Minimizing the Makespan and Economic Cost of Schedule for the Grid Applications

Avdhesh Gupta
IMS Engineering College
Ghaziabad, India

Pankaj Agarwal
IMS Engineering College
Ghaziabad, India

Shalini Gupta
Virendra Swarup Group of Institutions
Kanpur, India

Abstract

Grid computing is a distributed computing taken to next evolutionary level. In this work, a static methodology has been adopted for defining the weights of the computational tasks and communicating edges. Also, we defined the execution time (makespan) as the total time between the finish time of exit task and start time of the entry task in the given Directed Acyclic Graph (DAG). The algorithm has been implemented for evaluation of time and cost of different random task graph or DAG of different graph size. Also, the algorithm has been executed in a grid of heterogeneous cluster of different sizes with four resources in each cluster. The primary work is to find the primary scheduling i.e., total execution time and total cost with little or no changes in primary scheduling. We have proposed an efficient scheduling algorithm, which optimize the makespan and economic cost of the schedule and minimize the requirements of processors. The algorithm has been implemented to schedule different random DAGs onto different grids of heterogeneous clusters of various sizes.

Keywords

DAG, Grid, makespan, workflow

1. INTRODUCTION

Grid is a novel infrastructure for network computing on local or geographical scales that can dynamically represent heterogeneous computing resources. Grid computing is broadly used in many scientific and engineering application fields. Grid application addresses collaboration, data sharing, cycle sharing and other modes of interaction that involves distributed resources and services [1]. The field of Grid Computing is a manifestation of the development of distributed and cluster computing environments. The concept of a computational grid was first proposed by Ian Foster and Carl Kesselman in the mid 1990s. Since then the field has involved into one of the most exciting areas of study in the computing fraternity.

The Grid system is responsible for the execution of jobs submitted to it. The advanced Grid system will include a task scheduler which automatically finds the most appropriate machines on which a given job is to run. This resource selection [2] is very important in reducing the total execution time and cost of executing the tasks which depends on the task scheduling algorithm. The scheduling policy followed by the scheduler determines the Grid system throughput and utilization of the resources in to the grid.

In mathematics and computer science, a directed acyclic graph (DAG), is a directed graph with no directed cycles. That is, it is formed by a collection of vertices and directed edges, each edge connecting one vertex to another, such that there is no way to start at some vertex v and follow a sequence of edges that eventually loops back to v again. DAGs may be used to model several different kinds of structure in mathematics and computer science. A collection of tasks that must be ordered into a sequence, subject to constraints that certain tasks must be performed earlier than others, may be represented as a DAG with a vertex for each task and an edge for each constraint; algorithms for topological ordering may be used to generate a valid sequence. DAGs may also be used to model processes in which information flows in a consistent direction through a network of processors. The reachability relation in a DAG forms a partial order, and any finite partial order may be represented by a DAG using reachability. Additionally, DAGs may be used as a space-efficient representation of a collection of sequences with overlapping subsequences.

DAG has been extensively used in grid workflow modeling. Since the computational capacity of available grid resources tends to be heterogeneous, efficient and effective workflow job scheduling becomes essential. It poses great challenges to achieve minimum job accomplishing time while maintaining high grid resources utilization efficiency.

The Grid system is responsible for the execution of jobs submitted to it. The advanced Grid system will include a task scheduler which automatically finds the most appropriate machines on which a given job is to run. This resource selection [2] is very important in reducing the total execution time and cost of executing the tasks which depends on the task scheduling algorithm. The scheduling policy followed by the scheduler determines the Grid system throughput and utilization of the resources in to the grid.

2. EXISTING SCHEDULING TECHNIQUE

Non Duplication Technique: Two common approaches in non-duplication based scheduling are list scheduling and cluster-based scheduling.

