COMPARISONOF DIFFERENT ALGORITHMS TO IMPROVE THE PERFORMANCE OF GRID COMPUTING

Dr (Prof) Ajay Rana [1], Himanshu Dua[2]

[1] *Professor, Amity University (India)*

ABSTRACT

*In an assorted environment like Grid, in which we are allowedto use geographically widely distributed resources in order to sort large-level of applications.In order to maintainthe balance of load on working between infras like that of grid, using load balancing algorithms was important. Manyjobs are needed to create equally balance between multiple computing nodes. In order to completely utilize the resource in assorted network and growing computation and to improve the overall throughput of system aimsat load balancing algorithm. The importance of this paper lies in reviewing different load balancing algorithms for the assorted network like grid and in order to identify various basicsand gaps among them. Many of load balancing algorithms have already been implemented that works to solve various issues like scalability, heterogeneity, etc. Different lo*A comparison has been shown in the following table for different load balancing algorithms based on various metric such as communication overhead means to message traffic while communicating , load balancing time, scalability, heterogeneity etc;

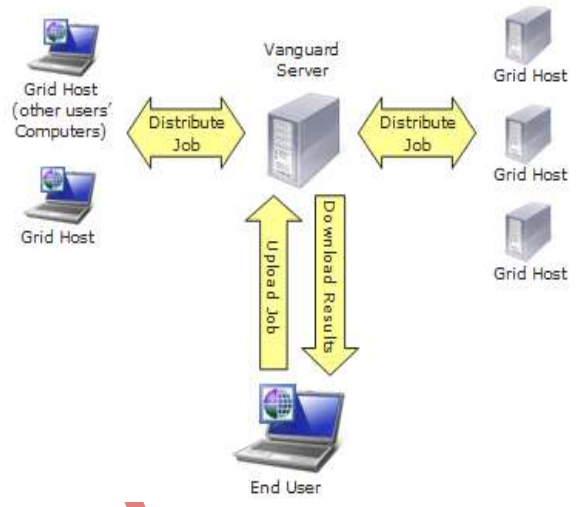
ad balancing algorithms woks on various metrics such as make span, average resource utilization rate, time, communication overhead, stability, fault tolerance, reliability.

Keywords: ***Grid Computing, Distributed Computing, Load Balancing, Resource Management.***

I INTRODUCTION

Grid computing is evolving as a high scale distributed infrastructure thatmaintains large scale sharing of resources and synchronizing the problem solving in dynamic and assorted network. Many resources are connected internally and are working independently by cooperating with each other. Workload represents the amount of work that needs to be performed where all resources have different speed of processing different tasks. A grid environment can offer results of resource balancing just by scheduling grid jobs properly [1]. An improved scheduling and efficient load balancing algorithm may improve the overall system performance with reduced execution time. Load balancing is required to equally distribute the tasks across different resources in order to increase the computation and minimum timefor executingtask. In a grid some nodes may be loaded with high data while others may be idle or stay under loaded. So a good load balancing algorithm is about to secure from the condition where certain resources are overburdened with data and other are not fully utilized. Grid environment involves various issues [2] such as follows.

- **Heterogeneity:** Heterogeneity refers to the use of various technologies and managerial policies that exists in both of computational and network resources.
- **Autonomy:** Autonomy refers to autonomous because the organizations that share Grid resources, a site are considered as an autonomous computational entity.
- **Scalability:** Problems that are involved when a grid grows from few of resources to multi-millions.
- **Dynamicity:** Resource failure is possible it occur due to certain hardware of software problems or connection disturbances. So as to adopt a heterogeneous behavior to deal with such circumstances is necessary.
- **Resource balancing:** Balancing the workload on millions of resources itself a challenge. Proper distribution and proper migration policies needs to be implemented.
- **Reliability and Management:** To keep the data's reliability form there are other issues involved that are to be handled.



A load balancing algorithm is used to fully utilizing the unused resources, and has the possibility of majorly increasing the Efficiency of resources usability, in order to enhance the performance and speed of system with no time wastage [3]. They are also important for the purpose of fulfilling the periodic computational needs and sharing of computational results.

