Survey of Resource and Job Management for Load Balancing In Grid Computing

Sakadasariya Achyut R.

Abstract: Load balancing is the process of load distribution, handling incoming requests and better resource utilization. In a distributed grid computing system it is desirable to achieve an efficient distribution of workload among systems so that each and every machine would have the same workload. No machine should remain idle while other machines are overloaded. Load distribution is done to achieve better response time, better resource utilization and thus improved performance. For improve the performance we have various load balancing algorithms, different types of load balancing strategies and techniques.

Keywords: Computational grid, resource utilization, request handling, data migration.

I. INTRODUCTION

Generally there are three type of phases related to Load balancing i.e. Information Collection, Decision Making, Data Migration. Grid computing is a type of parallel and distributed system that enables the distribution, selection and aggregation of geologically resources dynamically at run time depending on their availability, capability, performance, cost, user quality-of–self-service requirement [1]. Grid Computing should enable the job in question to be run on an idle machine elsewhere on the network [2]. Grids functionally bring together globally distributed computers and information systems for creating a universal source of computing power and information [3].

A key characteristic of Grids is that resources (e.g., CPU cycles and network capacities) are shared among various applications. Load balancing is a technique to enhance resources, utilizing parallelism, exploiting throughput improvisation, and to reduce response time through an appropriate distribution of the application [4].

II. LOAD BALANCING IN GRID ENVIRONMENT

A key characteristics of grid is that all the resources are shared among various applications and therefore, the amount of resources available to any given application highly fluctuates over times. Load balancing is the technique to enhance resources, utilizing

Parallelism, exploiting throughput improvisation, and to cut response time through the appropriate distribution of the application. To minimize the decision time is one of the objectives for load balancing which has yet not been achieved.

III. BASIC MODEL OF GRID

The basic grid model generally composed of number of hosts, each have several computational resources, which may be homogeneous or heterogeneous. The four basic building Blocks of grid model are user, resources broker, grid information system (GIS), and last but not the least resources. When user requires high speed execution, the job is submitted to the broker in grid. Broker splits the job into various tasks and distributes to several resources according to the user's requirements and availability of resources. GIS keeps the status information of all resources which help broker for scheduling.

IV. JOB AND RESOURCE MANAGEMENT ALGORITHMS IN GRID COMPUTING

Job scheduling: Job scheduling is the mapping of jobs to specific physical resources, trying to minimize some cost function specified by the user. Effective computation and job scheduling is rapidly becoming one of the main challenges in grid computing and is seen as being vital for its success.

A. Efficient Utilization of Computing Resources Using Highest Response Next Scheduling in Grid (HRN).

Description: Highest Response Next Scheduling [6] provides more responses with time, memory and CPU requirements. Here, jobs are allotted to number of processors based on job's priority and processor's capability. This scheme is adaptive for local jobs and remote jobs without any loss of performance and also highly adaptive for grid environment.

Advantages:

1) HRN with priority will effectively utilize the available resource and complete all the jobs quickly than FCFS.
2) It corrects some of the weakness of both Shortest Job First (SJF) and First Come First Serve (FCFS).
3) A priory information is not required like in shortest job first.

V. JOB AND RESOURCE MANAGEMENT ALGORITHMS IN GRID COMPUTING

Job scheduling: Job scheduling is the mapping of jobs to specific physical resources, trying to minimize some cost function specified by the user. Effective computation and job scheduling is rapidly becoming one of the main challenges in grid computing and is seen as being vital for its success.

Manuscript received on February 2013.
Mr. Sakadasariya Achyut R. Who is PG Scholar of Department of Computer Science & Technology, Gujarat Technological University, and Ahmeddbad, India.

A. Efficient Utilization of Computing Resources Using Highest Response Next Scheduling in Grid (HRN).

Description: Highest Response Next Scheduling [6] provides more responses with time, memory and CPU requirements. Here, jobs are allotted to number of processors based on job's priority and processor's capability. This scheme is adaptive for local jobs and remote jobs without any loss of performance and also highly adaptive for grid environment.

Advantages:

1) HRN with priority will effectively utilize the available resource and complete all the jobs quickly than FCFS.
2) It corrects some of the weakness of both Shortest Job First (SJF) and First Come First Serve (FCFS).

Disadvantages:

1) It is not suitable for more number of jobs allocations because finding priority of job is tedious one.

B. Node Allocation in Grid Computing Using Optimal Resource Constraint (ORC) Scheduling.

Description: The Optimal Resource Constraint algorithm [7] allocates the jobs according to processor's capability. It applies best fit algorithm followed by Round Robin (RR) scheduling which distributes the jobs among the available processors. ORC is compared with different algorithms like FCFS, SJF and RR. The comparison shows that ORC gives better performance than other algorithms in terms of turnaround time and average waiting time. It improves the efficiency of load balancing and dynamicity capability of the grid resources.