List scheduling is one of most commonly used scheduling algorithms. In list scheduling, a weight is assigned to each task and edge, based on which an ordered task list is constructed by assigning priority for each task. Then, tasks are selected in the order of their priorities, and each selected task is scheduled to a computing host that can minimize a predefined cost function. As two typical list scheduling heuristics,

HEFT (Heterogeneous Earliest Finish Time) and CPOP (Critical Path on a Processor) are studied in [3]. The upward rank and downward rank of each task are computed at the beginning. HEFT algorithm always selects the task with the highest upward rank at each step. Then the selected task is assigned to a host that can minimize its earliest finish time. In contrast, CPOP algorithm always selects the task with the highest total rank (upward rank + downward rank) value. In order to minimize the total execution time, CPOP schedules all critical tasks onto a single host with the best performance. During execution, if a selected task is non-critical, it will be mapped to a host which could minimize its earliest finish time, as in HEFT. Both HEFT and CPOP have low complexity, i.e., lower algorithm execution time. However, the study in [4] observed that the performances of these two algorithms are affected dramatically by how to assign weights to the nodes and edges. In some extreme cases, different weight assignment approaches can lead up to 47.2% of performance difference. In another popular scheduling heuristic group scheduling, tasks are sorted into groups, under the constraint that tasks in the same group should be independent. Tasks then are scheduled group by group. The studies in [5] proposed a hybrid remapping heuristic. Tasks in a DAG are partitioned into levels so that there is no dependency among tasks at the same level. Then, tasks are mapped to computing hosts with task/host pairs using a static algorithm (e.g., baseline). First assigned a priority, then the tasks is considered in on-ascending order of priorities for scheduling on a set of available processors. Despite the fact that the quality of their schedules is usually worse than that of other algorithm classes, low complexity of the list-based algorithms still make them attractive alternatives. In cluster-based scheduling, processors are treated as clusters and the completion time is minimized by moving tasks among clusters [10]. At the end of clustering, heavily communicating tasks are assigned to the same processor, reducing the inter-processor communication.

Duplication Technique: The proposed algorithm is a duplication-based static scheduling algorithm and it differs from the previous algorithms by addressing the minimization of the schedule length and the number of processors used as separate problems to be optimized in two distinct phases. Real world problem in grid are multi objective means they required more than one objective. For e.g. Total execution time or make span, economy, reliability, trustworthiness, etc. We proposed a scheduling algorithm based on multi objective namely total economy cost/execution time.

3. EXISTING SCHEDULING TECHNIQUE

Computational Grids computing systems are emerging as a new paradigm for solving large-scale problems in science, engineering and commerce [6, 7]. They enable the creation of virtual enterprises (VEs) for sharing and aggregation of millions of resources (e.g. SETI@Home [8]) geographically distributed across organizations and administrative domains. They comprise heterogeneous resources (PCs, workstations, clusters and supercomputers), fabric management systems (single system image OS, queuing systems, etc.) and policies, and applications (scientific, engineering and commercial) with varied requirements (CPU, I/O, memory and/or network intensive).

In [7, 9–11], they proposed and explored the usage of an economics-based paradigm for managing resource allocation in Grid computing environments. The economic approach provided a fair basis in successfully managing decentralization and heterogeneity that is present in human economies Competitive economic models provide algorithms/policies and tools for resource sharing or allocation in Grid systems. The models can be based on bartering or prices. Most of the related work in Grid computing dedicated to resource management and scheduling problems adopt a conventional style where a scheduling component decides which jobs are to be executed at which site based on certain cost functions (Legion [12], Condor [13], AppLeS [14], Netsolve [15], Punch [16]). Such cost functions are often driven by system-centric parameters that enhance system throughput and utilization rather than improving the utility of application processing.

The bi-criteria scheduling approach may require several different criteria to be considered simultaneously when evaluating the quality of solution or a schedule. In general, scheduling directed acyclic graph (DAG) in grid occupies large number of computing resources or processors.

RPS (Resource Prediction System) project [17, 18] is a resources-oriented system for online prediction and scheduling. It carries on explicit prediction based on the resource signal, and realizes time series models to predict resource information of hosts. RPS is consisting of sensor library, time series prediction library, mirror communication template library, scripts and other auxiliary codes. The sensor library provides acquisition mechanism of resources information to monitoring component and the time series prediction library provide as a scalable, object-oriented C++ template, as well as several linear models for prediction component. It fits data on models and generates prediction through the most appropriate model, then evaluates its performance in application.

