International Journal of Psychosocial Rehabilitation, Vol. 19, Issue 02, 2015

ISSN: 1475-7192

POWER MANAGEMENT AND AUTOMATIC RESOURCE PROVISIONING FRAMEWORK FOR CLOUD INFRASTRUCTURES BASED ON MACHINE LEARNING

[1]Nageswara Rao Rayapati, Assistant Professor, Dept. of C.S.E, Mallareddy Engineering College, Maisammaguda, Hyderabad
[2]Sreekanth Thullibilli, Assistant Professor, Dept. of C.S.E, Malineni Lakshmaiah Engineering College, Singarayakonda, Prakasam District

ABSTRACT: *Power is becoming an increasingly important concern for large-scale cloud computing systems. Meanwhile, cloud service providers leverage virtualization technologies to facilitate service consolidation and enhance resource utilization. In a virtualized environment, resource needs to be configured at runtime at the cloud, server and virtual machine levels to achieve high power efficiency. In addition, cloud power management should guarantee high users' SLA (service level agreement) satisfaction. This paper provides power management and automatic resource provisioning framework for cloud infrastructures based on machine learning. It is a proactive technique for power management and auto-scaling of resources that changes the number of resources for the private cloud dynamically based on system load is proposed. To configure cloud resources, we consider machine learning techniques to achieve automatic resource allocation and optimal power efficiency. The technique that supports both on-demand and advance reservation requests uses machine learning. Experimental results demonstrate that the proposed technique can effectively estimate the power usage of cloud servers and reduction of cost for the client enterprise. Additionally resource configuration approach achieves the lowest energy usage among the compared three approaches.*

KEYWORDS: *Power management, Resource allocation, Cloud server, virtual machines.*

I. INTRODUCTION

Power and energy consumption has become a critical design factor in modern cloud computing systems and data centers, because it directly affects the operation expense in the total cost of ownership (TCO) [1]. Recent studies show that the operation expenses for cooling and powering largescale data centers will soon outstrip the acquisition.

Existing auto-scaling techniques are often based on indicators for resources like CPU usage, storage usage and network traffic [2]. However, it is hard to accurately select the scaling indicators and thresholds, especially when the application models are complex, and the resource utilization indicators are limited and very low-level [3]. The system introduced in this paper does

not use such indicators but uses machine learning to predict future resource demands for performing auto-scaling operations.

One of the most important reasons for energy inefciency in data centers is the idle power wasted when servers run at a low load. Even at a very low utilization, such as 10% CPU usage, the power consumed is over 50% of the peak power. Dynamic consolidation has proven to be an effective technique for power reduction in data centers by turning off idle or under-utilized servers. However, achieving the desired level of Quality of Service (QoS) between user and a data center is critical. Therefore, the dynamic consolidation can save energy while maintaining an acceptable QoS [4].

The QoS requirements are formalized via Service Level Agreement (SLA) that describes such characteristics as minimal throughput, maximal response time or latency delivered by the deployed system. Service level agreements (SLAs) are contracts between a service provider and its clients. SLAs in general depend on certain chosen criteria, such as service latency, throughput, availability, security, etc. Moreover, virtualization is the most popular power management and resource allocation technique used by a data center. It allows a physical server (host) to be shared among multiple Virtual Machines (VMs) where each VM can run multiple application tasks [5]. The CPU and memory resources can be dynamically provisioned for a VM according to the current resource requirements. This makes virtualization perfectly t for the requirements of energy efficiency in a data center.

The private cloud may be managed by an intermediary enterprise providing resources on demand to users in an external client organization. Alternatively this management can be performed by the IT department that serves as an intermediary in a large organization and provides resources on demand to the various departments within the company. Multiple users submit requests to a broker (in the intermediary enterprise). The broker is responsible for allocation of resources to requests as well as for performing auto-scaling operations. The broker auto-scales the resources proactively using a prediction system based on a machine learning technique and computes the current number of resource required for handling the upcoming requests while ensuring that a profit is generated for the intermediary cloud provider.

Traditional power measuring approaches cannot be applied to VMs because it is not feasible to connect physical power sensors or meters to VMs. Instead, we can attempt to estimate the power usage of virtualized systems by correlating the values of hardware and software performance metrics with the power consumption. Virtualization enables a richer set of performance metrics from hardware, hypervisor, VM, and cloud application, which can be explored to make power usage estimation with desired accuracy.

II. LITERATURE SURVEY

Sadeka, I. et al, [6] analyzed the problem of resource provisioning from the application provider's point of viewpoint so that the hosted applications can make scaling decisions based on future resource usage. They also employed a set of machine learning techniques – NN and LR with both sliding and non-sliding window option. Though they reported impressive prediction accuracy (PRED 252) of about 85.7% using NN, they did not report about the testing of the trained prediction model. It is not uncommon to have impressive training model prediction accuracy and poor test prediction accuracy.

