Optimal Resource Estimation Policy Selection for Ecommerce Applications in Cloud

Mrs.Poornima Anand Bommannavar
Assistant Professor,
Department of Computer Science & Engineering,
Bangalore, India.
Poornima.dsce@gmail.com

Dr. Krishnan R
Professor,
Department of Computer Science & Engineering,
Bangalore, India.
krishnanr1234@gmail.com

Ms.Shahedha S
M.Tech CNE,
Department of Computer Science & Engineering,
Bangalore, India.
s.shahedha@gmail.com

***Abstract*— Cloud Computing is growing exponentially across organizations in various domains and it has a vast impact on the way software gets developed and tested. The web based applications these days have configuration settings different from deployment requirements. The main focus of Cloud Computing is to deliver reliable, secured, fault-tolerant and elastic infrastructures for hosting an E-Commerce application. Scheduling policies and allocation policies for resources which affect the performance and utilization of cloud infrastructure (i.e. hardware, software services) for various E-Commerce application under varying load and system size is highly challenging problem to deal with. Performance analysis and optimal resource management policies allows cloud service providers to improve their Quality of Service (QOS). This work focuses on the process of selecting the best resource estimation policy for a given workload from various policies such as, Maximum Log-Likelihood, Maximum Product of Spacing Estimator, and Probability Weighted Movements. Detailed experimentation has been carried out for using the Amazon Web Service(AWS) public cloud, mimicking it on the cloudsim simulator. The modeling of workload is quiet challenging due to the unavailability of trace logs for analysis. Hence, in this paper, workload is generated based on application model for E-commerce application considering varying user behavior relating to different user profiles. The amount of resources consumed is closely monitored and a resource usage model has been developed and validated to choose the best estimation policy**.

Keywords— Workload generation, cloud, Performance Distribution Policies, Estimation Policies

I. INTRODUCTION

Software testing using traditional techniques has become expensive in terms of cost, effort and other overheads. Leading IT companies are shifting to cloud for storage, computation, networking and testing as it is cost effective[1]. Testing a web based application these days uses cloud for computing resources such as hardware, network bandwidth , storage which is cost effective and efficient.

A new paradigm which has emerged with an aim to provide a reliable, customized and better Quality Of Service QOS guaranteed dynamic environment for end users using Cloud Computing. The principle behind cloud computing is that the data of end user is not stored locally but instead is stored in the data center with the help of wired network.

The organization that provides the cloud services is called the cloud service provider who is responsible to manage and maintain the data in datacenters. The end user can access their information at any time by using the API facilitated by the cloud provider. The different services provided by cloud provider are

1. Infrastructure as a Service (IAAS): IAAS involves the cloud orchestration technology. The primary goal of this is to manage the creation of virtual machine, and on which physical host it should be executed. The capabilities like provision processing, storage, network and other fundamental resources where the user may deploy the software are given to the customer.
2. Platform as a Service (PAAS): PAAS provides a platform to the customer where they can develop their software or products. PAAS is delivered in three ways as a public cloud to the provider where client controls the software deployment, as a private service within the firewall.
3. Software as a service (SAAS): SAAS is a licensing and delivery model where the software is licensed based on the subscription. It is accessed by end user using client via web browser. It is a common delivery model for many business applications.

There are many advantages in cloud computing one of the most important one is its cost effectiveness this it does by avoiding the capital investment of the organization and by renting the physical infrastructure from the third party provider. Cloud infrastructure scales on demand to support fluctuating workload. Cloud based application and data are accessible virtually.

Cloud computing is very flexible the reason behind this is the quick access and more resource from the cloud is obtained from cloud when it is needed to expand business. The resources can be accessed remotely which enables to access the cloud services from anywhere at any time. Resource allocation is the process of assigning the resources to the application needed through the internet. Resource allocation will suffer with a problem if the allocation is not managed properly. The resource provision will solve the problem by allowing the service provider to manage for each module. Resource allocation policy is all about the proper way of allocating the resources and integrating the cloud providers activities so as to meet the needs of the applications. The important information needed is the tasks/jobs and the amount of resources needed to complete them.