Advantages:

1) It overcomes the problem of FCFS and HRN scheduling policy as it is suitable for more number of jobs.
2) It helps to minimize the complexity of process allocation, reduces the turnaround time and average waiting time of jobs in the queue.
3) It avoids starvation problem.

Disadvantage:

1) High communication overhead.

C. Hierarchical Job Scheduling for Clusters of Workstations (HJS).

Description: The scheduling model is based on a hierarchical approach [8] using two level scheduling consisting of top level (global scheduling) and local level. The global scheduler uses single or separate queue for

Different type of the jobs for scheduling with the FCFS, SJF or First Fit (FF) policy. The local scheduler uses only one queue for all types of jobs with any one policy FCFS, SJF or FF. The global scheduler has a number of functions. One of these is matching of the resources requested by a job to those available in the participating clusters. Another is to obtain the best utilization of the available clusters. The local scheduler is responsible for scheduling jobs to a specific resource. At both levels, the schedulers strive to maintain a good load balance.

Advantages:

1) It tries to reduce overall turnaround time and maximize system utilization for high system loads.
2) For high system loads it uses multi queue to maintain the delay of job scheduling at global level.

Disadvantages:

1) SJF can result in extreme delays for long running jobs and also it is strongly biased against large jobs, so there may be starvation problems.
2) There may be a chance of underutilization of grid resources.
3) This algorithm does not consider the dynamic behavior of the grid resources.

D. Resource Co-allocation for Scheduling Tasks with Dependencies in grid (RCSTD).

Description: The Co-allocation scheduling algorithm [9] provides a strategy for scheduling the tasks with dependencies in grid environment. The algorithm applies on both inside and across the clusters. Every step combines or merge the clusters (tasks inside the cluster or clusters across the cluster) based on the dependencies between the combined clusters. Thus these clusters are combined if any dependencies exist between current and previous cluster. The main goal of the algorithm is to improve efficiency in terms of load balancing and minimum time for the execution of the tasks.

Advantages:

1) Minimize Execution Time of the Task.
2) The algorithm has a dynamic nature because inside a cluster the tasks are allocated to the suitable resource on which it can be scheduled at the earliest time.
3) Due to the decentralized strategy that Co-allocation uses, the method is more reliable than a centralized one for being less subject to single point of failure.
4) This scheduling algorithm obtains good load balancing among all the resources of the system in terms of number of tasks scheduled on each resource.

Disadvantages:

1) More Communication overhead inside and across the clusters.
2) It has not specified the requirements of a task.

Resource Scheduling: The grid resource scheduling process can be defined as the process of matching a query for resources, described in terms of required characteristics, to a set of resources that meet the expressed requirements. To make information available to users quickly and reliably, an effective and efficient resource scheduling mechanism is crucial. [1]- [5]. Generally grid resources are potentially very large in number with various individual resources that are not centrally controlled. These resources can enter as well as leave the grid systems at any time. For these reasons resource scheduling in large-scale grids can be very challenging.

A. Research on Novel Dynamic Resource Management and job scheduling in grid computing (RNDRM).

Description: This scheduling model is based on Heap Sort Tree (HST) [13] for computing the available computational power of the nodes (resource) as well as whole grid system. Here the resource with largest available computational ability among the whole grid system is selected to be the root node of the HST and it is ready for the scheduler to submit a job. The algorithm design for job scheduling is well suitable for the complex grids environment and it is based on agents.

Advantages:

1) 1) This algorithm makes the system more scalable, robust, fault-tolerant and high performance.
2) 2) This strategy provides dynamic status information of the resources in an unpredictable fast changing grid environment.

Disadvantages:

1) 1) This algorithm is silent at the condition of job submission failure.
2) 2) The job scheduling strategy may not utilize resource sufficiently.
3) 3) Job waiting time is high.
4) 4) It does not provide real time dynamic grid environment.

B. Agent Based Resource Management with Alternate Solution (ABRMAS).

Description: Agent based Resource Management with Alternate Solution [14] gives an alternate solution at the situation when resource discovery fails. Algorithm identifies an equivalent resource without affecting the performance and it also avoids unnecessary resource discovery. Sometimes resource discovery is done for time bound task and required resource is unavailable at that situation. Alternate solution reduces delay overhead in waiting for the unavailable resource and enhances the system's efficiency. Implementation result shows the system success rate is 30% higher with alternate solution.

Advantages:

1) It limits and steer the search towards the anticipated result and provide efficient resource discovery.
2) Useful in both cases when discovery fails and more than one solution proposal offered.