G. Dhiman et. al. [7] an online learning algorithm is proposed that dynamically selects different experts to make power management decisions at runtime, where each expert is a predesigned power management policy. Different experts outperform each other under different workloads and hardware characteristics. A. Verma et. al. [8] presents pMapper a power-aware application placement controller in virtualized heterogeneous systems for minimizing power consumption and migration cost at each time frame.

Wood, T. et al [9] focused on estimating the resource requirements an application would require in a virtual environment from utilization traces collected in the application's native environment. They applied a tracebased approach of historical data to forecast future CPU usage as a means of capacity planning. The mapping of native to virtual environment was hinged on a high correlation and proportionality between the two environments. They included various workload mixes in their work, something a typical enterprise application would exhibit. Using CPU traces from TPC-W and RUBiS applications, they employed Linear Regression to forecast future CPU utilization. While the authors reported a prediction error of less than 5% in the 90th percentile,

they also used only CPU as a metric and reported their inability to include response time in their prediction model.

N. Bobroff et. al. [10] a dynamic server migration is described to improve the amount of required capacity and the rate of SLA violation. It predicts variable workloads over intervals shorter than the time scale of demand variability. This work focuses on dynamic consolidation utilizing but it does not perform energy-aware placement on servers.

III. POWER MANAGEMENT AND AUTOMATIC RESOURCE PROVISIONING FRAMEWORK

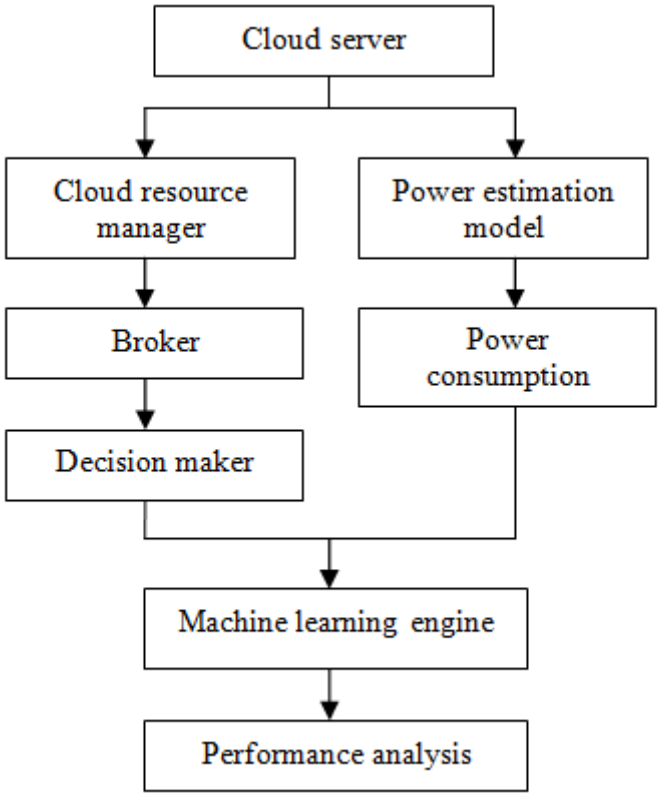

Fig. 1: SYSTEM ARCHITECTURE

The framework of power management and automatic resource provisioning for cloud infrastructures based on machine learning is represented in above Fig. 1. We present a framework for practical system-wide power management and automatic resource provisioning. Firstly, it provides the functionality of performance metrics collection in virtual machine

instances to estimate physical server power consumption. Secondly, based on the preprocessed information about power consumption and performance, the framework can determine to reconfigure the cloud computing system with automatic resource provisioning in order to optimize the trade-off between power efficiency and performance.

In the framework, Power Estimation Model collects resource utilization information from all virtual machines to estimate physical server power consumption. The power consumption information estimated by the power estimation model is transferred to power management model as a feedback for making decisions of auto-configuration. Power management model compiles the performance and power consumption to generate a immediate reward for the current configuration. Based on the reward, machine learning model refines the learning system for making better decision on system configurations.

At the same time, Resource management is performed by two components: a broker that receives requests from User that is an abstraction of the various users of the private cloud system and determines whether the request can be accepted or rejected. The Decision Maker (DM) and the Machine Learning Engine (MLE) are used for predicting the future workload characteristics used to perform auto-scaling of resources. The broker is responsible for handling the user requests. Upon arrival of a request it decides on whether to accept or reject the request by the matchmaking and scheduling component. Additionally, the broker sends the requests to MLE via DM. This allows MLE to update its prediction by learning the behaviour of the new requests. Broker uses the MatchMakeSched module for matchmaking and scheduling that are two important operations performed by the resource management. The module is also used by MLE to simulate the matchmaking and scheduling operations for predicted requests.

