An Architectural Framework for Workload Demand Prediction in Scalable Federated Clouds

A. Stanislas
Ph. D. Scholar, Department of Computer Science
St. Joseph's College (Autonomous)
Tiruchirappalli, Tamil Nadu

L. Arockiam
Associate Professor, Department of Computer Science
St. Joseph's College (Autonomous),
Tiruchirappalli, Tamil Nadu

ABSTRACT
Cloud computing is a wonderful paradigm which assures the customers of providing computing resources instantly whenever they are in need. It is the virtualization technology that makes this paradigm a reality. But the present technology which is used for provisioning virtual machines is not adequate. Thus, there is latency in service provisioning and the long waiting time of virtual machine provisioning hampers the future popularity of cloud computing. So, high scalability which is the key factor of cloud computing is not easily possible. Therefore, there is a need for a mechanism to enable the service provisioning effectively with high scalability. In view of that, this paper presents a system which predicts the workload demands of the service requests automatically so as to prepare the virtual machines in advance in order to ensure the customers with instant services efficiently without much delay. Trend value analysis using various methods is carried out in the prediction system.

Keywords
Scalability, Workload, Virtualization Technology, AUTOPRED, Prediction System

1. INTRODUCTION
Cloud computing is an emerging computing paradigm where users can request for computing resources on-demand through networks and cloud computing platforms anytime and anywhere. Some of the distinguishing characteristics of cloud computing are elasticity, scalability, hardware virtualization, and fast service configuration, etc. In cloud computing environments, three major kinds of services can be provided, including Infrastructure as a Service (IaaS), Platform as a Service (PaaS) and Software as a Service (SaaS). These cloud services can be composed into a value-added service to satisfy the dynamic needs of the cloud users. Due to heterogeneous, autonomous and dynamic characteristics of cloud services, various uncertainties can affect the correctness, availability and reliability of service composition in cloud computing environments [1]. Cloud computing offers its customers an economical and convenient pay-as-you-go service model, known also as usage-based pricing. Cloud customers pay only for the actual use of computing resources, storage and bandwidth, according to their changing needs, utilizing the cloud's scalable and elastic computational capabilities [2] [3].

The evolution of IT software services towards cloud computing took an advanced step for the efficient use of hardware resources through the virtualization technology. In traditional approach of hosting services, the user receives certain amount of hardware resources that the customer makes use of. Whereas on the other hand, the cloud approach is to offer on-demand virtualized resources to its users. There is a possibility of dynamic scaling because; the virtual resources can be added or removed at any time during the lifetime of the application hosted on a cloud. Also, dynamic scaling can be easily automated both sides; either at cloud provider level or cloud client level through the use of the cloud provider's APIs. A cloud provider needs to make accurate decisions on when to scale up and to scale down in order to take full advantage of the benefits of dynamic scaling,. If the provider wants to achieve good performance, then he or she should be able to make an accurate scaling decision. In view of this, a new auto-scaling strategy can be thought of. This strategy can be used to identify the usage patterns that have occurred in the past and having a high similarity to the present usage pattern, a decision can be made as to the necessity of scaling for the future requirements [4]. This paper presents a new approach called AUTOPRED to the resource usage prediction problem based on identifying past occurrences' patterns that is similar to the present use of the system. An algorithm for identifying the patterns by using an approximate matching approach is presented.

The rest of the paper is organized in the following manner. Different works related to the proposed system by a number of authors are presented in section 2 and section 3 presents the proposed system architecture. Section 4 presents the various approaches for the workload demand prediction with the Logical Flow Diagram and the algorithm. Finally, section 5 concludes with the suggestions indicating the research areas for the future work.