NWS (Network Weather Service) project [19, 20] is a distributed system for generation and publication of computing resources prediction, periodically and dynamically. It maintains a group of distributed performance sensors, such as CPU sensors, network sensors, etc. NWS collects information from these sensors on computing nodes, and predicts resource usages in certain time interval ahead, using multiple models such as mean based one, median based one, and autoregressive method. NWS and RPS are supplement to each other. For example, RPS can use NWS sensors, while NWS can use RPS prediction model. Latest versions of the two systems are both extended to support grid systems.

CORI (Collectors of Resource Information) project [21] designs a performance subsystem to enable DIET (Distributed Interactive Engineering Toolbox) project [22] interfacing with third-party performance prediction tools. They also mentioned the importance of prediction, though they didn't propose any prediction method themselves.

GHS (Grid Harvest Service) project [23] is a performance evaluation and task scheduling system for solving large-scale applications in shared environment. Its framework includes predictors in application-level and system-level, as well as interactions with other components within the system. Their efforts are dedicated on the systematic structure rather than prediction methodology, thus their implementation simply uses mean based method, whereas other prediction methods are welcomed to replace theirs, which provides wide extension space for further researches.

Resource prediction is based on resource monitoring. It sums up historical data for modeling, and seeks to find the variation principles of resources, and makes judgment or prediction of short-term or even long-term in future interval. Performance of different models is distinguished by the prediction techniques employed. Several representative ones are discussed and compared as follows:

Linear time series: Resource variations are considered a linear time series regression process in many researches [17, 18, 19, and 20]. Box-Jenkins models are a series of linear time series ones, which are also well known as AR-class models, including AR (purely autoregressive), MA (purely moving average), ARMA (autoregressive moving average), ARIMA (autoregressive integrated moving average), and ARFIMA (autoregressive fractionally integrated moving average). Subsequently, ARCH (autoregressive conditional heteroskedasticity) and GARCH (generalized autoregressive conditional heterosjedasticity) are also added based on secondary modeling to the error. AR-class models are universal thus other linear models like Markov process or Mean/Median process can also be expressed using AR-class models, partly or completely.

Wavelet analysis: In this method, resource variations are considered a superposition of multiple waveforms. It generates prediction based on periodicity in variations. It is doing well on signal with periodical behavior, and it has a good self-adaptability. While the drawback is that it is not feasible for the application with too much randomness, therefore it is usually combined with other techniques in modeling, for example in combination with support vector machine [25].

Stochastic information: It takes resource variations as a stochastic process [26]. This method is based on the assumption that the resource information follows normal distribution. However, it is not the truth in most of practical applications. Its reliability can be improved by adapting the original assumption that is replacing normal distribution with interval distribution. This method is simple, intuition and fast, while its limitation is that the distribution of the interval values must be unified.

Artificial neural network: The ANNs (Artificial Neural Networks) are powerful tools for self-learning, and they can generalize the characteristics of resource variations by proper training. ANNs are born with distributed architecture as well as robustness. They are suitable for multi-information fusion, and are competent for quantitative and qualitative analysis. ANNs have been employed by many researches in resource prediction. In the research of [27], it is indicated that the ANNs prediction are more accurate and outperform the methods in NWS. However, ANN's learning process is quite complex, thus is inefficient in modeling. Furthermore, the choices of model structures and parameters are lack of standard theory, so that it usually suffers from over-fitting or under-fitting with ill chosen parameters.