II RELATED WORK

Many load balancing algorithms have been proposed in this field. It is hard to achieve load balancing in grid systems as compared to that in traditional distributed computing environment dueto various issues and its dynamic nature. Many ways presented rely on centralized structure. All of them suffered from significant deficiencies, such as scalability issues. Static and dynamic load balancing techniques are mostly used for allocation of tasks in grid environment [6]. Several works have been done on dynamic load balancing approach. A load balancing model on tree re-

presentation of a grid is proposed. In [5] Yagoubi, et al presented a hierarchical load balancing strategy that uses a task-level load balancing**.** In [6], Menno, et al calculated the effectiveness of dynamic load balancing by means of trace-driven simulations.

They also proposed a solution to users of parallel application and distributed environment that weather to use DLB or JR. Agent-based approaches have tried to provide load balancing in cluster of machines [7]. Junwei Cao, et al concerned on load balancing while developing parallel and distributed computing applications. & when the problems related to cross-domain and large-scale scheduling comes then emergence in computational grid continues the problem. So In this author proposed working with an agent-based grid management infrastructure which is joined with a performance-driven task scheduler which has been developed for local grid load balancing. In [8], Shah et al represented two job migration algorithms, which are MELISA (Modified ELISA) and LBA (Load Balancing on Arrival).

These approaches differ in the way load balancing is carried out and has proved its efficiency in reducing the response time to minimum on large and small-scale heterogeneous Grid environments, authors proposed a decentralized grid model and then introduced a dynamic load balancing algorithm (DLBA) which performs intra-cluster and inter-cluster (grid) load balancing. Yajun Li et al in presented an hybrid strategy [9] for load balancing in grid environment which takes the use of two approaches average based, instantaneous approach by combining them. A new decentralized algorithm [10] proposed algorithm at Meta scheduler and cluster or resource level. Jasma et al proposed a fault optimal load balancing algorithm [11] by understandingthe challenges in grid environment. In order to ensure the reliability in distributed grid environment, fault tolerance needs be high. In a grid like environment where multiple computing nodes are connected to each other, reliability of every resource cannot be guaranteed. Hence, it's necessary in order to eliminate the probability of failure in grid computing. Main aim is to prevent it from condition where some processors are overloaded with a set of tasks while others are underloaded.

III LOAD BALANCING APPROACHES

a. Static load balancing*:* In static the number of processors is static, it is assumed that certain priori information exist, but if any change occur in problem size, the fixed number of processors may not be sufficient and in certain environments all the processors cannot be used all the time. So it is necessary to have some strategy which deals with such circumstances and overcome this problem. Round-robin, simulated annealing, randomized are some of techniques for static load balancing. It leads to use of dynamic load balancing [1].

b. Dynamic load balancing: It make changes to the distribution of work among computing nodes at run-time. It makesthe use of current and recent load information when making distribution decisions and they continually monitor the load on all the processors and when the load imbalance reach some predefined level, the redistribution of work is done [2].

The Static approach is more used because of it being simple and minimized runtime overhead. However it has dis-advantage that assumes the characteristics of the computing resources and communication network are all known in advance and will be constant. Such assumptions cannot be applied to grid environment.

c. Parameters-There are majorly three important parameters that determine into which load- balancing strategy will be used

a) Who takes the decision for load balancing?

b) What type of information is required for making the load balancing decision?

c) Where the decision about load balancing is made?

