Legacy Application Workload Prediction Scheme by Sequentially segmented pattern (SSP) in Cloud Computing Infrastructure

Dr.P.VijayaKarthick[1], Dr.E.Gajendran[2], Mr.Y.M.Vijayakumara[3] , Mrs.M.Suguna[4]

[1] Professor, Cognizant Innovative research Lab, Dept of ISE, Sir. M Visvesvaraya Institute of Technology, Bengaluru, Karnataka, India- 562157, vijay691988@gmail.com
[2] Associate Professor, Dept. of CSE, Oxford college of Engineering, Tamilnadu.
[3] Assistant Professor, Dept of ISE, Sir. M Visvesvaraya Institute of Technology,Bengaluru
[4] Assistant Professor, Dept of ISE, Sir. M Visvesvaraya Institute of Technology,Bengaluru,

---

Abstract - Predicting Cloud performance could be a advanced task as a result of heterogeneous resource nodes are concerned in a distributed environment. Long execution work on a Cloud is even tougher to predict because of significant load fluctuations. The ability to accurately predict future resource capabilities is of great importance for applications in hierarchical Cloud environment. Our contribution is to predict workload by sequentially segmented pattern(SSP) in Cloud infra structure. The hierarchical Cloud environment contains set of resources and machines in each resource. The resources provided are in a hierarchical format using some priority. The workload is submitted to the resources based on the hierarchy. The purpose is to smooth the prediction process from being disturbed by load fluctuations. We use sequentially segmented pattern(SSP) method to reduce prediction errors. Some of the parameters used for job creation and resource creation are used to reduce the prediction errors using sequentially segmented pattern(SSP). Poise window is generated by DALP interval to get the load index range for future workload. Finally Load prediction is done by Autoregressive moving Average. ARMA models can also be used to forecast behavior of a time series from previous values . Such a prediction can be used as a baseline to evaluate possible importance of other variables to the system. At the end, we have a tendency to discuss extended analysis issues and gear development for Cloud performance prediction.

Index Terms—Cloud computing, performance forecasting, workload characterization, auto regression moving average method,sequentially segmented pattern(SSP), Data Aggregation based long term prediction mechanism, and parallel applications.

1. INTRODUCTION:

AGGREGATED Cloud performance is directly associated with the collective workload to be dead on an outsized range of processors scattered on all collaborating Cloud sites. Forecasting the collective Cloud workload could be a terribly difficult task [1], [3], [13], [34], [43] as a result of heterogeneous resources are cosmopolitan beneath the management of various administrations. We have a tendency to propose a replacement reconciling approach to forecasting workload on process Clouds among a poise window that is dynamically trained with the load variations.

The Cloud workload is diagrammatical by a collective load index among all processors. The load index X(i) is that the percentage of processors utilized among a unit measure [i-1,i]. All discrete-time instances i are denoted by nonnegative integers. For simplicity, we have a tendency to assume five minutes per time step. Load index reflects the central processor utilization rate among all the processors during a Cloud platform. For example, X(i)=0.65 implies that sixty five percent processors are busy throughout the observation time.

Workload is tough to predict as a result of the dearth of runtime information on job planning and resource allocation on remote machines [22], [29]. Expected workload might contain errors, if loading noises can not be sift out. Some previous workload calculation ways have unheeded couple problems: One is that the workloads measure errors and another is that the load information noise introduced by employment fluctuation [11], [19].

In this paper, we provide an aggregation technique to predict the workload for parallel execution on Cloud resource sites. We use sequentially segmented pattern(SSP) filter to sift out potential errors. Sifting out noises from workload fluctuation and menstruation errors, one will predict Cloud workload in high perfect way. This prediction theme will forecast execution time [44], [46] and guide the task planning ways [21], [37].

Traditional point prediction ways [11], [40], [41] apply a awfully short prediction window. though they'll work well to predict hardware load in centralized computer systems, they are doing not work well on large-scale production Clouds because long execution time is anticipated. In fact, point value prediction will hardly cowl employment fluctuation during a long time frame. In general, the load index is employed to estimate the percentage of peak performance realizable on a given Cloud. However, it's rather difficult to predict hardware utilization rate within the future [21]. The Cloud performance in T flops may be foretold by dividing the total workload integrated over the whole observation period by the execution time of job completed [4], [11], [19].