2. BACKGROUND SCENARIO
Since, the present system for provisioning computing resources could not be instantly ready in cloud; the IT enterprise customers preferred traditional datacenters though it involved huge costs. To respond to these concerns of instant resource provisioning and minimizing the cost, Yexi Jiang et al. [5] proposed a system called A Self-Adaptive Prediction System (ASAP) to predict the cloud virtual machine demands and to prepare the virtual machines in advance in order provision the resources instantly. Their proposed ASAP system consists of three modules; 1) raw data filtering and aggregating module, which does the preprocessing works for the follow-up modules, 2) model generation module which is responsible for building the models for separate predictors, and demand prediction module that reconstructs the prediction models according to the information stored in the model file. Based on the cloud demands prediction, they also proposed an asymmetric and heterogeneous cost measurement system called Cloud Prediction Cost (CPC) to guide the model selection and training.

There is constant increase in energy consumption at data

centers due to large amount of data stored in cloud. This becomes a key issue to be addressed as energy is an important resource in the field of IT industries. Energy can be conserved by migrating the virtual machines (VMs) running on underutilized machines to other machines and hibernating such underutilized machines. Avinash Mehta et al. [6] designed a strategy for energy-efficient cloud data center which makes use of historical data for service request prediction model. It enables the identification of the number of active servers required at a given moment thus making possible the hibernation of underutilized servers. To achieve this goal a slight modification is made in the existing cloud infrastructure where there are four controllers; cloud controller which manages cloud applications through their entire life cycles, cluster controller that controls the execution of VMs running on the nodes, storage controller which is used to provide central storage in a cluster and node controller that controls VM activities, including the execution, inspection and termination of VM instances. In their service prediction model they have used the kNN algorithm with a slight modification to obtain an adaptive kNN algorithm which predicts the (n+1)th value based on some given n past values.

High scalability, flexibility, and cost effectiveness are the key benefits of cloud computing. For efficient provisioning of computing resources, the administrators need the abilities to characterize and to predict the workload on the virtual machines. Arijhit Khan et al. [7] developed a co-clustering technique to identify groups of virtual machines that frequently exhibit correlated workload patterns and also the time periods in which the virtual machines are active. They also introduced a method based on Hidden Markov Modeling (HMM) to characterize and to predict the variations of workload patterns. They apply a multiple time series approach to analyze the workload at the group level rather than at the individual virtual machine level. Their approach was to identify only the common workload patterns across multiple VMs of CPU utilizations in different levels and not of various other utilizations.

A key challenge to be addressed to materialize the advantages promised by cloud computing is the design of effective auto-scaling and self-tuning mechanisms capable of ensuring pre-determined QoS levels at minimum cost in face of changing workload conditions. To address this issue, Bruno Ciciani et al. [8] present a design underlying the development of Cloud-TM's Workload Analyzer (WA), a crucial component placed between the Workload and Performance Monitor (WPM) and the Adaptation Manager (AM) of the Cloud-TM platform that has key functions of data aggregating and filtering, workload and resource demand characterization, workload and resource demand prediction, and QoS monitoring and alert notification.

Efficient resource scaling is mandate for the cost-saving benefit in cloud phenomenon. Cloud computing uses virtual resources that have setup time which is not negligible. So, prediction of the workload is a necessary concern for the dynamic scaling. Eddy Caron et al. [9] proposed an approach to overcome this problem. Their approach was based on identifying similar past occurrences of the current short-term workload history. They have used Knuth-Morris-Pratt (KMP) String Matching algorithm for finding solution to this problem. They presented in detail a resource auto-scaling algorithm for their approach.

Cloud customers are in dilemma in choosing the right cloud that performs better for their application. Efficient resource provisioning is one of the concerns that needed to be taken into consideration towards application performance. Ang Li et al. [10] proposed a trace-and-replay tool called CloudProphet which traces the workload of the applications running locally and replays the same workload in the cloud for prediction. Depending on the accuracy, the prediction result can directly point the customer to the best performance provider. They used two common approaches for performance prediction; standard benchmark that can provide a baseline to compare the performance of different providers and another widely used approach called modeling that predicts the performance of simple computation. In this approach, the prediction was based on the current workload and the past occurrences were not taken into account and thus the performance prediction may not be accurate.