IV STRATEGIES FOR LOAD BALANCING

4.1 Centralized & Decentralized Strategy

a. Centralized Strategy-In centralized approach [6], only 1 node in the distributed system acts as the central controller. This main node has global view on the load information of all nodes connected to it, and decides how to allot jobs to each and every node. And rest of the nodes act as slaves.

b. Decentralized-All nodes available in the distributed system are involved in making the load balancing decision. It is commonly agreed that distributed algorithms are more scalable and have better fault tolerance

4.2 Sender-Initiated & Receiver-Initiated Strategies

a. Sender-initiated: In sender initiated strategy, Congested nodes attempt to transfer work towards under-loaded nodes. Sender-initiated policy works well than the receiver-initiated strategy at low or moderate system loads. The major reasons for the same is that, the probability of finding a lightly-loaded node is higher than that of finding a heavily-loaded node.

b. Receiver-initiated-In this type, Less-loaded nodes look for heavily-loaded nodes from which work may be received similarly, at high system loads; the receiver initiated policy works better.

4.3 Global & Local Strategies

a. Global Strategy:-The load balancer uses the performance enhancer of all available nodes. Global or local both replies the question on which information will be used to make a load balancing decision in globaldecisions. For global schemes, balance load speed is faster when compared to that of a local scheme since all workstations are considered at the same position of time.

b. Local Strategy-.In local scheme workstations are divided into various groups. The benefit of a local scheme is that performance profile information is only exchanged within the group..

4.4 Co-operative & Non-co- operative

a. Co-operative strategy-It is one in which load of one is shared by other node, In other words nodes co-operate with each other. & on the other side, if they don't reflects **non co-operative** strategy behavior. It takes their decision own to balance load. These are the main strategies used in load balancing mechanism.

V POLICIES OF LOAD BALANCING

A good load balancing algorithm is defined by certain basic policies which are Transfer policy, Selection policy, Location policy and Information policy, resources type policy, triggering policy [1].

a. Information policy: specifies what amount of workload information has to be collectedand updated, When it has to be collected and from where.

b. Triggering policy: determines the appropriate period of time to start a load balancing operation.

c. Resource type policy: classifies a resource as server or receiver of tasks according to its availability status. Whether resource is available to retrieve the task from most overloaded resource.

d. Location policy: uses the results of the resource policy to find a suitable partner for a server or receiver

e. Selection policy: defines the tasks that should be migrated from busiest resources (source) to most available resources (receiver). A selection policy considers several factors in selecting a task, for e.g. transfer of small task will take less overhead.

On the basis of Location policy, the dynamic load balancing algorithms can be further classified into sender initiated algorithms, receiver initiated algorithms and symmetrically-initiated algorithms

VI LOAD BALANCING STEPS

There are some main steps that almost all load balancing algorithms have in common [4].

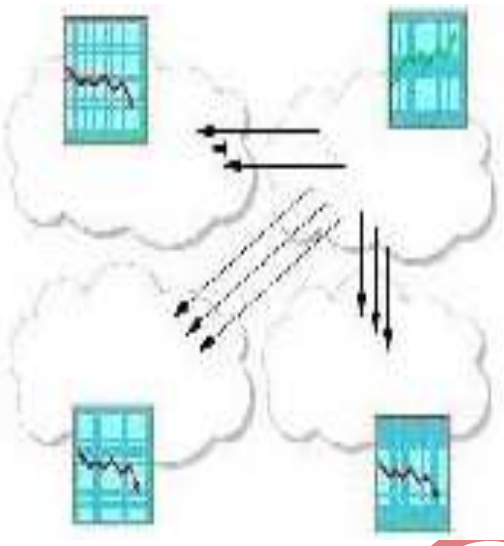
a. Monitoring -Monitoring resource load state.

b. Synchronization-Synchronizing load and state information between resources.

c. Rebalancing Criteria-Calculating the new work distribution and making work moment decision.

d. Job Migration-Its actual movement of data.

It provides when a system decides to export a process. It decides whether to create it on local site or create it on a remote processing site.



VII VARIOUS ISSUES

Dynamic load balancing may consider various issues mentioned here, however it needs to collect and maintain information about all the available nodes [1].

a. Process transfer issue: It takes care to determining whether to execute a process locally or remotely or globally.

b. State information exchange issue: It determines how to get exchange the collected load information among various availablenodes.

c. Load estimation issue: This policy specify the issue regarding to estimate the workload of a particular node of the system.

d. Migration issue: Main job of this policy is to migrate the load from one state to another. It determines total number of times a process is migrating.