Workload managers in massive Cloud infrastructures square measure notoriously weak in crucial correct planning eventualities, which affects the applying execution time on the Cloud. This paper is regarding predicting the longer term workload among a reasonable poise range. The narrower is that the prediction range, the upper is that the accuracy of forecasting. Extended-range workload prediction is absolute to have some errors. However we tend to attempt to minimize the prediction errors by mistreatment lookahead filtering techniques among a trained poise range.

The residue of the paper is arranged as follows Section 2 introduces connected works and descriptions our distinctive approach. Section 3 discusses the proposed work. Section 4 reports the experimental results. Finally, in Section 5, we summarize the contributions and recommend directions for further analysis.

2. RELATED WORK:

In the past, several analysis teams tried to predict Cloud performance [5], [3], [13], [14], [9], [11]. Cloud workload varies with time however it's correlate in several time spans [10], [41]. Thus, system workload is foreseeable from checking the historical performance trace. By associating historical workload knowledge, we tend to predict the long run employment.

F'son has developed some workload models for parallel computers [12], [30]. Berman's cluster at UCSD has developed tools for forecasting parallel applications on Clouds [34]. The Cloud analysis cluster junction rectifier by Fortes at University of Everglade State has developed a prophetic application performance model for machine Clouds [24]. Mainframe workloads were characterized in [6] and [36]. Chiang Hon et al. [31] have studied the dynamic mapping problems in heterogeneous systems. Risk-tolerant programming of parallel tasks on Clouds is studied by Song et al.[37]. Li and associates have rumored progress on employment characterization for Clouds [24], [25], [26]. specially, Li's PhD Thesis [25] has given a comprehensive treatment of this branch of knowledge.

The idea of poise range was initially projected by Schopf and Berman [34] to deal with the accuracy issue. Historical workload knowledge are utilized by tendency-track models, e.g., AR model or polynomial fitting ways [32], [33], to forecast future employment standing. However, this tendency may be distorted or hid in shrieking knowledge, which will consequently impair the accuracy of employment prediction. Workload disparity and resource consumption in Cloud environments exhibit a large vary of dynamics, such as sudden native amendment, abrupt level amendment, etc., [27].

Adaptive techniques are introduced. the aim is to capture the dynamic characteristics. Normally, 2 varieties of adaptation approaches are introduced to enhance prediction accuracy. One approach is that the use of adaptive static predictor [41], [47] with mounted parameters as employed in ARmodel. This scheme works beneath the hypothesis that predicctor varies with resource varieties [40]. Still, the predictor for a particular resource may additionally fluctuates by time. This approach is to select the most effective predictor among variety of predictors with the low forecast errors for a particular load pattern.

The second approach is adaptive prediction enforced with parameter adaptation [27]. once associate adaptation is triggered by workload variation, this adaptive forecast automatically change some system data to adapt with the resource utilization outline. The pupose of this approach is to realize lower errors in workload forecast than exploitation AR technique with mounted parameters. Jiang et al. [20] extend the prediction by exploitation Markoffmodel-based meta predictor additionally to seasonal variation recognition for 1-step-ahead forecast.

Multiple resource forecasting model is projected in [27], which uses both autocorrelation and cross correlation to realize a higher forecasting accuracy. Several employment prediction ways [4], [10], [15], [16],[20], [34], [40], [41], [46] measured mean or median performance, or exploitation AR, Markov model, or seasonal variation to predict performance with various look ahead times. in a very large-scale Cloud setting, tasks typically need long term to run, thus, the task computer hardware needs an oversized look ahead span to predict the performance [28].