DM along with MLE run independently from the rest of the system. After a set time interval that corresponds to the training time duration, they cooperate to predict the characteristic of each of the next k requests Req = {Req1, Req2, …, Reqk} arriving on the system:
Arrival Time (AT): The time at which the request arrives on the system.
Earliest Start Time (EST): The earliest time the request is allowed to begin execution.
Service Time (ST): The time taken by the request to execute.

Deadline (DL): The time by which the request must complete its execution.

DM performs the scaling decisions after receiving a response for its request prediction query to MLE that uses Weka to predict future requests using a machine learning technique based on Linear Regression (LR). Machine learning aided by LR determines the predicted characteristics for a predetermined number of requests. In addition to LR, other machine learning algorithms such as Support Vector Machines (SVM) has also been simulated.

DM determines the number of resources currently available on the system and the requests scheduled on each resource. Next, DM requests MLE to determine the characteristics of the next k predicted requests and the state of the N resources that are currently acquired by DM. Then DM uses the predicted requests and simulates scheduling and matchmaking for the k predicted requests that are expected to arrive on the system next. During the simulation of matchmaking and scheduling performed by DM for the k predicted requests, if the existing resources are not found to be adequate to meet deadlines for the predicted set of requests, the algorithm considers the acquisition of more resources.

IV. RESULT ANALYSIS

The simulation experiments are performed using a PC with an Intel Core i7 CPU, 8 cores (2.8 GHz) with 4 GB of RAM. The system used for the prototype is described next. The prototype broker and other components are run on same the machine described in the previous sentence. The user module generating requests runs on a different PC with an Intel Dual Core, (3.0 GHz) CPU and 4 GB of RAM.

Energy consumption metric is the total energy consumption by the physical resources of a data center caused by the application workloads. Table I illustrates the power consumption characteristics of the selected servers in the simulator. Since the utilization of the CPU may change over time due to the workload variability. Thus, the CPU utilization is a function of time and is represented as U(t). Therefore, the total energy consumption by a physical node (E) can be defined as an integral of the power consumption function over a period of time as shown in Equation (1).

$$E = \int_{t_0}^{t_1} P(U(t))dt \ldots\ldots (1)$$

Fig. 2 shows the proposed dynamic VM consolidation based on described method can bring higher energy saving in comparison to other policies without learning of previous information. By enabling the learning algorithms presented in the power and resource management of cloud with ML method, a significant reduction of the energy consumption in the real workload.

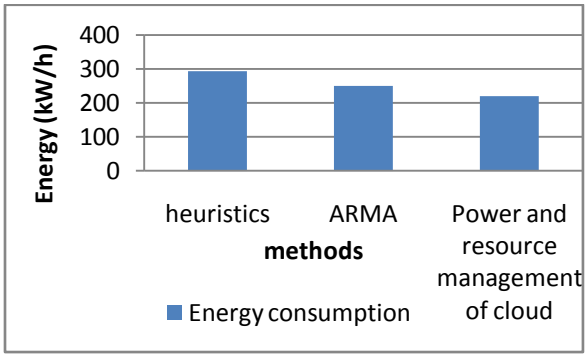

Fig. 2: ENERGY CONSUMPTION

All requests are considered to be arriving from a group of users that belong to the client enterprise, abstracted as the user module. Hence, an increase in arrival rate implies the collective users paying more for the higher values of λ. Fig. 3 presents a comparison of all three systems as heuristics-based method, ARMA (Auto-Regression Moving Average) method and described power and resource management of cloud with ML method. As shown in the figure, power and resource management of cloud leads to a lower total user cost in comparison to the other two methods.

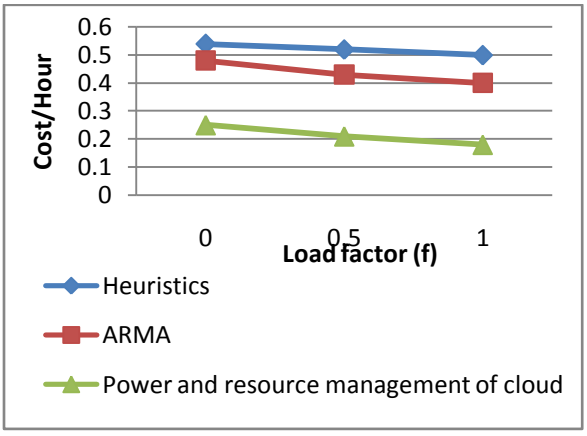

Fig. 3: COMPARISON OF TOTAL USER COST

We select Apache Hadoop MapReduce Wordcount workload as our application to test the system. As a particular project developing for open source software in reliable, scalable and distributed computing environment specializing in processing large data set, Apache Hadoop is an ideal test-bed for cloud computing experiment. Fig. 4 represents the power consumption of three three systems as heuristics-based method, ARMA (Auto-Regression Moving Average) method and described power and resource management of cloud with ML method (considered as CAPM (cloud adaptive power management)).