Support vector machine: As a promising solution to nonlinear regression problems, SVM (Support Vector Machine) [28] has recently been winning popularity due to its remarkable characteristics such as good generalization performance, the absence of local minima and sparse representation of the solution. The traditional regression techniques, including neural networks, are based on the ERM (Empirical Risk Minimization) principle, while SVM is proposed based on the SRM (Structural Risk Minimization) principle, which tries to control model complexity as well as the upper bound of generalization risk, rather than minimizing the training error only, thus is expected to achieve better performance than traditional methods. Prem and Raghavan [29] have explored the possibility of applying SVM to forecast resource measures and indicated that the SVMs forecasts are more accurate and outperform the NWS methods, such as Autoregressive ones. Available bandwidth and host load are two representative resource elements in computing Grid, therefore their benchmark data sets are chosen to evaluate the performance of prediction models. We prefer using public data rather than historical data recorded by ourselves, for the purpose of giving comparable and reproducible results. For available bandwidth prediction, we choose ―iepm-bw.bnl.gov.iperf2‖ [30]. It is published by the Stanford Linear Accelerator Center, University of Stanford. For host load prediction, we choose ―mystere10000.dat‖ [31]. It is published by the Department of Computer Science, University of Chicago. Mean/Median based ones. In general, scheduling directed acyclic graph (DAG) in grid occupies large number of computing resources or processors. To address this problem, we present an effective bi-criteria scheduling heuristic called schedule optimization with duplication-based bi-criteria scheduling algorithm (SODA)[32]. In general, minimization of total execution time (or makespan) of an application schedule is applied as the most important scheduling criteria used by nearly all existing scheduling heuristics. Most of the existing grid computing systems are based on system-centric policies whose objectives are to optimize the system-wide metrics of performance, i.e., makespan. The convergence of grid computing toward the service-oriented approach is fostering a new vision where economic aspects are key elements in increasing the adoption of computing as a utility. In current economic market models, economic cost (cost of executing a workflow on grid) has been considered as an important scheduling criterion to employ the user-centric policies, since different resources, belonging to different organizations, may have different polices of charging.

Paradoxically, the literature shows that in the majority of problems addressed, schedulers were generated keeping a single criterion. Considering multiple criteria enables us to propose a more realistic solution. Therefore, an efficient multi-criteria scheduling heuristic is required for execution of workflow on grid while assuring the high speed of communication, reducing the tasks execution time and economic cost. As the DAG scheduling problem in grid is NP-complete, we have emphasized on heuristics for scheduling rather than the exact methods. In literature, many bi-criteria scheduling algorithms have been, which minimizes both the makespan and economic cost of the schedule but only few of them address the workflow type of applications. [33]Proposed an efficient bi-criterion scheduling algorithm called ‗dynamic constraint algorithm' (DCA) based on a sliding constraint. Our work presents an approach where processor requirement is minimized under two criteria (i.e., makespan and economic cost) for workflows. The study also shows that heuristics performing best in static environment [34] have the highest potential to perform better in more accurately modeled grid environments.

4. APPLICATION MODEL TO BE DEVELOP

DAG has been extensively used in grid workflow modeling. Since the computational capacity of available grid resources tends to be heterogeneous, efficient and effective workflow job scheduling becomes essential. It poses great challenges to achieve minimum job accomplishing time while maintaining high grid resources utilization efficiency. A workflow consists of a sequence of connected steps. It is a depiction of a sequence of operations, declared as work of a person, a group of persons, an organization of staff, or one or more simple or complex mechanisms. Workflow may be seen as any abstraction of real work. For control purposes, workflow may be a view on real work under a chosen aspect, thus serving as a virtual representation of actual work. The flow being described may refer to a document or product that is being transferred from one step to another. A workflow is a model to represent real work for further assessment, e.g., for describing a reliably repeatable sequence of operations. More abstractly, a workflow is a pattern of activity enabled by a systematic organization of resources, defined roles and mass, energy and information flows, into a work process that can be documented and learned.[3][4] Workflows are designed to achieve processing intents of some sort, such as physical transformation, service provision, or information processing.

Workflow concepts are closely related to other concepts used to describe organizational structure, such as silos, functions, teams, projects, policies and hierarchies. Workflows may be viewed as one primitive building block of organizations. The relationships among these concepts are described later in this entry. The term workflow is used in computer programming to capture and develop human-to-machine interaction.

5. MOTIVATION

Resource management and task scheduling are very important and complex problems in grid computing. In grid environment, autonomy resources are linked though the internet to form a huge virtual seamless environment. The resources in the grid are heterogeneous and the structure of the grid is changing all the time. With the grid become a viable high performance computing alternative to the traditional supercomputing environment, various aspects of effective Grid resources utilization are gaining significance. With its multitude of resources, a proper scheduling of time and cost across the grid can lead to improved overall system performance and a lower turn - around time for individual jobs.