Network Weather prediction Service (NWPS) in Australia [40] provides a dynamically observance and statement technique to implement 1-step-ahead prediction of work load in cloud which have impact in its performance.Numerous prediction ways are used along within the NWS to forecast the performance of a Cloud system. Dinda and O'Hallaron [10] calculate the prediction power of many models, together with the AR, Moving Average (MA), etc., methods. Their analysis results show that a simple predictor model like AR is enough for one CPU load prediction. In [44], many 1-step lookahead predictors are evaluated. Static prediction ways are solely effective with steady employment. adaptive prediction is needed to manage time-varying workloads. The prediction of future price is adjusted in keeping with the magnitudes of the last workload measure. Standard point price prediction models are typically inaccurate, since they'll solely represent one purpose in a very vary of possible behaviors. .

Adaptive predictor integration [47] adopts associate approach for predictor integration supported learning historical predictions. Classification algorithms just like the k-nearest neighbor are support to direct erudition. This approach can achieve higher predictor accuracy. A multiresource prediction model was projected in [27] exploitation each autocorrelation of single resource and cross link over multiple resources.

In [48],algorithm presents a performance modelling framework.,it report ongoing progress to develop a general performance prediction framework to predict and explain the performance of scientific applications on current and future HPC platforms. The framework is not designed for a specific application or architecture but is designed to work for an arbitrary application on an arbitrary machine. The LINPACK benchmark is further used to investigate methods to reduce the time required to make accurate performance predictions with the framework. In previous work we introduced our convolution method that is a mapping of an application's signature (a representation of an applications fundamental operations) onto a machine profile (a characterization of a machine's ability to perform fundamental operations) to arrive at a performance prediction.

In [49], the future of Cloud computing and on the role of the Cloud sim Toolkit in future Cloud standards is performed. UNIX is specified as an operating system focused low-end systems being used for research and particularly software development environments. It was designed to be portable, and run on nearly any hard-ware with a modest porting report. It was also built as a small, minimal system, with much of the functionality provided in libraries which were not generally expected to be standard. Three distinct forces have been pushing the Unix/Linux community to a new focus around shared interfaces, function, and implementations. For multi-billion dollar business, For such companies, a large number of incompatible Unix systems were a porting and support cost and an impediment to growth. Second, By promising compatibility across many of these platforms, Microsoft was able to gain porting commitments from these third party software companies over many of the Unix dialects. Third, Linux had begun to grow like wilder based on tight control by a strong technical team, innovative licensing (the Gnu Public License) which prevented fragmentation, and a strong free software ideological culture. Cloud computing is revolutionary by enabling access to unprecedented computing power and shared information. With almost the entire IT industry, certainly all the major commercial companies launching large corporate-wide projects to take advantage of the coming service-oriented paradigm shift, a real danger exists that Cloud could fragment into incompatible islands, forced upon us by standards driven mainly by short-term commercial interests. We hope the Cloud sim Alliance will continue to follow the Internet style and spirit, encouraging cooperation and community building, by maintaining an open, non-commercial attitude and a modular Cloud simToolkit architecture