Quality of Service (QoS) is an important factor for any non-IT customers to use the services in cloud. The current infrastructures in cloud environment only support resource-level metrics –e.g. CPU share and memory allocation. There is not a well-defined mechanism to translate from service-level metrics to resource-level metrics. Moreover, the lack of precise information regarding the requirements of the services leads to an inefficient resource allocation. So, providers allocate the whole resources to prevent SLA violations. Gemma Reiget al. [11] proposed a novel mechanism to overcome this translation problem using an online prediction system which includes a fast Analytical Predictor (AP) and an Adaptive Machine Learning Based Predictor. They have also shown how a deadline scheduler could use these predictions to help providers to make the most of their resources. The solution of their approach helps dynamically allocating resources to the applications depending on their needs, while guaranteeing that each of them has always enough resources to meet the agreed service level metrics.

There is an increase in demand for cloud computing resources which causes the increase in power consumption. So, there needs to be a system for optimization of power consumption. John et al. [12] have introduced a framework for load demand prediction which would lead to optimal cloud resource allocation by minimizing energy consumption. They use neural network and autoregressive linear prediction algorithm to forecast loads in cloud data center applications. In the neural network model, they use the back-propagation algorithm which consists of a forward pass and a backward pass. The goal of the linear prediction filter is to observe the last N samples of the cloud channel and to update the predictor with reflecting the most recent statistical samples.

Since cloud computing centers are designed to be scalable and to process varieties of software applications, the power consumption is very high to meet the demands. So, there is a need of an Ultra-Low Power Computing System. Kranthimanoj Nagothu et al. [13] introduced a novel concept which involves using a variety of heterogeneous processors, each with different power and performance capabilities. By predicting the load and jointly allocating the tasks to the processors, they have reduced the power consumption. Though there are a number of algorithms available for prediction, the issue that arises often is stability. So, they use Wiener Filter equation for an Auto Regressive Moving Average (ARMA) and also Burg's algorithm which involves backward and forward prediction errors. To ensure stability, they constrain the reflection coefficients to be less than one.

The vast and widespread use of computing resources in the heterogeneous cloud computing environment necessitates the introduction of a management mechanism to ease the administration complexity and optimize the resource utilization. To this end, an appropriate model for the management of computational system resource is proposed by Eleni Patouni et al. [14] which is enhanced with prediction schemes. An

algorithmic framework is introduced for the proactive load balancing of user decision-making requests and an analytical model has been proposed to compute the predicted values of the user satisfaction.

Cloud computing paradigm has brought a relief for the IT resource users from the burden of worrying about the infrastructure and system administration details. The significant problem that exists even now is with regard to the efficient provisioning and delivery of applications using cloud-based IT resources. To improve the efficiency of the system, Rodrogo N. Calheiros et al. [15] proposed a mechanism called analytical performance, and workload information to supply intelligent input to virtual machine generator about the system requirements to an application provider with the limited information about the physical infrastructure. The proposed provisioning technique detects the changes in workload intensity (arrival pattern, resource demands) that occurs over time and allocates multiple virtualized IT resources accordingly to achieve application QoS targets. Their analytical performance model allows the system to predict the effects of provisioning schedule on target QoS.

3. PROPOSED SYSTEM

Since the cloud computing paradigm has brought a relief for the IT resource users from the burden of worrying about the infrastructure and system administration details, there is an increase in demand for cloud computing resources which causes difficulties in scaling according to the demand in cloud environment. The significant problem that exists even now is with regard to the efficient provisioning and delivery of cloud-based IT resources. For efficient provisioning of computing resources, the administrators need the abilities to characterize and to predict the service requests in order to prepare the virtual machines in advance. To achieve good performance, a new auto-scaling strategy can be elaborated namely, by identifying usage patterns that have occurred in the past and have a high similarity to the present usage pattern, a decision can be made as to the necessity of scaling for the future requirements. Thus, a new approach called AUTOPRED is proposed in this paper to predict the service requests based on identifying past occurrences' patterns that is similar to the present use of the system. The architecture of AUTOPRED is presented in Fig. 1.