6. PROBLEM STATEMENT

The main issue is to analyze for minimization the total execution time (makespan) and economic cost of the schedule obtained for executing the grid applications. The makespan of a schedule is the earliest finish time of exit task. Similarly, the economic cost is the sum of costs of executing tasks over grid resources.

7. RESULT ANALYSIS

In this section we will discuss the computation of makespan or total execution time followed by cost of executing the DAG. In this we can use the DAG of different sizes. The algorithm has been implemented in JAVA for evaluation of time and cost of different random task graph or DAG of different graph size (100,200,300,400,500).The algorithm have been executed in a grid of heterogeneous cluster of different sizes(5,10,15,20,25) with four resources in each cluster. Algorithm is run to find the primary scheduling i.e., total execution time and the secondary scheduling i.e., total cost with little or no changes in primary scheduling. The results and the graphs reveal that the proposed scheduling approach is Time and Cost effective for scheduling in heterogeneous environment as shown by the following graphs:

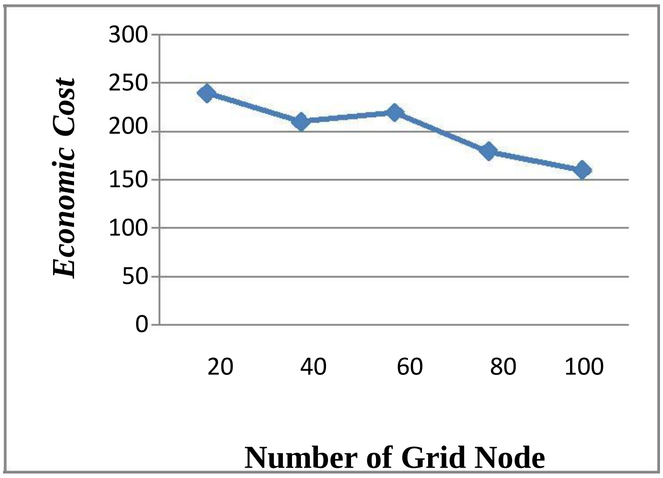

Fig. 1: Effect of grid sizes on EC (number of tasks=100)

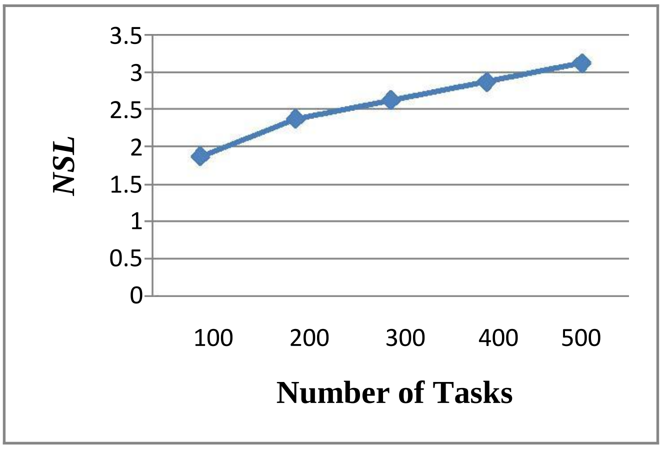

Fig.2: Normalize schedule length on different workflow sizes (number of processor=50)

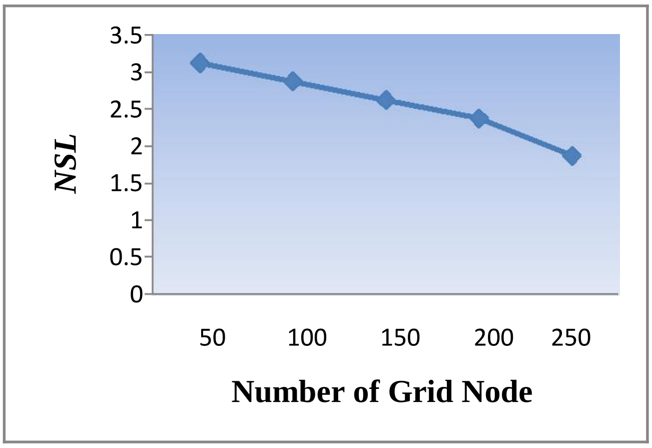

Fig.3: Normalize schedule length on different grid (number of task=500)