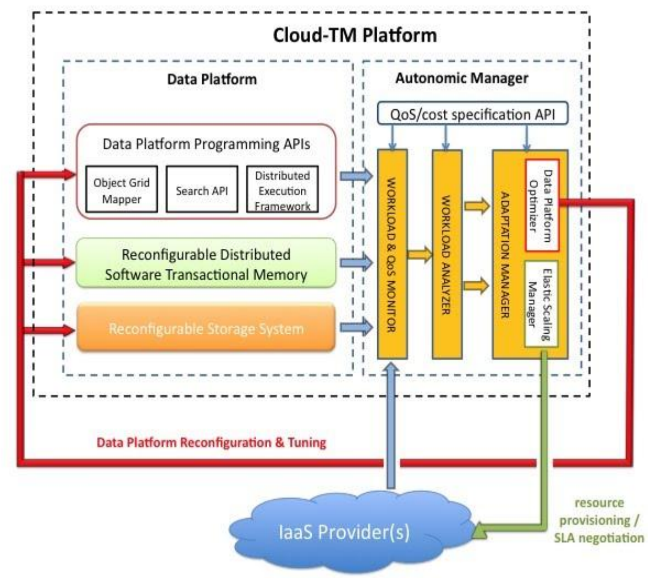

Fig1:Cloud Workload Architecture

3. PROPOSED WORK:

Contribution of this paper is to find the Legacy Application Workload Prediction Scheme and predict workload of Cloud performance. The hierarchical Cloud environment contains set of resources and machines in each resource. The resources provided are in a hierarchical format using some priority. The workload is submitted to the resources based on the hierarchy. The purpose is to smooth the prediction process from being disturbed by load fluctuations. The sequentially segmented pattern(SSP) filter to reduce prediction errors. Some of the parameters used for job creation and resource creation are used to reduce the prediction errors using sequentially segmented pattern(SSP) filter. Poise window is generated by DA2LP interval to get the load index range for future workload. Finally Load prediction is done by Autoregressive moving Average.

3.1. Hierarchical Cloud Environment

The basic Cloud environment is shown in fig1. In the hierarchical scheme, different levels of schedulers share the Scheduling process. The higher-level schedulers manage larger sets of resources and lower level schedulers manage smaller sets of resources. A higher-level scheduler has no direct control of a resource if there is one lower-level scheduler between the higher–level scheduler and the resource. A higher-level scheduler can only consider the capability of the set of resources managed by a lower-level scheduler as a whole entity, and utilizes the capability through invoking the lower-level scheduler. Compared with the centralized Scheduling, hierarchical Scheduling addresses the scalability and the problem of single point failure issue. Nevertheless, it also retains some of the advantages of the centralized scheme.

In Hierarchical Cloud, the Cloud environment is initialized for workload prediction of Cloud performance. Resources needed for Cloud implementation is created. Cloud Resources are created with number of machines in a resource and resource characteristics. Cloud users are created to submit jobs in the Cloud environment for workload prediction. Here we set priority to the resources based on their baud rate. Based on the baud rate we categorized resources into three forms i.e. Maximum priority, Normal priority and Minimum priority. The Cloudlets are assigned to the resources based on this priority.

3.2. Flaw Reduction

In Flaw reduction , the workload prediction errors are reduced. We use sequentially segmented pattern(SSP)for flaw reduction. sequentially segmented pattern(SSP) could track the phase transition and find a consistent solution with realistic error estimates. sequentially segmented pattern(SSP)n filter is used to deal with high dimensionality data. It is a recursive filter suitable for large number of variables in geophysical models.

$$\mathbf{K=CH^T (HCH^T+R)^{-1}} \qquad (1)$$

Where 'H' is the observation matrix, 'K' is the sample covariance, 'C' is the ensemble covariance and 'R' is the covariance matrix.

Observation matrix is used to map the observed values to the vector of fitted value which is calculated as below

$$\mathbf{H=X(X^TX)^{-1} X^T} \qquad (2)$$

Ensemble Covariance, $\mathbf{C=AA^T/N}$ where $\mathbf{A=X-E(X)}$Covariance matrix is defined as the covariance between the i^{th}, j^{th} element of a random vector. It is calculated as shown below

$$\mathbf{\Sigma=E\ [(X-E[X])\ (X-E[X])^T]}$$

3.3. Sequentially segmented pattern(SSP) Generation

In this step, a sequence of poise windows is trained to smooth the workload prediction process. Poise windows are generated using Data Aggregation based Long term load prediction (DALP).In order to reduce the number of prediction step and increase the amount of useful input load details, the data aggregation conception is commenced. Here we define the range of the workload series. Aggregate the workload of each resources in order to find out the range of poise window. The algorithm for sequentially segmented pattern(SSP) is shown in fig 1.

3.4. Legacy Application work Load Prediction:

In Load prediction , the workload of Cloud performance is predicted using Autoregressive-Moving-Average. Autoregressive-moving-average (ARMA) models are mathematical models of the persistence, or autocorrelation, in a time series. Modeling can contribute to understanding the physical system by revealing something about the physical process that builds persistence into the series.