Cloud is used as a platform for vivid applications which expect different Quality of Service aspects such as high performance, reliability and high availability. The service providers find it challenging to keep up these aspects mentioned in the Service Level Agreement between them and the customers as clouds have varying demand, infrastructures of different sizes, and different resource configurations for different user profiles.

978-1-7281-0418-8/19/$31.00©2019 IEEE

II. BACKGROUND

Rodrigo N in his work [1] has proposed an extensible simulation toolkit named Cloudsim which enables modeling for cloud environments. The Cloudsim consist of multilayer design which support for the creation of event and entities like Data center, broker and CIS registry. A cloud provider can also study different scheduling policies in this Cloudsim simulator which is cost effective. Here the datacenters can be created and hosts can be added to it. The host may have one or more data centers but one data center will have only one host. The hosts in the datacenter use round robin scheduling policy to share the resources. Two policies are implemented namely space shared and time shared. It also has a broker agent which will communicate between the entities.

Chen in [4] discusses distributed modeling analysis with the aim of achieving high resource utilization at low cost. The application discussed here is similar to the map reduce application type where the goal to trade-off between the latency and resource utilization. The test environment used in his work is Amazon Web Services. However this paper does not present any verification results for the work.

Moreno [3] describes about the hierarchical and multiple time scale approaches to characterize e-commerce workload. The characterization is done in terms of sessions where every customer executes tasks such as browsing, selecting, adding, and registering. Workload deals with file size, file probability, self-similarity in web traffic. The authors have identified 10 different workload properties that have been inferred from two different website data to understand the workload patterns are derived from analysis of the data. The main challenging thing Moreno inferred was that the number of users have been growing exponentially in past few years and so handling the workload for an E-commerce application has become a major challenge.

Kavulya in [7] proposed a model based on the map reduce application where the workload running in map reduce environment is advantageous to both the user and the cloud service provider. This work has been performed on dataset released by yahoo which is an open source. The hadoop frame work has been utilized in this work and only one single parameter has been estimated here i.e. CPU utilization There is no verification provided for this work as described in the paper that Wang has built a simulator for map reduce workload that can predict the application performance with different configuration up to the high prediction rate. The study also brings out the idea of analyzing the cluster patterns and bringing out the best results.

Archana Ganapathi in [8] has described about her statistical model using which they were e able to predict the resource requirements of the cloud application such as models that will help us to predict the frame work that can guide the system design and deployment decision such as scale, scheduling and capacity. They have used the Google trace logs. The paper is not clear about the which granularity of scheduling decisions in hadoop.

III. PROPOSED SYSTEM

The process involved in creating the model is listed in fig 1. The proposed system involves generating the desired workload using the Apache JMeter workload generation tool. We use the same tool to generate the workload on cloudsim and AWS cloud environments. The user activity and performance modeling is carried out in three steps:
i. Statistical analysis used to analyze data characteristics to determine if some data transformations are needed to define candidate distribution to represent a model.
ii Parameter estimation uses estimation methods for a selected distribution to set the parameters of the model for the collected samples.
iii.GoF test is being used to assess if the distributions and parameters offer a suitable approximation on the observed data.
In this experiment we consider two user profiles namely browsers and buyers. The various parameters considered for experimentation includes the number of user requests generated by JMeter, the CPU utilization for the processing the request, memory and disk utilization for the request and the response time of the server to process the request. This is listed in table 1.1

Table I.: Experimentation Assumptions

Experimentation Assumptions	
User Profiles	Buyers
	Browsers
Parameters Considered	User Requests
	CPU Utilization
	Memory Utilization
	Disk Utilization
	Response Time

Fig1: Modules

The various modules implemented are
a) Workload generation – here we generate the workload
b) Resource Utilization profile
c) Validation.
d) Best Fit policy.

3.1 Workload Generation

The workload for an application is defined as the amount of work performed by an entity over a given period of time. Workload marks as a benchmark in the process of evaluating the performance of an application. Study says that workload modeling in cloud has not received much attention[29]. In our work we have used a performance model of 3 tier architecture which has functionalities such as browsing and buying. Fig 2 shows the workload generation module that is being implemented.