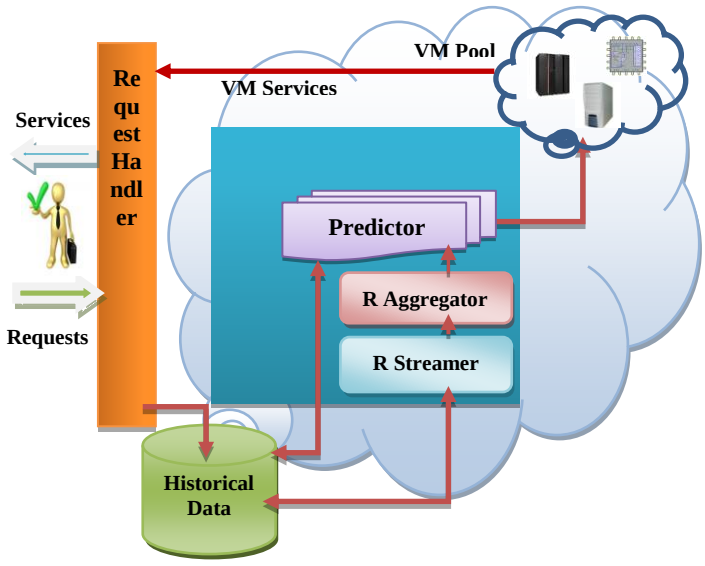

Fig.1: AUTOPRED Architecture

As shown in Fig. 1, all the requests stored in the repository of historical data are separated according to the type of the service requests in correspondence with their time of requests made. The separated requests are once again aggregated together to predict the total number of service requests and as well as the number of each type of the service requests. This information is sent to the virtual machine pool in order that the required services are made available in advance in the form of virtual machines so as to allocate them to the requested customers.

4. WORKLOAD DEMAND PREDICTION

The workload demand prediction service has several purposes: i) to decide on workload's demand pattern; ii) to recognize whether a workload's demands change significantly over time; iii) to support the generation of synthetic demand traces that represent future demands for each workload, e. g., demands for a particular period of time in a day or several weeks or months into the future, to support capacity planning exercises; and, iv) to provide a convenient model that can be used to support forecasting exercises. This section describes the techniques used to implement this service

4.1 Time Series Trend Analysis

A method for finding the trend value is presented for deducing patterns, assessing their quality, and classifying them with regard to quality. A new approach is presented that assesses the similarity among occurrences of a pattern.

A set of ordered observations of quantitative variables taken at successive points in time, in terms of hours, days, months and years is considered for the historical data. The historical data is generated empirically to find the trend value in time series analysis as the nature of the data is concerned with the time for different intervals.

It can be mathematically defined by the functional relationship $y_t = f(t)$, where y_t is the value of the phenomenon under considerations at time t. If the values of a phenomenon or variables at times $t_1, t_2, t_3, \ldots, t_n$ are $y_1, y_2, y_3, \ldots, y_n$ respectively, then the series is

$t : t_1, t_2, t_3, \ldots, t_n$

$y_t : t_1, y_2, y_3, \ldots, y_n$

constitute a time series. Thus, a time series invariably gives a bivariate distribution, one of the two variables being time (t) and the other being the value (yt) of the phenomenon at different points of time. The values of t may be given yearly, monthly, daily or even hourly, usually but not always at equal intervals of time. For example, (i) the number of requests (yt) made per day at the interval of 0-4, 4-8, 8-12, 12-4, 4-8 and 8-12 hours (t), (ii) the number of requests (yt) made during the days of week (t), (iii) the number of requests (yt) made for every month of a year (t) and (iv) the number of requests (yt) made for ten years (t) can be taken as the historical data from any environment for sample to find the trend value.