Given a series of time data, the ARMA model could be a tool for understanding and, perhaps, predicting future values during this series. The model consists of 2 elements, autoregressive (AR) and a moving average (MA) . The model is sometimes remarked because the ARMA (p,q) model wherever p is that the order of the autoregressive and q is the order of the moving average .

$$\mathbf{X_t = C + \varepsilon_t + \sum_{i=1}^{p} \varphi_i X_{t-i} + \sum_{i=1}^{q} \theta_i \varepsilon_{t-i}}.$$

Where $\varphi_1, \varphi_2 \ldots \varphi_p$ and $\theta_1, \theta_2, \ldots \theta_p$ are parameters, c is constant, $\varepsilon_t, \varepsilon_{t-1}$ are white noises.

3.5. Performance Evaluation

In performance evaluation module, the performance of workload prediction in hierarchical Cloud environment is evaluated by using cloud sim tool kit.. Performance is evaluated among the resources in the Cloud environment. The resource characteristics of each resource in the Cloud differs depends upon number of PEs(Processor Elements), speed rate of machines in the resource. Hence, load prediction performances of Cloud resources are evaluated. The performance analysis results is explain detail in the following section.

3.6.Legacy Application workload Prediction Algorithm.

In this section we analyze the results of our proposed algorithm with the existing algorithm in terms of their error performance value

Algorithm: Legacy Application workload Prediction

Input: Load during various time series X_{t-a}, X_{t-a+1}, .X_{t-1}, Daggr, Maggr, n

Output: Forecasted workload values for n look ahead span

1. Aggregate the existing time series $X_{t-a}, X_{t-a+1}, \ldots X_{t-1}$,in order to find out the new aggregated series $y_1, y_2, .. y_l$ by $D_{aggr.}$

2. Calculate the AR coefficients $\varphi_1, \varphi_2 \ldots \varphi_{aggr}$ based on $y_1, y_2, .. y_l$

3. For i=1 to [n/ D_{aggr}]

4. Forecast y_{l+i} according to $\{y_{l+i-Maggr}, y_{l+i-Maggr+1}, \ldots, y_{l+i-1}\}$

5. End

4.Performance results on using sequentially segmented pattern (SSP) filter

The Sequentially segmented pattern(SSP) Filter could track the phase transition and find a Fig 3: Data Aggregation based load prediction In this section we compare the sequentially segmented pattern(SSP) filter with other filters like kalman filter, particle filter in terms of their error reduction performance as shown in fig 2.In order to compare we use different sets of workload pattern.

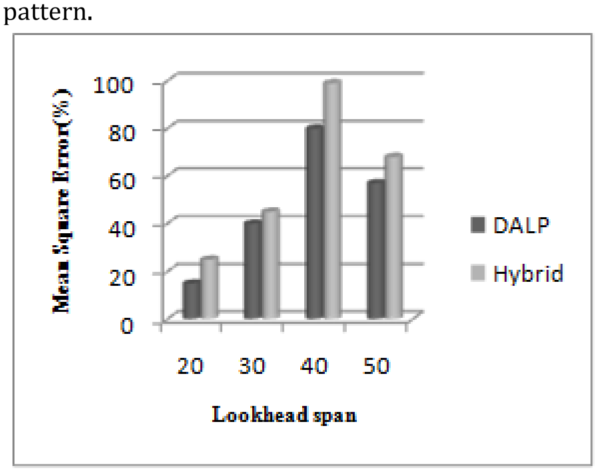

Fig 4.1: Error reduction performance of various filters

4.1. Performance of Using DALP

Here we show the results of using DALP concept in order to generate poise window. This is measured by using the mean square error. Mean Square Error (MSE) is one way to quantify the distinction between values inexplicit by estimator and also the true values of the number being calculable. MSE could be a risk perform, akin to the arithmetic mean of the square error loss or quadratic loss.

4.2. Poise Window Distribution

The poise window can be describe by mean confidence and standard deviation value. The fig 4.2 a) & b) shows the comparison of mean confidence and standard deviation of Hybrid and DALP method.