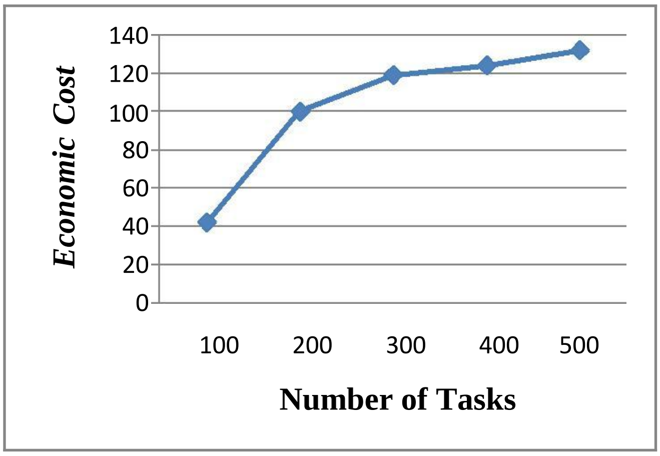

Fig.4: Effective cost on different workflow sizes (number of processor=50)

Fig. 1 shows that when we increase the number of processor for a given number of tasks then economic cost should become lesser. In Fig. 2 we shows that when we increase the number of tasks for a given processors then NSL become greater. Fig. 3 shows that when we increase number of grid node for a given number of tasks then value of the NSL become lesser. In Fig. 4 when we increase the number tasks then economic cost will be greater for a given number of processor.

Table 1: Variable used in the simulation

Grid environment layout	
Number of grid resources	[20,100]
Resource bandwidth	[100 Mbps,1 Gbps]
Number of tasks	[100,500]
Computation cost of the tasks	[5 msec,200 msec]
Data transfer size	[20 Kbytes,2 Mbytes]
Resource capability(MIPS)	[220,580]
Execution cost(MIPS)	[1-5 grid dollar per MIPS]

8. CONCLUSION

Computational Grids enable the creation of a virtual computing environment for sharing and aggregation of distributed resources for solving large-scale problems in science, engineering and commerce. The resources in the Grid are geographically distributed and owned by multiple organizations with different usage and cost policies. They have a large number of self-interested entities (distributed owners and users) with different objectives, priorities and goals that vary from time to time. The management of resources in such a large and distributed environment is a complex task. In this, a novel bi-criteria workflow scheduling approach has been presented and analyzed. We have proposed an efficient scheduling algorithm called Time and Cost based Task Scheduling in Grid Environment, which optimizes the makespan and economic cost of the schedule and minimizes the requirements of processors. The algorithms have been implemented to schedule different random DAGs onto different grids of heterogeneous clusters of various sizes. The schedule generated by algorithm is better than other related bi-criteria algorithms in respect of both makespan and economic cost.

Future work would involve developing a scheduling system which also considers the load balancing or other objectives through which we minimizes the makespan and the cost.

9. REFERENCES

[1] Fost, Ian, Carl Kesselman, Steve Tuecke, The Aanatomy of the Grid: Enabling Scalble Virtual Organization, International Journal of Supercomputer Application, 2001.

[2] Klaus Krauter, Rajkumar Buyya, and Muthucumaru Maheswaran. ―A Taxonomy and Survey of Grid Resource Management System for Distributed Computing‖, Software: Practice and Experience (SPE) journal, Wiley Press, USA, 2001.

[3] H. Topcuoglu, S. Hariri, and M. Wu, ―Performance Effective and Low-complexity Task Scheduling for Heterogeneous Computing‖, IEEE Transactions on Parallel and Distributed Systems, 13(3), pp.260-274, 2002.

[4] R. Sakellariou and H. Zhao, ―A Hybrid Heuristic for DAG Scheduling on Heterogeneous Systems‖, in Proceedings of the 13th Heterogeneous Computing Workshop (HCW'04), Santa Fe, New Mexico, USA, pp.26 - 30, 2004.