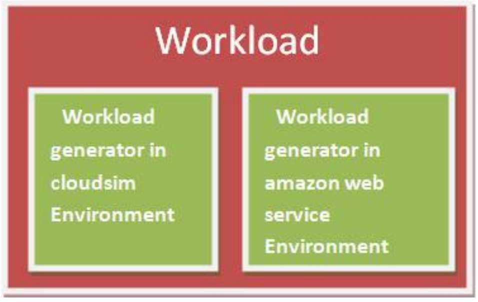

Fig2: ModuleI: Workload Generation

Workload in our implementation is the number of user requests generated by JMeter. We categorize the workload based on the type of user profiles i.e buyers and browsers. We have used apache Jmeter workload generator tool to generate the workload in the AWS cloud. We have plugged in apache JMeter into cloudsim to generate the desired workload. The configuration of the Apache Jmeter is illustrated in table II

Table II: Apache Jmeter Configuration

Configuration	Value
Cloudlets	1000
User Profile	Browsing/Buying

3.2 Resource usage for different user profile:

Each request uses certain system resources to process the request. These resources are CPU Utilization, Memory Utilization, Disk Utilization. We also capture certain other parameters such as the number of user request processed by the server and the server's response time which is the time taken to process the user request and reply back with the result. These parameters are listed in Table I which are called as Resource Usage. We capture the resource usage by different user type of users which is later validated. The validation process is explained in Section 3.3. The figure 3 depicts the various the parameters of Resource Usage captured for different user profiles namely Buyers and Browsers.

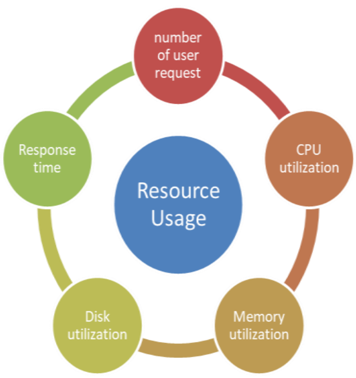

Fig3: Module II: Parameters for Resource Usage

3.3 Validation:

The resource usage captured for different types of user profiles namely Browsers and Buyers, from the AWS cloud is called as Actual data. The resource usage captured from cloudsim is called as simulated data. In the Validation module the user requests are separated based on different types of user profiles i.e. Buyers and Browsers. Then we compare the results of actual cloud and simulated cloud using the Wilcoxon signed rank test which is a non-statistical hypothesis test. The large number of comparisons is carried out and the relative error rate is calculated. The validation process is described in Fig 4.

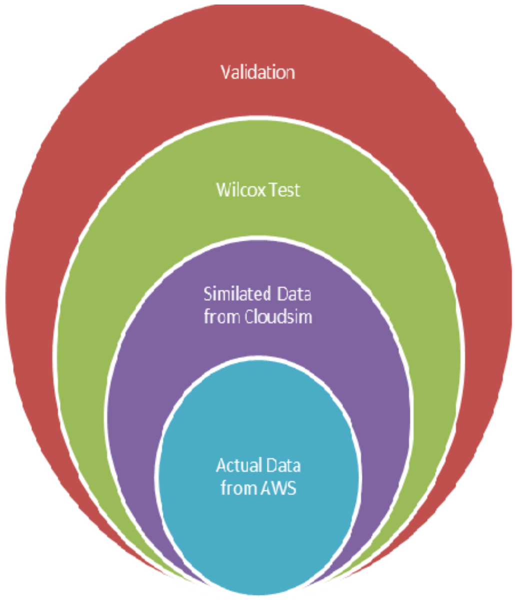

Fig 4: Module III: Workload Generation

3.4 Best Policy for each Parameter:

After performing the Wilcoxon signed rank test we choose the best estimation policy along with the various distributions. Fig 5 shows the various distributions techniques used in this work. The various distributions considered are Gauss-Laplace[GL] and Maximum likelihood Policy[MLLP], Gauss-Laplace[GL] and maximum product of spacing policy[MPSP], Gauss-Laplace[GL] and product of weighted moments Policy[PWMP], Gauss-Laplace[GL] and Quantile matching Policy[QMP], Gauss-Laplace[GL] and Histogram matching Policy[HMP], GEV and Maximum likelihood Policy[MLLP], GEV and maximum product of spacing policy[MPSP], GEV and product of weighted moments Policy[PWMP], GEV and Quantile matching Policy[QMP], GEV and Histogram matching Policy[HMP].