The empirical method is used to generate the historical data for different intervals of time for five years. The maximum number of service requests for five years is assumed as less than 100000 which, implies that the total number of request per year is less than 20000, the total number of requests per month is less than 1800, and the total number of requests per day is less than 60. The sample observed data generated by the empirical method for six intervals of time per day is given in the following Table 1and Fig.2.

Table 1: Observed Data for a Day

Time	0 – 4	4 – 8	8 – 12	12 – 4	4 – 8	8 – 12
Probability	0.10	0.11	0.12	0.13	0.14	0.15
Cumulative	0.10	0.21	0.33	0.46	0.60	0.75
Y	4	7	9	9	10	8

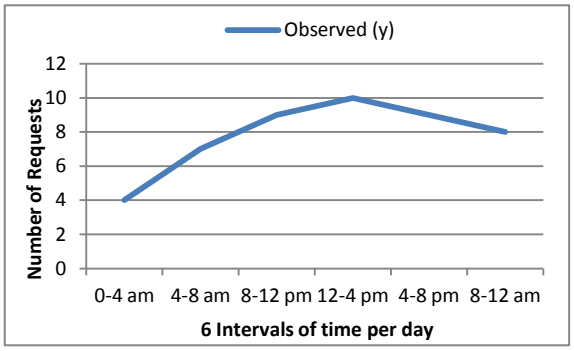

Fig.2: Observed Requests curve per day

As shown in the Table.1 and Fig.2, similar observed requests are generated for a week, month, and year and for five consecutive years respectively. Using these observed data, the value of y', that is the predicted trend value for all the intervals of time, namely, for a day, month, year and for five years are derived by the curve fitting methods in time series analysis.

The four popular methods namely, straight line curve fitting, exponential curve fitting, geometrical curve fitting and hyperbola curve fitting are considered for the trend analysis. The curve which is closest to the observed request curve is considered to be the predicted trend value. To find the précised closest curve, the sum of difference between the observed data (y) and (y') is calculated using $\sum(y - y')^2$ for all the four methods and the minimum among the four is said to be the nearest curve for the observed data curve and that will be the predicted trend value. The following table and figure give the experimental result. It is observed and is evident from the Table 2 and Fig.3 that the geometrical curve fitting method is the nearest curve to the observed data curve and thus, the trend value by geometrical curve fitting method is the accurate prediction model for this nature of workload demand prediction. Therefore, trend value ($Y = a.x^b$) by geometrical curve fitting method is used in the prediction algorithm.

Table 2: Trend Value Analysis

X		1	2	3	4	5	6	$\sum (Y-Y')^2$
Y		4	7	9	9	10	8	
Y'	Str	5.76	10.69	15.62	20.56	25.49	30.42	937.254
	Exp	5.44	6.19	7.05	8.02	9.13	10.40	13.982
	Geo	4.65	6.30	7.54	8.55	9.44	10.22	**8.509**
	Hyp	5.18	5.84	6.71	7.87	9.52	12.04	25.843

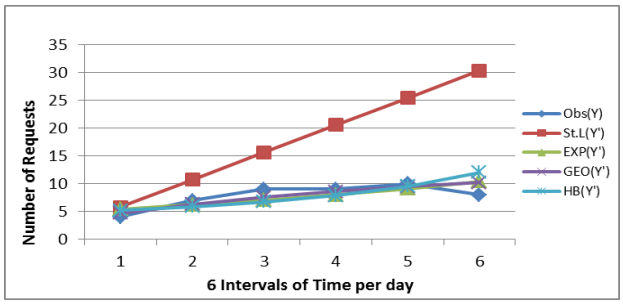

Fig. 3: Trend Value Analysis

4.2 Logical Flow Diagram of AUTOPRED

The exact prediction of the customer's request enhances scalability in federated clouds. The information is sent also to the repository of the historical data to update the service requests for the future prediction. The workload demand prediction through various analyses is carried out. The entire process of working of the AUTOPRED prediction system is given in the Logical Flow Diagram (LFD) with the pseudo code for the algorithm in the following Fig. 4.