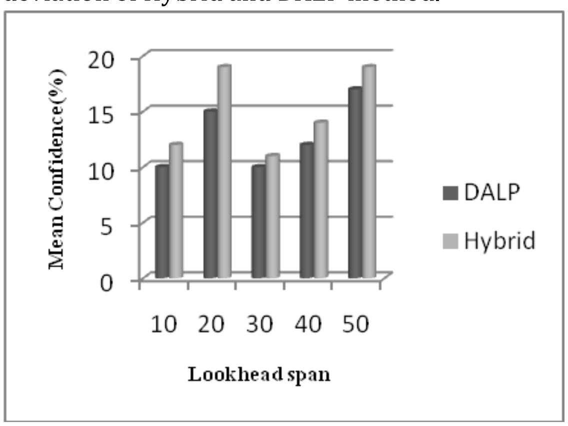

Fig 4.2(a): Comparison of Mean Confidence value

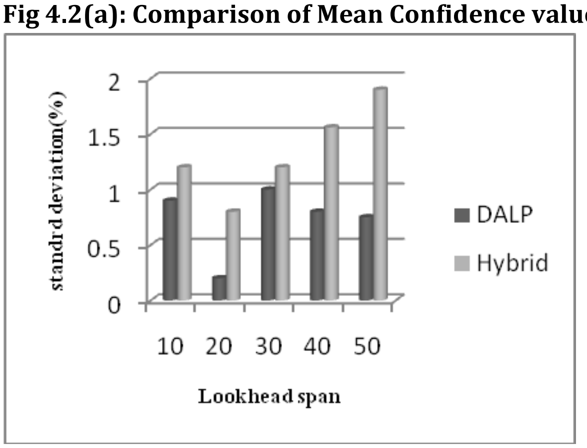

Fig 4.2(b): Comparison of Std deviation value

MSE measures the common of the squares of the "errors."The error is the amount by which the value implied by the estimator differs from the quantity to be estimated. Here the mean square value of the DALP algorithm is get compared to existing algorithms like Hybrid model. It is clearly shown in the fig 3 that DALP achieves low MSE.

From fig 6(a), 6(b) it is clearly shown that the DALP algorithm is best when compared to the existing algorithm.

5. Conclusion and Future Work.

The traditional point forecasting approach is insufficient to predict work in computational Clouds as a result of they can't handle load variations in a long execution environment. we have Legacy Application Workload Prediction Scheme which is developed a new look ahead work prediction schemes for assessing long-term Cloud performance in varying loads. The core idea introduced here is to use the efficient filter technique. The SSP filter is used to reduce the error value when the loads are measured. By using efficient filer technique we get more accurate workload measure. Here we introduce sequentially segmented pattern SSP filter in order to get accurate results. We obtained encouraging results to demonstrate the effectiveness of our filter scheme. We also increase the point value to a poise window, which is dynamically adjusted against load variations. The significant gain in prediction accuracy makes the DALP model terribly engaging to predict Cloud performance. The model was well-tried particularly effective to large work that demands terribly long execution time.

In future we use some advanced adaptive scheme in various level in order to address the Cloud scalability and reliability issues. Then build up some benchmarks for evaluating some Cloud performance. Try to implement our concept in real world Cloud application.

REFERENCES

[1] H. Akaike, "Fitting Autoregressive Models for Prediction," Annals of the Inst. of Statistical Math., vol. 21, no. 1, pp. 243-247, Dec. 1969.

[2] S. Akioka and Y. Muraoka, "Extended Forecast of CPU and Network Load on Computational Cloud," Proc. IEEE Int'l Symp. Cluster Computing and Cloud (CCCloud '04), pp. 765-772, 2004.

[3]AuverCloud Workload Report, http://gwa.ewi.tudelft.nl/pmwiki/ reports/gwa-t-4/trace_analysis_report.html, 2009.

[4] Cloud Computing: Making the Global Infrastructure a Reality, F. Berman, G.C. Fox, and T. Hey, eds. Wiley, 2003.

[5] L. Carrington, A. Snavely, and N. Wolter, "A Performance Prediction Framework for Scientific Applications," Future Generation Computer Systems, vol. 22, pp. 336-346, 2006.

[6] W. Cirne and F. Berman, "A Comprehensive Model of the Supercomputer Workload," Proc. IEEE Fourth Ann. Workshop
Workloads Characterization, 2001.