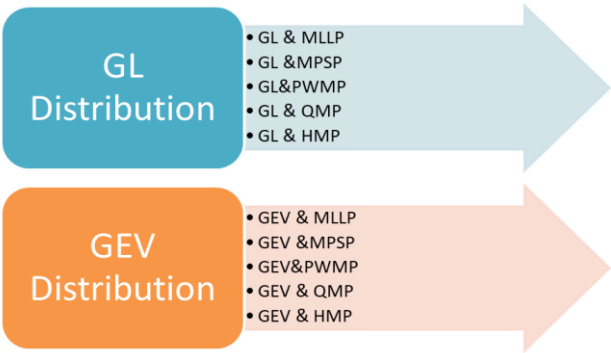

Fig5: Module IV: Policy Estimation

IV. Implementation and Validation

Fig 6 shows the various modules to be implemented for workload modeling and resource utilization.

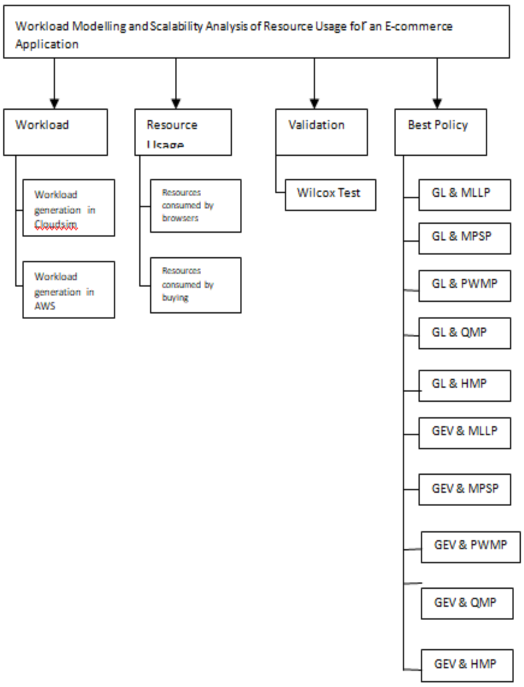

Fig6: Implementation of the modules

The implementation begins by accepting the number of datacenter, broker, host and the number of cloudlets i.e. user requests for both cloudsim and AWS cloud environments. We use the Apache jmeter which is plugged into cloudsim to generate the desired workload. The same tool is integrated with AWS cloud to generate the workload. We then capture the parameters listed in table I namely the number of user requests, response time, CPU utilization, Memory Utilization, Disk Utilization. To evaluate the accuracy of the model, we compare the simulated data from cloudsim and observed data from AWS cloud. We then use the WilcoxMann–Whitney(WMW) hypothesis test [28] to evaluate the hypothesis that the simulated data belongs to the population that follows the probability distribution of the observed data. If p-Value >α ,the hypothesis cannot be rejected. Otherwise, the null hypothesis is rejected. The level of significance was set at α=0.05 for all the tests. This is one of the non-parametric hypothesis test which is used to compare the samples of related examples, it gives out the matched or unmatched measurement based on a single sample. Wilcox test even tells us that the samples collected resemble to the same distribution or not. There are few assumption made in the wilcoxon test so that each pair is randomly selected to test.

1. The primary step of the Wilcox test is to calculate the difference in reoccurred measurements and also to measure the accurate difference.
2. The next step of the test is to rearrange the values in the ascending order with the accuracy of difference measured.
3. The third step in the Wilcox test is to rank the differences measured and to ignore the cases where the difference is measured as zero.
4. The fourth step of the Wilcox test is to sign each difference measured. If the measured difference is less than zero then it is multiplied by -1, and if the difference is more than zero than it is multiplied by +1 and remains positive.

Best Policy for each parameter

Here we consider two types of distributions namely Generalized Lambda and Generalized Extreme Value distribution to choose best resource estimation policy from among the maximum likely hood policy, Probability weighted moments policy, Maximum Product of spacing.