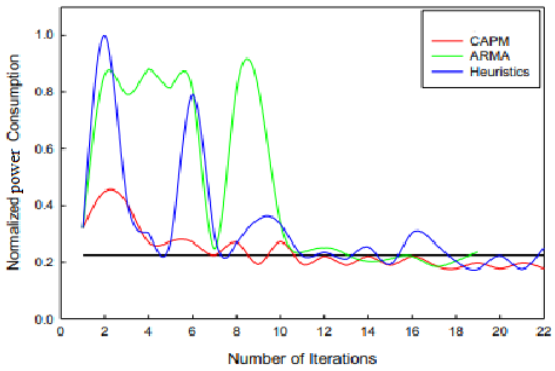

Fig. 4: POWER CONSUMPTION

From Fig. 4 it is observed that, power consumption of described power and resource management of cloud with ML method is very less compared to the heuristics-based method, ARMA (Auto-Regression Moving Average) method.

V. CONCLUSION

This paper describes a new framework for power management and automatic resource provisioning for cloud infrastructures based on machine learning that perform proactive auto-scaling. The proposed broker-based system determines the number of resources to be used in the system by predicting the characteristics of future requests and estimates system instantaneous power consumption. Energy consumption, user cost and power consumption parameters are analyzed in the result analysis. Three methods are compared in this analysis and these are heuristics-based method, ARMA (Auto-Regression Moving Average) method and described power and resource management of cloud with ML method. Therefore less power and energy consumed by described power and resource management of cloud with ML method. Also demonstrated that using the proposed proactive broker can lead to a higher profit as compared to

other non-proactive systems. Performance analysis of the proposed system using real workload traces forms an interesting direction for future research.

VI. REFERENCES

[1] Wonok Kwon, Hagyoung Kim, "Efficient Server Power Supply Configuration for Cloud Computing Data Center", 2014 International Conference on Computational Science and Computational Intelligence, Volume: 2, Year: 2014

[2] Jie Bao, Zhihui Lu, Jie Wu, Shiyong Zhang Yiping Zhong, "Implementing a novel load-aware auto scale scheme for private cloud resource management platform", 2014 IEEE Network Operations and Management Symposium (NOMS), Year: 2014

[3] Nguyen Trung Hieu, Mario Di Francesco, Antti Ylä Jääski, "A virtual machine placement algorithm for balanced resource utilization in cloud data centers", 2014 IEEE 7th International Conference on Cloud Computing, Year: 2014

[4] Wang En Dong, Wu Nan, Li Xu, "QoS-Oriented Monitoring Model of Cloud Computing Resources Availability", 2013 International Conference on Computational and Information Sciences, Year: 2013

[5] Keting Yin, Shan Wang, Gang Wang, Zhengong Cai, Yixi Chen, "Optimizing deployment of VMs in cloud computing environment", Proceedings of 2013 3rd International Conference on Computer Science and Network Technology, Year: 2013

[6] Sadeka, I. et al., "Empirical prediction models for adaptive resource provisioning in the cloud", Future Generation Computer Systems, vol. 28, no. 1, pp 155 – 165, January, 2012

[7] G. Dhiman and T. S. Rosing, " System-level power management using online learning", Proceedings of the Computer-Aided Design of Integrated Circuits and Systems (CADICS), pp. 676–689, 2009.

[8] A. Verma, P. Ahuja, and A. Neogi, "pMapper: power and migration cost aware application placement in virtualized systems", Proceedings of the 9th ACM/IFIP/USENIX International Conference on Middleware, p.p 243–264, 2008.

[9] Wood, T. et al., "Profiling and Modeling Resource Usage of Virtualized Applications" Proceedings of the 9th ACM/IFIP/USENIX International Conference on Middleware, New York, USA pp. 366-387, 2008.

[10] N. Bobroff, A. Kochut, and K. Beaty, "Dynamic placement of virtual machines for managing SLA violations", Proceedings of the 10th IFIP/IEEE Intl. Symp. on Integrated Network Management (IM), pp.119–128, 2007.