VIII COMPARISON OF DIIFERENT LOAD BALANCING ALGORITHM IN GRID BASED ON VARIOUS METRIC/ ISSUES

A comparison has been shown in the following table for different load balancing algorithms based on various metric such as communication overhead means to message traffic while communicating , load balancing time, scalability, heterogeneity etc;

Algorithm/Metric	Agent based approaches for load balancing	Fault Tolerant optional neighbor load balancing	Dynamic load balancing algo-rithm in grids	Decentralized load balancing algorithm in grids
Communication Over-head	More	More	Less	More
Make span	Less	Average	More	Less
Load balancing time	Less	Less	More	Less
Scalability	Scalable	Scalable	Scalable	Scalable
Avg. Resources Utiliza-tion rate	Average	Improved	More	More
Fault Tolerance	Integrated	Very high	Integrated	Integrated
Reliability	Integrated	High	Integrated	Integrated

IX CONCLUSION

This paper presents a comparative survey of load balancing algorithms in grid environment. The accepted techniques of load balancing in grid environment with their importance, combinations and variations have been discussed. Grid application performance remains a challenge in dynamic grid environment. Resources submitted to Grid and can be withdrawn from Grid at any moment. Main objective of load balancing algorithm is to achieve high performance in grid environment by optimal usage of geographically distributed and heterogeneous resources. So such an algorithm which efficiently manage and balance the workload also according to working capacity of processor and minimized the execution time and increase the global throughput of system, is required in such an unpredictable environment of grid. However, accepting the importance of all the aforesaid areas, to put forward a future direction of work, this research would next focus on finding optimal approach for better performance of applications running in grid.

REFERENCES

[1]. LalithaHima,Venkatesan,Karunya university, coimbatore, India “Perspective Study on Resource level Load balancing in Grid Computing Environments” 2011 IEEE.

[2]. Miguel L. Bote-Lorenzo, Yannis A. Dimitriadis,”Grid Characteristics and its usage Springer-Verlag LNCS 2970,Spain, 2009.

[3]. M. Baker, R. Buyya, and D. Laforenza, “Grids and Grid Technologies for Wide Distributed Computing”, International Journal of Software: Practice and Experience (SPE), Vol. 32, no. 2010.

[4]. NeerajRathore,Chana “A Cognitive Analysis of Load Balancing and Job Migration Techniques”World Congress on Information and Communication Technologies, January IEEE 2011.

[5]. B. Yagoubi, and M. Meddeber, "A load balancing model for grid environment", Proceeding of 22nd International Symposium on Computer and Information Sciences (ISCISC 2007), pp. 1-7, 7 November 2007.

[6]. M. Dobber, R. Mei, and G. Koole, "Dynamic Load Balancing and Job Replication in a Global-Scale Grid Environment: A Comparison", IEEE Transaction on Parallel and Distributed Systems, Vol. 20, no. 2, pp. 207-218, February 2009.

[7]. Junwei Cao "agent based using performance driven task scheduling" C&C Research Laboratories, NEC Europe Ltd., Sankt Augustine, Germany IEEE.

[8]. Shah, Veeravalli, "On the Design of Adaptive and Decentralized Load-Balancing Algorithms with Load Estimation for Computational Grid Environments" IEEE Transactions On Parallel And Distributed Systems, 2007.

[9]. "A Hybrid Load balancing Strategy of Sequential Tasks for Computational Grids" Yajun Li, Yuhang Yang Rongbo Zhu Electronic Engineering Department Shanghai Jiaotong University, China 2009.

[10]. RajkumarRajavel, Kongu Engineering College, Perundurai, Erode, India. December,2010 "Decentralized Load Balancing for the Computational Grid Environment".

[11]. Jasma, NedunchezhianRaju Department of Computer Science and Engineering, Atria Institute of Technology, "A Fault Tolerance Optimal Neighbour Load Balancing Algorithm for Grid Environment",2010.