V. Result and Analysis

5.1 Workload Generator

We have used Apache JMeter workload generator for generating the workload. We accept the number of datacenters, number of brokers, number of virtual machines needed to complete the task, number of user requests and the number of users as input. We setup a server is in AWS cloud and record the metrics mentioned in table 1.1. The same workload is generated in clousim environment using the same Apache Jmeter workload generator tool on a virtual server. The captured metrics of AWS and Cloudsim environment are then compared. Fig 7 and Fig 8 show the workload generation for cloudsim and AWS cloud.

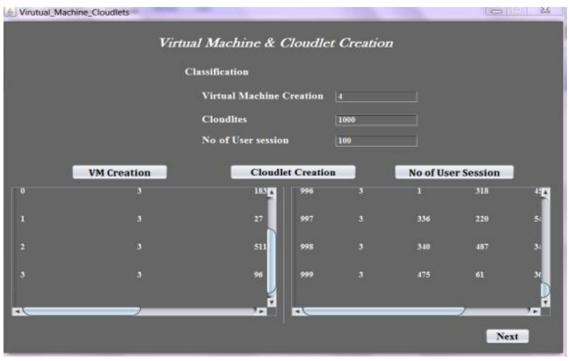

Fig7:: Workload Generation : Input Values

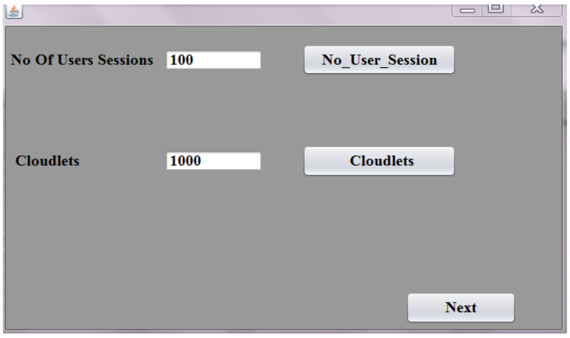

Fig8: ModuleI: Workload Generation for cloudsim and AWS

5.2 Collection of Resource Usage Profile

The metrics from cloudsim environment and AWS cloud that are captured is shown in fig 9 and Fig 10. The various metrics captured are Number of instructions executed by a server, the CPU utilization, the Memory utilization, Disk utilization and the Response time of the server.

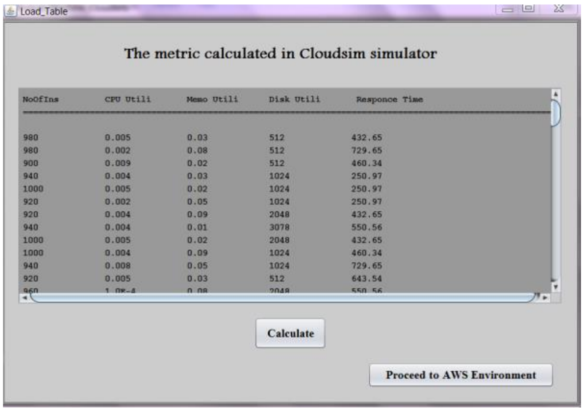

Fig9: Parameter captured from Cloudsim

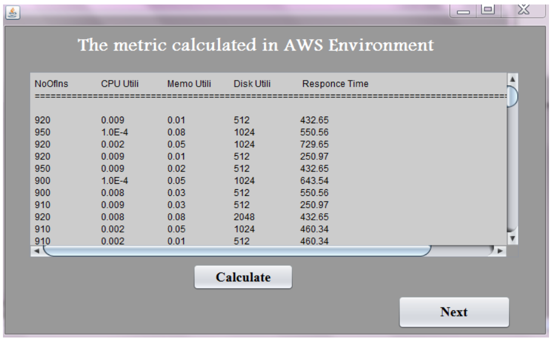

Fig10: Parameters captured from AWS

5.3 Validation

The validation process starts by performing the null hypotheses test using Wilcoxon signed rank test for normal distribution on the captured data from AWS and cloudsim simulator. We calculate the p-value of each parameter. The value of H_0 is set to 0.05. Based on this if the p-value is less than 0.05 we reject the hypothesis. We then calculate the error rate as the number of rejection from the number of p-values obtained. The below snap shot of Fig 11 and Fig 12 depicts the p-value for the number of instruction for different user profiles. The error rate for other parameters is also obtained in a similar way.

