scaling and resource-level scaling[10]. The basic idea behind self-healing scaling is that when two VMs of a service are allocated, resources of VMs may complement each other. For example, one or more CPUs assigned to a VM which has low CPU utilization rate can be removed to another VM with already saturated existing CPU resources. In resource-level scaling up, unallocated resources available at a particular cluster node (physical machine) are used to scale up a VM executing on it. For example, a cluster node with low CPU and memory utilizations can allocate these types of resources to one of VMs executing on it thus scaling it up. Horizontal scaling includes VM-level scaling which deals with the adjustment of VM instances. Fig. 3 show horizontal scaling and vertical scaling.

Both self-healing and resource-level scaling methods can scale up and down resources in few milliseconds. The resources can be scaled with less cost because the self-healing scaling isfree of scaling cost and hence it is executed first. Then the resource-level scaling is performed.

Horizontal scaling provides a large scale resource, but there arises a problem that it will take few seconds or minutes to complete the scaling process. This may cause more user SLA violations. The vertical scaling will be completed in less time but it is limited by scale. These two situations decreases the performance of the system. Hence it would be beneficial if the scaling process is performed earlier than the time when the workload actually increases.

Machine learning algorithms such as Linear regression model, Neural Networks and Support Vector Machine can be used to predict the future workload of different services. Then the auto-scaling approach will be executed. The pre-scaling method of this approach will scale the resources based on the predicted workload from the algorithms. Due to abnormal increase or decrease of workloads in cloud, there may be deviation in prediction. The workload that has been predicted may be lower than the actual workload, hence the resources scaled at the previous interval may be not enough to satisfy requests. This leads to decrease in the performance of the system. To overcome this situation, real-time scaling method is executed to minimize the impact. Hence the auto-scaling approach can scale resources with less cost and better performance in cloud environment.

IV.CONCLUSION

Scalability is one of the key features of cloud computing. The resources can be scaled up to accommodate increased business needs or changes.

The scaling process is performed based on the predicted workload. Various machine learning algorithms such as Neural Networks, Linear Regression Model, Support Vector machine are used to forecast the workload at the next interval with the help of previous workloads. An auto-scaling approach which consists of real-time scaling and pre-scaling methods is designed to scale depending on the workload prediction. The integration of workload prediction methods with auto-scaling process will effectively improve the performance in the cloud environment in terms of both performance and cost.

REFERENCES

[1]Jingqi Yang, Chuanchang Liu, Yanlei Shang, Bo Cheng, Zexiang Mao, Chunhong Liu, LishaNiu, Junliang Chen, "A cost-aware auto-scaling approach using the workload prediction in service clouds",Information System Frontiers,2014.

[2] Samuel A. Ajila, Akindele A. Bankole,"Cloud Client Prediction Models Using Machine Learning Techniques", In Proceedings of 2013 IEEE 37th Annual Computer Software and Applications Conference.

[3] Wang, W., Chen, H., Chen, X, "An availability-aware approach to resource placement of dynamic scaling in clouds", In Proceedings of the 2012 IEEE 5th international conference on cloud computing (pp. 930–931).

[4] John J. Prevost, KranthiManojNagothu, Brian Kelley, Mo Jamshidi, "Prediction of Cloud Data Center Networks Loads Using Stochastic and Neural Models", In Proceedings of the 2011 IEEE 6th international conference on System of Systems Engineering.

[5] ProdipGhosh, Sudip Das, DebabrataSarddar, "Future Load Prediction of Cloud Data Center using Neural Networks", International Journal of Latest Trends in Engineering and Technology, 2015

[6] Baltagi, B.H. (1998). Econometrics (pp. 41-69). Berlin: Springer

[7]Smola, A.J and Scholkopf, B., “A Tutorial on Support Vector Regression” in Statistics and Computing vol 14, pp. 199 – 222, 2004.

[8]Chih-Wei, H. et al, “A practical guide to support vector classification”. Technical report, Department of Computer Science and Information Engineering, National Taiwan University, Taipei, 2003.

[9] Dashevskiy, M. and Luo, Z. "Time series prediction with performance guarantee". IET Communications. Vol. 5, Issue 8, pp. 1044–1051, 2010.

[10] Han, R., Guo, L., Ghanem, M.M., Guo, Y. (2012). Lightweight resource scaling for cloud applications.In Proceedings of the 12th IEEE/ACM international symposium on cluster, cloud and grid computing (pp. 644–651).