Disadvantages:

1) For large agent hierarchy proposal‘s invitations may be restricted to sub hierarchy. 2) It is not explicit.

C. New Resource Mechanism with Negotiate Solution based on agent in grid environments (NRMNS).

Description: Agent Based Resource Management with Negotiate Solution gives an alternate solution [15] at the situation of resource discovery failure. Algorithm adds the middleware Grid Architecture for Computational Economy (GRACE) with Resource Pricing Fluctuation Manager (RPFM) into ABRMAS in order to improve the efficiency of the resource management scheduling allocation in Grid Computing. The feedback model plays a very important role in the agent-based system when resource discovery failed for cost bound.

Advantages:

1) The resource provider can get the maximum investment profit.
2) Feedback capability of RPFM is used to adapt the highly dynamic grid environment.
3) Simulation result shows successful rate of resource discovery increases by about 10%.

Disadvantage:

1) The resource discovery is aborted when the RPA (Resource provider agent) refuses to decrease the cost of the resource; this one is the major drawback.

D. Improved Resource discovery approach using P2P model for condor (IRP2P).

Description: IRP2P is a grid middleware. It is a decentralized technique [16] which opposes traditional client - server model. Goal of the model is to improve performance of condor middleware. Proposed hybrid model uses four axis frameworks in P2P approach. Each framework overcome some limitations of condor middleware and makes it more reliable, robust and scalable. By implementing membership protocol, network communication is easy and using overlay construction algorithm intercrosses communication is also allowed which is restricted in condor.

Advantages:

1) Independence from central global control.
2) Fast discovery of resources using DHTs and indexing concept.
3) Scalability.
4) Support for intermittent resource participation.

Disadvantages:

1) Need to have strong self-organization capabilities in order to be able to maintain their rigid structure.
2) High maintenance cost in the presence of high churn.

E. Virtual Computing Grid using Resource Pooling (VCGRP).

Description: The System is based on loosely coupled concept. Virtual Computing Grid means the system can choose a resource and allocate tasks to it. Here, it is a single point web based access known as Virtual Computing Grid Portal and the Virtual Computing Grid Monitor is a central resource manager for the System. [17].

Advantage:

1) 1) Cost Effective model.

Disadvantages:

1) 1) Not much Reliable because of only one central manager and single point web access.
2) 2) Since it is cost effective solution quality of service has been play down in the prototype model.

VI. ANALYSIS

A. Analysis and Comparisons between various papers depending upon various parameters

Job Scheduling:

HRN [6] is highly adaptive in grid environment but it is not suitable for more number of jobs in homogeneous environment. Optimal Resource Constraint (ORC) scheduling algorithm [7] overcomes the FCFS and HRN algorithm problem and reduced the turnaround time as well as waiting time of the job, but there is a high communication overhead. Hierarchal job scheduling [8] reduces overall turnaround time and maximize system utilization but, there is CPU power wastage. Resource Co-allocation for scheduling tasks with dependencies algorithm [9] minimizes execution time of the task, but with high communication overhead inside and across the clusters. Scheduling framework for bandwidth-aware job grouping-based algorithm [10] minimizes the wastage of CPU power and reduces the network latency but with a high preprocessing time for job-grouping and resource selection. It does not consider memory-size constraint and dynamic resource characteristics and the strategy do not utilize resource sufficiently. Grouping based fine grained job scheduling algorithm [11] reduces execution time, network latency and processing time but have higher time complexity. A job schedule model based algorithm [12] maximizes CPU utilization and minimizes turnaround time but with high communication overhead. Considering all these criteria and referring table I and simulation result, it is found that Grouping-based Fine-gained Job scheduling algorithm [GFJS] provides a near optimal job scheduling among all surveyed job scheduling algorithms.

Resource Scheduling:

Research on novel dynamic resource management and job scheduling in grid [13] computing makes system more scalable, robust and fault-tolerant with high load balance but time complexity is high whereas virtual resource pooling [17] fully utilizes resources with less reliability. An improved resource discovery approach using p2p model [16] for condor along with grid middleware makes condor more reliable, scalable and robust for working in heterogeneous environment. But it needs some strong self-managing organization capability. Alternate solution [14] helps during resource discovery failure which is not explicit. Negotiation solution [15] is much adaptive in grid, higher resource discovery success rate, high resource utilization and also cost bounded. Referring table I and simulation result, it is concluded that a Research on novel dynamic resource management and job scheduling [RNDRM] in grid computing is best for resource scheduling.