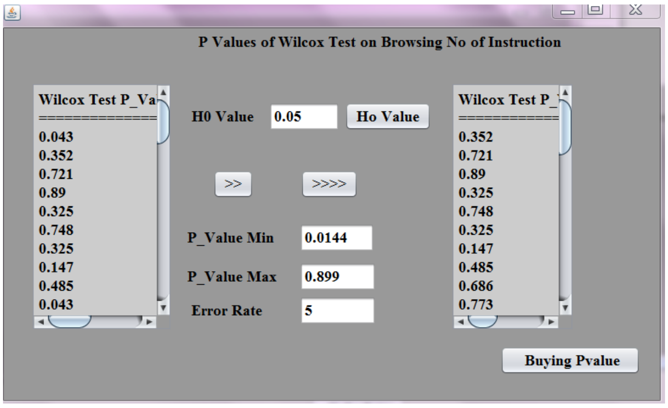

Fig11: Validation: Browser User Type

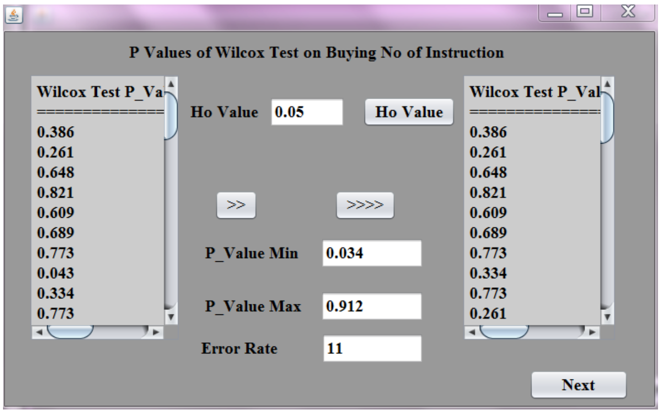

Fig12: Validation: Buyer User Type

5.4 Best Policy

Here to determine the best resource estimation policy we consider the resource estimation for different user types and obtain the location, skewness, kurtosis. Based on these parameters the best policy is determined such as maximum of location, minimum of skewness and maximum of kurtosis and the best policy is displayed in the jframe and the related graphs are obtained. Fig 13 depicts the same.

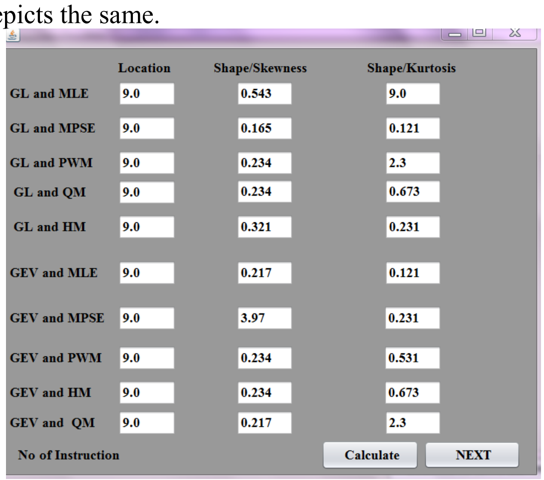

Fig13: Policy Estimation

The following Fig14 and Fig 15 depicts the comparison of the best policy chosen for simulated data using cloudsim and actual data using AWS cloud. It can be observed that for browsing user profile the cloudsim chooses GL & Maximum Product of Spacing Estimator where as AWS cloud chooses GEV & Maximum Product of Spacing Estimator and AWS has performed better in selecting this policy. The other policies are the same for cloudsim and AWS cloud.