[7] A.A. Chien, X. Sun, and Z. Xu, "Viewpoints on Cloud Standards," J. Computer Science and Technology, vol. 20, no. 1, 2005.

[8] ChinaCloud Website, http://www.chinaCloud.edu.cn, 2006.

[9] M.J. Clement and M.J. Quinn, "Analytical Performance Prediction on Multicomputers," J. Supercomputing, pp. 886-894, 1993.

[10] P.A. Dinda and D.R. O'Hallaron, "Host Load Prediction Using Linear Models," Cluster Computing, vol. 3, pp. 265-280, 2000.

[11] Sourcebook of Parallel Computing, J. Dongarra, I. Fister, G. Fox, W. Gropp, K. Kennedy, L. Torczon, and A. White, eds. Kaufman Publishers, 2002.

[12] D.G. Feitelson, Workload Modeling for Computer Systems Performance Evaluation, Draft Version 0.7, Hebrew Univ. of Jerusalem, 2006.

[13] I. Foster and C. Kesselman, The Cloud: Blueprint for a New Computing Infrastructure. Kaufmann Publishers, 2004.

[14] G. Fox, D. Gannon, and M. Thomas, "Overview of Cloud Computing environments," Cloud Computing, F. Berman, G. Fox,
and T. Hey, eds., Chapter 20, Wiley, 2003.

[15] L. Gong, S.H. Sun, and E.F. Watson, "Performance Modeling and Prediction of Non-Dedicated Network Computing," IEEE Trans. Computers, vol. 51, no. 9, pp. 1041-1055, Sept. 2002.

[16] G.C. Goodwin and K.S. Sin, Adaptive Filtering Prediction and Control. Prentice-Hall, 1984.

[17] Cloud Workloads Archive, http://gwa.ewi.tudelft.nl/, 2009.

[18] Cloud'5000, ALADDIN-G5K: Ensuring the Development of Cloud '5000, https://www.Cloud5000.fr/, 2009.

[19] K. Hwang and Z. Xu, Scalable Parallel Computing. McGraw-Hill, 1998.

[20] S. Jang, X. Wu, and V. Taylor, "Using Performance Prediction to Allocate Cloud Resources," GriPhyN Technical Report 2004-25, pp. 1-11, 2004.

[21] M.A. Iverson, F. Ozguner, and L. Potter, "Statistical Prediction of Task Execution Time through Analytical Benchmarking for Scheduling in a Heterogeneous Environment," IEEE Trans. Computers, vol. 48, no. 12, pp. 1374-1379, Dec. 1999.

[22] M. Kalantari and M. Akbari, "Fault-Aware Cloud Scheduling Using Performance Prediction by Workload Modeling," J. Supercomputing, vol. 46, no. 1, pp. 15-39, Oct. 2008.

[23] R.E. Kalman, "A New Approach to Linear Filtering and Prediction Problems," Trans. ASME—J. Basic Eng., vol. 82, pp. 35-45, Mar. 1960.

[24] N.H. Kapadia, J.A. Fortes, and C.E. Brodley, "Predictive Application-Performance Modeling in a Computational Cloud
Environment," Proc. IEEE Int'l Symp. High-Performance Distributed Computing (HPDC), 1999.

[25] H. Li, "Workload Characterization, Modeling, and Prediction in Cloud Computing," PhD thesis, ASCI Graduate School, Univ. of Leiden, Jan. 2008.

[26] H. Li and R. Buyya, "Model-Driven Simulation of Cloud Scheduling Strategies," Proc. Third IEEE Int'l Conf. e-Science and Cloud Computing (eScience), 2007.

[27] J. Liang, K. Nahrstedt, and Y. Zhou, "Adaptive Multi-Resource Prediction in Distributed Resource Sharing Environment," Proc. IEEE Int'l Symp. Cluster Computing and Cloud (CCCloud '04), pp. 1-8,
2004.