[5] [M. Maheswaran and H. J. Siegel, ― A Dynamic Matching and Scheduling Algorithm for Heterogeneous Computing Systems‖, in Proceedings of the Seventh Heterogeneous Computing Workshop, 1998.

[6] Abramson D, Giddy J, Kotler L.‖ High performance Parametric Modeling with Nimrod/G: Killer Application for the Global Grid‖? Proceedings International Parallel and Distributed Processing Symposium (IPDPS 2000), Cancun, Mexico, 1–5May 2000. IEEE Computer Society Press: Los Alamitos, CA, 2000.

[7] Buyya R, Abramson D, Giddy J.‖ A Case for Economy Grid Architecture for Service-Oriented Grid Computing‖. Proceedings of the International Parallel and Distributed Processing Symposium: 10th IEEE International Heterogeneous Computing Workshop (HCW 2001), 23 April 2001, San Francisco, CA. IEEE Computer Society Press: Los Alamitos, CA, 2001.

[8] Sullivan III WT, Werthimer D, Bowyer S, Cobb J, Gedye D, Anderson D.‖ A New Major SETIPproject Based on Project Serendip Data and 100,000 Personal Computers‖. Proceedings of the 5th International Conference astronomy 1997 http://setiathome.ssl.berkeley.edu/woody paper.html.

[9] Buyya R, Abramson D, Giddy J. Nimrod/G: Architecture for a Resource Management and Scheduling System in a Global Computational Grid‖. Proceedings 4th International Conference and Exhibition on High Performance Computing in Asia-Pacific Region (HPC ASIA 2000), Beijing, China, 14–17 May 2000. IEEE Computer Society Press: Los Alamitos, CA, 2000.

[10] Rajkumar Buyya, David Abramson, and Srikumar Venugopal, "The Grid Economy, Special Issue on Grid Computing", Proceedings of the IEEE, Manish Parashar and Craig Lee, IEEE Press, New York, USA, ISSN 0018-9219, Vol. 93, Issue 3, pp. 698 - 714, March 2005.

[11] R. Garg and A. K. Singh ―Reference Point Based Evolutionary Approach for Workflow Grid Scheduling‖. Proceedings of International Journal of Information and Electronics Engineering, Vol. 2, No. 4, July 2012

[12] Buyya R, Giddy J, Abramson D. ― An Evaluation of Economy-Based Resource Trading and Scheduling on Computational Power Grids for Parameter Sweep Applications‖. Proceedings of the 2nd International Workshop on Active Middleware Services AMS 2000), Pittsburgh, PA, 1 August 2000. Kluwer Academic Press, 2000.

[13] Chapin S, Karpovich J, Grimshaw A, ―The Legion Resource Management System‖. Proceedings of the 5th Workshop on Job Scheduling Strategies for Parallel Processing , San Juan, Puerto

Rico, 16 April 1999. Springer: Berlin, 1999.

[14] Litzkow M, Livny M, Mutka M. Condor—a Hunter of Idle Workstations. Proceedings 8th International Conference of Distributed Computing Systems (ICDCS 1988), San Jose, CA, January 1988. IEEE Computer Society Press: Los Alamitos, CA, 1988.

[15] Berman F, Wolski R. The AppLeS Project: A status report. Proceedings of the 8th NEC Research Symposium, Berlin, Germany, May 1997.

[16] Casanova H, Dongarra J. NetSolve: ―A Network Server for Solving Computational Science Problems‖. International Journal of Supercomputing Applications and High Performance Computing 1997; 11(3):212–223.

[12] Kapadia N, Fortes J. PUNCH: ―Architecture for Web-Enabled Wide-Area Network-Computing‖. Cluster Computing: The Journal of Networks, Software Tools and Applications 1999; 2(2):153–164.

[13] P. A. Dinda, ―Design, Implementation, and Performance of an Extensible Toolkit for Resource Prediction in Distributed Systems‖, IEEE Trans. Parallel Distrib. Syst., 2006, 17(2), pp. 160-173.

[14] P. A. Dinda, and D. R. O'Hallaron, ―Host Load Prediction Using Linear Models‖, Cluster Computing, 2000, 3(4), pp. 265-280.