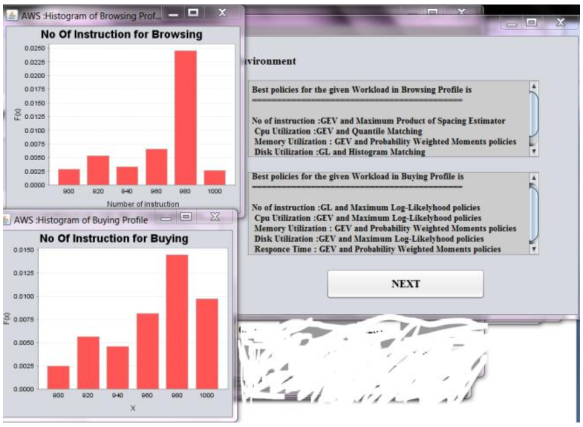

Fig14: Cloudsim Vs AWS Cloud

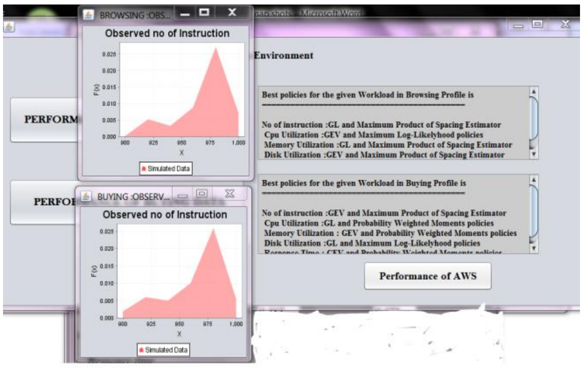

Fig1: Cloudsim Vs AWS Cloud considering the number of Instructions

VI. CONCLUSION AND FUTURE WORK

VII.

In this work, we were able to generate a desired workload for different user profiles of a web application using an efficient workload generator to select the policy for optimal resource utilization. We captured certain important parameters from both the entities and performed Wilcox test to capture the similarity between the cloudsim and AWS cloud. Using different parameters such as location, skewness, kurtosis of p-value are captured and based on these values we determine the best resource estimation policy for both the environments. The important observations that can be taken from the work i) using this work we can say that the type of user profile plays a vital role in estimating resource utilization for varying user profiles where in use various statistical distributions to represent various metrics such as the number of instructions, CPU utilization, Memory utilization. ii) we have used GEV distribution to represent the instruction arrival instead of the standard Poisson methods used earlier and we have used GEV and GL methods to represent session time for users instead of Exponential distributions. iii) we have compared the results of both cloudsim simulator and AWS cloud service and we were able to mimic the AWS cloud on cloudsim simulator with good accuracy rate, and hence can be used by researchers.

REFERENCES

[1] Calheiros Rondrigo N, Ranjan R, Beloglazov A, De Rose CAF, Buyya R. Cloudsim: a toolkit for modeling and simulation of cloud computing environments and evaluation of re- source provisioning algorithms. Softw: Pract Exp 2011;41:2350.

[2] FeitelsonDG. Workload modeling for computer systems performance evaluation. Cambridge University Press;2015.

[3] Moreno I, Garraghan P, Townend P, Xu J. An approach for characterizing workloads in Google cloud to derive realistic resource utilization models. In: Proceedings of 7th internationalsymposiumonserviceorientedsystemengineering(SOSE).IEEE;2013. p. 4960.

[4] Chen Y, Ganapathi AS, Griffith R, Katz RH. Towards understanding cloud perfor-mancetradeoffsusingstatisticalworkloadanalysisandreplay.TechnicalReport.Uni- versity of California at Berkeley;2010.

[5] Mulia WD, Sehgal N, Sohoni S, Acken JM, Stanberry CL, Fritz DJ. Cloud workload characterization. IETE Tech Rev2013;30:38297.

[6] Reiss C, Wilkes J, Hellerstein JL. Google cluster-usage traces: format + schema. Technical Report. Google Inc.; 2011. URL: http://code.google.com/p/ googleclus- terdata/wiki/TraceVersion2.

[7] Kavulya S, Tan J, Gandhi R, Narasimhan P. An analysis of traces from a production MapReduce cluster. In: Proceedings of 10th IEEE/ACM international conference on cluster, cloud and grid computing (CCGrid); 2010. p. 94103.