[28] C. Liu, L. Yang, I. Foster, and D. Angulo, "Design and Evaluation of a Resource Selection Framework for Cloud Applications," Proc. 11th IEEE Int'l Symp. High-Performance Distributed Computing (HPDC '02), 2002.

[29] D. Lu, H. Sheng, and P. Dinda, "Size-Based Scheduling Policies with Inaccurate Scheduling Information," Proc. 12th IEEE Int'l Symp. Modeling, Analysis, and Simulation of Computer and Telecomm. Systems (MASCOTS '04), pp. 31-38, 2004.

[30] U. Lublin and D.G. Feitelson, "The Workload on Parallel Supercomputers: Modeling the Characteristics of Rigid Jobs," J. Parallel and Distributed Computing (JPDC), vol. 63, no. 11, pp. 1105-1122, Nov. 2003.
[31] M. Maheswaran, H.J. Siegel, D. Haensgen, and R.F. Freud, "Dynamic Mapping of a Class of Independent Tasks onto
Heterogeneous Computing Systems," J. Parallel and Distributed Computing (JPDC), vol. 59, no. 2, pp. 107-131, Feb. 1999.
[32] S.L. Marple Jr., Digital Spectral Analysis with Applications. Prentice- Hall, 1987.
[33] S.J. Orfanidis, Introduction to Signal Processing. Prentice-Hall, 1996.
[34] J. Schopf and F. Berman, "Performance Prediction in Production Environments," Proc. 12th Int'l Parallel Processing Symp., pp. 647- 653, Apr. 1998.
[35] W. Smith, I. Foster, and V. Taylor, "Predicting Application Run Times with Historical Information," J. Parallel and Distributed Computing (JPDC), vol. 64, no. 9, pp. 1007-1016, 2004.
[36] M. Snir and D.A. Bader, "A Framework for Measuring Supercomputer Productivity," Int'l J. High-Performance Computer Applications, vol. 18, no. 4, pp. 417-432, 2004.
[37] S. Song, K. Hwang, and Y. Kwok, "Risk-Tolerant Heuristics and Genetic Algorithms for Security-Assured Cloud Job Scheduling," IEEE Trans. Computers, vol. 55, no. 6, pp. 703-719, June 2006.
[38] V. Taylor, X. Wu, J. Geisler, and R. Stevens, "Using Kernel Computings to Predict Parallel Application Performance," Proc. Int'l Symp. High-Performance Distributed Computing (HPDC), 2002.
[39] G. Welch and G. Bishop, "An Introduction to the Kalman Filter," Univ. of North Carolina at Chapel Hill, 1995.
[40] R. Wolski, "Dynamically Forecasting Network Performance Using the Network Weather Service," J. Cluster Computing, vol. 1, pp. 119-132, Jan. 1998.
[41] R. Wolski, N. Spring, and J. Hayes, "Predicting the CPU Availability of Time-Shared Unix Systems," Proc. Eighth IEEE High Performance Distributed Computing Conf. (HPDC), 1999.
[42] Y. Yuan, Y. Wu, G. Yang, and W. Zheng, "Adaptive Hydrid Model for Long-Term Load Prediction in Computational Cloud," Proc. IEEE Eighth Int'l Conf. Cluster Computing and Cloud (CCCloud '08),
pp. 340-347, May 2008.
[43] Z. Xu and K. Hwang, "Early Prediction of MPP Performance: SP2, T3D, and Paragon Experiences," J. Parallel Computing, vol. 22, no. 7, pp. 917-942, Oct. 1996.
[44] L. Yang, I. Foster, and J.M. Schopf, "Homeostatic and Tendency- Based CPU Load Predictions," Proc. Int'l Parallel and Distributed Processing Symp. (IPDPS '03), pp. 42-50, 2003.
[45] L. Yang, X. Ma, and F. Muller, "Cross-Platform Performance Prediction of Parallel Applications Using Partial Execution," Proc. Supercomputing Conf., 2005.
[48] [49] "VIEWPOINTS ON CLOUD STANDARDS" Andrew A. Chien1, Xian-He Sun2, and Zhi-Wei Xu3. 1Computer Science and Engineering, University of California, San Diego,2005