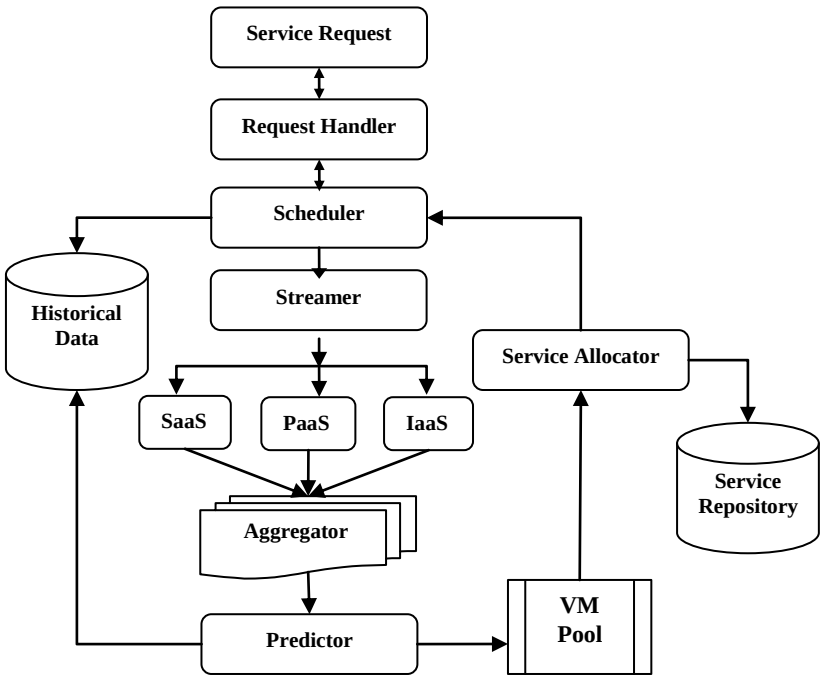

Fig. 4: Logical Flow Diagram of AUTOPRED

As shown in the Fig. 4, the request handler schedules the requests after storing them in the repository of the historical data. The requests are separated based on the types of the service the customers require and aggregate them together to predict the total number of requests of each type and the overall total number of requests. This information is sent to the VM pool for the optimal allocation of the resources. An algorithm is developed and it is simulated using java programming. The pseudo code for the algorithm is given in the following table 3.

Table 3: Pseudo Code for AUTOPRED Algorithm

// Declaration
SR ← Service Request
RH ← Request Handler
SC ← Scheduler
HD ← Historical Data
SA ← Service Allocator
STR ← Streamer
ATR ← Aggregator
PTR ← Predictor
VMP ← Virtual Machine Pool
SRY ← Service Repository

// Pseudo code

1. Start
2. SR submitted in RH
3. Requests are deposited in HD and Scheduled to STR
4. STR streams the requests based on the type
5. ATR aggregates the streamed requests
6. PTR predicts the demand using the trend value $Y = a.x^b$ and sends to VMP & HD
7. SA allocates from VMP & stores information in SRY
8. End.

---

5. CONCLUSION AND FUTURE WORKS

One of the most important benefits of cloud computing is the ability for a cloud client to adapt the number of resources used based on its actual use. This has great implications on cost savings as resources are not paid for when they are not used. This is made reality by dynamic scalability which is achieved through virtualization. The downside of virtualization is that it has a non-zero setup time that has to be taken into consideration for an efficient use of the platform. It follows that an accurate prediction method would greatly aid a cloud client in making its auto-scaling decisions.