VII. CONCLUSION AND FUTUREWORK

In this paper, various scheduling algorithms in grid computing have been surveyed. Simulation result has shown their processing time with respect to number of jobs. A comparison on various parameters like distributed, hierarchical, centralized, response time, load balancing, resource utilization was done get feedback on different types of job and resource scheduling. The researchers can use these facts to develop better algorithms. In the above study it was found that no paper has specified memory requirement of the jobs while submitting the jobs to the selected resources. Memory requirement of a job is vital in completing the execution of jobs at the selected resources within a time bound in realizing a real grid system. Our future work will be based on the above findings to develop a more efficient algorithm for job scheduling and resource selection that will reduce the preprocessing time of jobs and considering memory constraint for resource selection.

REFERENCES

1. Ian Foster and Carl Kesselman, "The Grid: Blueprint for a New Computing Infrastructure," Elsevier Inc., Singapore, Second Edition, 2004.
2. K.Ranganathan and I.Foster, Simulation studies of computation and data scheduling algorithms for data grids. Journal of Grid Computing, 1(1), 53-62, 2003.
3. M. Caramia, S. Giordani and A. Iovanella, Grid scheduling by on-line rectangle packing. Networks, B. Smith, "An approach to graphs of linear forms (Unpublished work style)," unpublished, 44(2), 106-119, 2004.
4. Y. Gao, H. Rong and J. Huang, "Adaptive grid job scheduling with genetic algorithms. Future Generation Computer Systems", 21, 151-161, 2005.
5. C. Weng and X. Lu, "Heuristic scheduling for bag-of-tasks application in combination with QoS in the computational grid", Future Generation Computer Systems, 21, 271-280, 2005.
6. K.Somasundaram, S.Radhakrishnan, M.Gomathynayaga "Efficient Utilization of Computing Resources using Highest Response Next Scheduling in Grid" 6 (5): 544-547, Asian Journal of Information Technology, 2007.
7. K.Somasundaram, S.Radhakrishnan, "Node Allocation In Grid Computing Using Optimal Resource Constraint (ORC) Scheduling**",** VOL.8 No.6, IJCSNS International Journal of Computer Science and Network Security, June 2008.
8. J. Santoso; G.D. van Albada; B.A.A. Nazief and P.M.A. Sloot: "Hierarchical Job Scheduling for Clusters of Workstations", ASCI 2000, pp. 99-105. ASCI, Delft, June 2000.
9. Diana Moise, Izabela Moise , Florin Pop,Valentin Cristae, "Resource CoAllocation for Scheduling Tasks with Dependencies, in Grid", The Second International Workshop on High Performance in Grid Middleware HiPerGRID 2008.
10. Ng Wai Keat,Ang Tan Fong,Ling Teck chaw,Liew Chee Sun, "SCHEDULING FRAMEWORK FOR BANDHWIDTH-AWARE JOB GROUPING-BASED SCHEDULING IN GRID COMPUTING", Vol.19(2), Malaysian Journal of Computer Science, 2006.
11. Quan Liu, Yeqing Liao, "Grouping-Based Fine-grained Job Scheduling in Grid Computing", Vol.1, pp. 556-559, IEEE First International Workshop on Education Technology and Computer Science, 2009.
12. Homer Wu,Chong-Yen Lee,Wuu-Yee chen,Tsang Lee, "A Job schedule Model Based on Grid Environment", IEEE Proceeding of the First International Conference on Complex, Intelligent and Software Intensive System, CISIS'07 2007.
13. Fufang Li, Deyu Qi, Limin Zhang, Xianguang Zhang, and Zhili Zhang, "Research on Novel Dynamic Resource Management and Job Scheduling in Grid Compuing", IEEE Proceedings of the First International Multi-Symposiums on Computer and Computational Sciences, IMSCCS 2006.
14. Ms.P.Muthuchelvi, Dr.V.Ramachandran, "ABRMAS: Agent Based Resource Management with Alternate Solution," IEEE, the Sixth International Conference on Grid and Cooperative Computing, GCC 2007.
15. Junyan Wang, Yuebin Xu, Guanfeng Liu, Zhenkuan Pan, and Yongsheng Hao, "New Resource Discovery Mechanism with Negotiate Solution Based on Agent in Grid Environments", IEEE The 3rd International Conference on Grid and Pervasive Computing – Workshops, 2008.
16. Anju Sharma, and Seema Bawa, "An Improved Resource Discovery Approach Using P2P Model for Condor: A Grid Middleware", World Academy of Science, Engineering and Technology, 2006.
17. Alpana Rajan, Anil Rawat, Rajesh Kumar Verma, "Virtual Computing Grid using Resource Pooling", IEEE, International Conference on Information Technology, 2008.
18. R. Buyya and M. Murshed, GridSim; A toolkit for the modeling and Simulation of distributed management and scheduling for grid Computing, 2002. World Academy of Science, Engineering and Technology 40 2010.