[15] D. M. Swany, and R. Wolski, ―Multivariate Resource Performance Forecasting in the Network Weather Service‖, In Proceedings of the 2002 ACM/IEEE conference on Supercomputing Baltimore, Maryland, November 2002, pp. 1-10.

[16] R. Wolski, L. Miller, G. Obertelli, and M. Swany, ―Performance Information Services for Computational Grids‖, In: Resource Management for Grid Computing, J. Nabrzyski, J. Schopf, and J. Weglarz, editors, Kluwer Publishers, Fall, 2003.

[17] E. Caron, A. Chis, F. Desprez, and A. Su, ―Design of Plug-In Schedulers for a GRIDRPC Environment‖, Future Generation Computer Systems, 2008, 24(1), pp. 46-57.

[18] E. Caron, and F. Desprez, ―DIET: A Scalable Toolbox to Build Network Enabled Servers on the Grid‖, International Journal of High Performance Computing Applications, 2006, 20(3), pp. 335-352.

[19] A. K. Talukder, M. Kirley, and R. Buyya, ―Multi-Objective Differential Evolution for Scheduling Workflow Applications on Global Grids,‖ John Wiley and Sons, Ltd., DOI: 10.1002/cpe.1417,

[20] 2009.L.-T. Lee, D.-F. Tao, and C. Tsao, ―An Adaptive Scheme for Predicting the Usage of Grid Resources‖, Comput. Electr. Eng. Pergamon Press, Inc., Tarrytown, NY, USA, 2007, 33(1), pp. 1-11.

[21] Q. Huang, N. Xiao, and B. Liu, ―Grid Load Forecasting Based on Least Squares Support Vector Machine‖, Computer Technology and development, 2007, 17(6), pp. 32-35.

[22] B. Vandy, and G. Bruno, ―Brokering Strategies in Computational Grids using Stochastic Prediction Models‖, Parallel Computing, May 2007, 33(4-5), pp. 238-249.

[23] Chee Shin Yeo and Rajkumar Buyya, "Service Level Agreement based Allocation of Cluster Resources: Handling Penalty to Enhance Utility", Proceedings of the 7th IEEE International Conference on Cluster Computing, Cluster 2005, IEEE CS Press, Los Alamitos, CA, USA, Boston, Massachusetts, USA, September 27-30, 2005.

[24] Cost Marcos Dias de Assuncao and Rajkumar Buyya, A Cost-Aware Resource Exchange Mechanism for Load Management across Grids, Proceedings of the 14th IEEE International Conference on Parallel and Distributed Systems (ICPADS 2008, IEEE CS Press, Los Alamitos, CA, USA), Dec. 8-10, 2008, Melbourne, Australia.

[25] Srikumar Venugopal and Rajkumar Buyya, Cost-based Scheduling for Data-Intensive Applications on Global Grids, Proceedings of the 14th IEEE International Symposium on High Performance Distributed Computing (HPDC-14, 2-pages Poster Paper, IEEE CS Press, Los Alamitos, CA, USA), July 24-27, 2005, Research Triangle Park, North Corolina, USA.

[26] Jia Yu, Rajkumar Buyya, and Chen Khong Tham, Cost-based Scheduling of Workflow Applications on Utility Grids, Proceedings of the 1st IEEE International Conference on e-Science and Grid Computing (e-Science 2005, IEEE CS Press, Los Alamitos, CA, USA), Dec. 5 -8, 2005, Melbourne, Australia.

[27] Weina Lu, Shoubao Yang, Leitao Guo, Rui Zhang, ―Reputation-aware Transaction Mechanisms in Grid Resource Market‖, Sixth Intl: conf: on Grid and Cooperative computing(GCC‗07), 16-18 Aug, 2007

[28] Chunlin, L. and L. Layuan, "A Pricing Approach For Grid Resource Scheduling With QoS Guarantees " Fundam. Inf. 76 (1-2): 59-73, 2007.

[29] A. Eswaradass, X.-H. Sun, and M. Wu, ―A Neural Network Based Predictive Mechanism for Available Bandwidth‖, In Proceedings of the 19th IEEE International Parallel and Distributed Processing Symposium (