In this paper, a new approach called 'AUTOPRED' is presented to the resource usage prediction problem based on identifying past patterns that are similar to the present use of the system. We present an algorithm for identifying the patterns by using an approximate matching approach. It uses a set of historic data to identify similar usage patterns to a current window of records that occurred in the past. The algorithm then predicts the system usage by interpolating what follows after the identified patterns from the historical data. Experiments have shown that the algorithm has good results when presented with relevant input data and, more importantly, that its results can improve by increasing the historic data size. This makes the evaluation of the algorithm be context dependent. As future work directions, we will be looking into ways that a relevant set of historic data can be composed for a particular application domain.

6. REFERENCES

[1] Wenrui L, Pengcheng Z, Zhongxue Y., 2012. A framework for self-healing service compositions in cloud computing environments. In the Proceedings of the19th International Conference on Web Services, IEEE Computer Society, 690-691.

[2] Armbrust M., Fox A., Griffith R., Joseph A., Katz R., Konwinski A., Lee G., Patterson D., Rabkin A., and Stoica I., 2010. A view of cloud computing, Communications of the ACM,50–58.

[3] Eyal Z., Israel C., Osnat M., 2011. The Power of Prediction: Cloud Bandwidth and Cost Reduction. SIGCOMM/ACM, pp.86-97.

[4] Eddy C, Frederic D., Adrian M., 2011. Forecasting for Cloud computing on-demand resources based on pattern matching. INRIA, 1-27.

[5] Yexi J., Chang-Shing P., Tao L., Rong C., 2011.ASAP: A Self-Adaptive Prediction System for Instant Cloud Resource Demand Provisioning. In the proceedings of 11th IEEE International Conference on Data Mining, IEEE Computer Society, 1104-1109.

[6] Avinash M., Mukesh M., Sanket D.,Shrisha R., 2011. Energy Conservation in Cloud Infrastructure. In the Proceedings of 5thAnnual IEEE International Systems Conference (IEEE SysCon 2011), Montreal, Canada.

[7] Arijit K., Xifeng Y., Shu T., Nikos A., 2012. Workload Characterization and Prediction in the Cloud: A Multiple Time Series Approach. In the Proceedings of IEEE/IFIP 3rdWorkshop on Cloud Management (CloudMan), 1287-1294.

[8] Bruno C., Diego D., Pierangelo D. S., Roberto P., Sebastiano P., Francesco Q., Paolo R., 2012. Automated Workload Characterization in Cloud-based Transactional Data Grids. In the Proceedings of the 26th International Parallel and Distributed Processing Symposium Workshops & PhD Forum, IEEE Computer Society, 1525-1533.

[9] Eddy C., Frederic D., Adrian M., 2011. Pattern Matching Based Forecast of Non-periodic Repetitive Behavior for Cloud Clients. Springer Science + Business Media B.V., 1-16.

[10] Ang L., Xuanran Z., Srikanth K., Xiaowei Y., Ming Z., 2011. CloudProphet: Towards Application Performance Prediction in Cloud. SIGCOM/ACM, 426-427.

[11] Gemma R., Javier A., Jordi G., 2010. Prediction of Job Resource Requirements for Deadline Schedulers to Manage High-Level SLAs on the Cloud. In the Proceedings of the 9thIEEE International Symposium on Network Computing and Applications, IEEE Computer Society, 162-167.

[12] John J. P., Kranthimanoj N., Brian K., Mo J. Prediction of Cloud Data Center Networks Loads Using Stochastic and Neural Models.

[13] Kranthimanoj N., Brain K., Jeff P., Mo J., 2010. On Prediction to Dynamically Assign Heterogeneous Microprocessors to the Minimum Joint Power State to Achieve Ultra Low Power Cloud Computing. IEEE.

[14] Eleni P., Damianos K., Nancy A., 2012. A Lightweight Framework for Prediction-based Resource Management in Future Wireless Networks. EURASIP Journal on Wireless Communications and Networking, 1-12.

[15] Rodrigo N. C., Rajiv R., Buyya R. Virtual Machine Provisioning Based on Analytical Performance and QoS in Cloud Computing Environments.