[8] Archana Ganapathi, Yanpei Chen, Armando Fox, Randy Katz, David Patterson. Statistics-Driven Workload Modeling for the Cloud

[9] Automated Workload Generation for Testing Elastic Web Applications Michel Albon- ico, Jean-Marie Mottu, GersonSunye

[10] Real workload characterization and synthetic workload generation. KeerthiBangari1, Chittipothula C YRao

[11] TarunGoyal a AjitSingh b AakankshaAgrawal a Cloudsim: simulator for cloud com- puting infrastructure andmodeling

[12] Calheiros RN, Ranjan R, Beloglazov A, De Rose CAF, Buyya R. Cloudsim: a toolkit for modeling and simulation of cloud computing environments and evaluation of re- sourcevprovisioning algorithms. Softw: Pract Exp2011;41:2350.

[13] Garraghan P, Townend P, Xu J. An approach for characterizing workloads to derive realistic resource utilization models. In Proceedings of 5th international symposium on service oriented system engineering (SOSE). IEEE; 2013. p. 4960.

[14] Policy for Resource Allocation in Cloud Computing,Amitha B., Shreenath Acharya, Department of Computer Science, St. Joseph Engineering College, Mangalore,India

[15] Nagesh Hawanna, Kushang Parikh, Haleemap. k, Jayasubalakshmi R and N.Ch.S. Nlyengar, Virtual Machine Allocated Policy in Cloud Computing Using Cloudsim in Java, Vol.8, No.1, pp.145-158, ISSN: 2005-4262,2015.

[16] S. ThamaraiSelvi, C. Valliyammai and V. Neelaya Dhatchayani, Resource Allocation IssuesandChallengesinCloudComputing,IEEEInternationalConferenceonRecent Trends in Information Technology, ISBN: 978-1-4799-4989-2/14, 2014,Chennai

[17] Narander Kumar, Swati Saxena, A Preference-based Resource Allocation In Cloud Computing Systems, 3rd International Conference on Recent Trends in Computing, 104 111, Lucknow, 2015, doi:10.1016/j.procs.2015.07.375.

[18] Tchana A, Dillenseger B, De Palma N, Etchevers X, Vincent J, Salmi N, et al. A self- scalable and auto-regulated request injection benchmarking tool for automatic satura- tion detection. IEEE Trans Cloud Comput 2014.doi:10.1109/TCC.2014.2321169.

[19] Openstack cloud software. 2013. URL:http://www.openstack.org/.

[20] Paxson V. Bro: a system for detecting network intruders in real-time. Comput Netw: Int J Comput Telecommun 1999;31:243563.

[21] Hashemian R, Krishnamurthy D, Arlitt M. Web workload generation challenges an empirical investigation. Softw: Pract Exp 2012;42:62947.

[22] Tcpdump/libpcap public repository. 2015. URL:http://www.tcpdump.org/.

[23] A. Weiss, Computing in the clouds, netWorker, vol. 11 no 4,pp. 16-25, Dec. 2007. International Journal on Future Revolution in Computer Science & Communication Engineering ISSN: 2454-4248 Volume: 4 Issue: 3 410414

[24]ConfidenceInCloudGrows,PavingWayForNewLevelsOfBusinessEfficiency,Fujitsu, 2010.

[25] Junzan Zhou, Shanping Li, Zhen Zhang, Zhen Ye(2013) [Migrating performance test- ing to cloud: Challenges and issues a survey paper]

[26]V.Priyadharshini,Dr.A.Malathi[Surveyonsoftwaretestingtechniquesincloudcom- puting2013]

[27]Anirvan Chakraborty and Probal Chaudhuri . A

[28]Wilcoxon-mann-whitney type test for infinite dimensional data

[29]Shi Hengliang1 , Zhao Changwei1 , Huang Tao1 and Dong Yongsheng[RESEARCH ON DISTRIBUTED SOFTWARE TESTING PLATFORM BASED ON CLOUDRE- SOURCE]

[30] GanapathiA,ChenY,FoxA,KatzR,PattersonD.Statisticsdrivenworkloadmodeling for the cloud. In: Proceeding so international conference on the data engineering workshops. IEEE; 2010. p. 87 92.