INTERNATIONAL JOURNAL FOR ADVANCE RESEARCH IN ENGINEERING AND TECHNOLOGY
WINGS TO YOUR THOUGHTS…..

To Enhance Performance of Grid Computing using Scheduling Algorithms

Mamta Sharma[1], Manish mahajan[2]

[1]M.Tech, Chandigarh Engineering College, Landran
sharma.mamta0801@gmail.com
[2]Chandigarh Engineering College, Landran
cec.manish@gmail.com

Abstract: *Grid computing is a computer network in which each computer resources are shared with every other computer in the system. The processing power, memory and data storage are all community resources. These resources are authorized users that can tap into and leverage for specific tasks. A grid computing system can be as simple or complex. It can be as simple as that the similar computers running on the same operating system or as complex as inter networked systems comprised of every computer platform. Grid computing systems work on the principle of pooled resources. The use of computational grids is to provide an integrated computer platform. It composed of differentiated and distributed systems and presents fundamental resource. Key services such as resource discovery, monitoring and scheduling are inherently more complicated in a grid environment where the resource pool is large, dynamic and architecturally diverse. In the grid computing many problems like Authentication and authorization, secure and reliable file transfer, distributed storage management and resource scheduling across organizational boundaries are need to be solved. It can be solve by the use of scheduling techniques. In this paper, we discuss about the different scheduling techniques.*

Keywords: *Grid Computing, scheduling, resources, performance.*

1. INTRODUCTION

Grid computing has progressed towards a service oriented paradigm. It is the new way of the service provisioning based on utility computing models. The scientist has started to adapt to the grid computing techniques for the past many years. This has increased the computing power and capability of inter organization, national and international grid computing infrastructures.[1] The goal of the grid computing is to integrate the clusters into global infrastructures. Grid computing is driven by five big areas:

Resource sharing: Global sharing is the very essence of grid computing. It has the crucial image in the society.

Secure access: Trust between resource providers and user is essential, especially when they don't know each other. Sharing resources conflicts with security policies in many individual computer centers, and on individual PCs, so getting grid security right is crucial.

Resource use: Efficient, balanced use of computing resources is essential for the human development.

The death of distance: Distance should make no difference**:** you should be able to access to computer resources from where ever you are.

Open standards: Interoperability between different grids is a big goal, and is driven forward by the adoption of open standards for grid development, making it possible for everyone can contribute constructively to grid development. Standardization also encourages industry to invest in developing commercial grid services and there infrastructure.

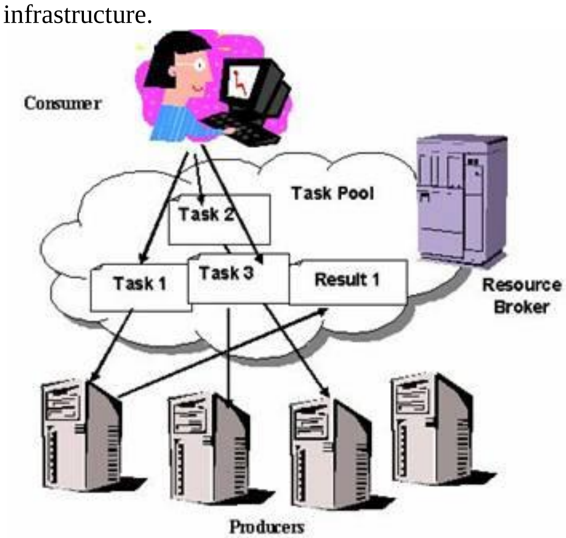

Figure 1: basics of grid computing

INTERNATIONAL JOURNAL FOR ADVANCE RESEARCH IN ENGINEERING AND TECHNOLOGY
WINGS TO YOUR THOUGHTS…..

In the figure 1, there is a consumer, and a work is divided into the several tasks, and it sends it to the resource breaker to break it and finally it sends to the producers.

Architecture of grid Computing:

A grid's architecture is often described in terms of layers, where each layer has a specific function. The higher layers are generally user-centric, whereas lower layers are more hardware centric, focused on computers and networks.

- The lowest layer is the network, which connects grid resources.
- Above the network layer lies the **r**esource layer: actual grid resources, such as computers, storage systems, electronic data catalogues, sensors and telescopes that are connected to the network.
- The middleware layer provides the tools that enable the various elements to participate in a grid. The middleware layer is sometimes the "brains" behind a computing grid.
- The highest layer of the structure is the application layer, which includes applications in science, engineering, business, finance and more, as well as portals and development toolkits to support the applications. This is the layer that grid users see and interact with. The application layer often includes the service ware, which performs general management functions like tracking who is providing grid resources and who is using them.

The grid users need not be aware of the computational resources that are used for executing their jobs and storing their data. Grid computing is an emerging technology. The next generation of parallel distributed computing platform uses the grid computing for solving large scale computational and data intensive problems in science, engineering and commerce [3]. In the grid computing many problems like Authentication and authorization, secure and reliable file transfer, distributed storage management and resource scheduling across organizational boundaries are need to be solved. The Grid users need not be aware about the computational resources that are used for executing their applications and storing their data. The resource allocation to a large number of jobs is hard and much more difficult than the LAN computational environments [2].

The load balancing of the available resources in the computational grid is another important factor. The application adaptive resource management and scheduling are technical challenges in the Grid. Now days, grid computing is used in many applications. These applications are beyond to the distribution and sharing of resources. The distributed resources are useful only if the Grid resources are scheduled. A scheduling is a process that maps and manages the execution of interdependent tasks on the distributed resources. It allocates suitable resources to workflow tasks so that the execution can be completed to satisfy objective functions purposed by users. Proper scheduling can have significant impact on the performance of the system.

There are basic two types of scheduler are present in the grid computing.

- Optimal scheduler
- Poor scheduler

In case of the optimal scheduler, the performance of the grid computing is high, whereas in case of poor scheduler the performance is vice-versa. The grid scheduling is a big topic in grid environment for new algorithm model. The scheduling in Grid environment has to satisfy a number of constraints on different problems.

There are many techniques used for the grid computing. The existing technique for scheduling in grid applications uses queuing systems or ad-hoc schedulers. These schedulers use the specific knowledge of the underlying grid infrastructure to achieve an efficient resource allocation.

These approaches cannot deal with the complexity of the problem due to dynamic nature of the grid. Grid scheduler is also known as the resource broker, because it acts as an interface between the user and distributed resources. It hides the complexities of the computational grid from the user. The scheduler does not have full control over the grid and it cannot assume that it has a global view of the grid. The single most challenging issue of the grid scheduler encounters is the dynamicity of resources. The grid scheduling consists of three stages [4].

- Resource discovery and filtering
- Resource selection and scheduling
- Job submission

The high performance computing and high throughput computing are two main goals of grid scheduling algorithm. The main goal of the high performance computing is to minimize the execution time for an application. In Grid computing, the scheduling must be made in the shortest time possible, because there are many users competing for resources, and time slots desired by one user could be taken up by another user at any moment. Many heuristics and meta heuristics based algorithms have been proposed to schedule workflow applications.

INTERNATIONAL JOURNAL FOR ADVANCE RESEARCH IN ENGINEERING AND TECHNOLOGY
WINGS TO YOUR THOUGHTS…..

1.1 Dynamic Scheduling Techniques

Grid computing is dynamic in nature. So in the Grid computing, the utilization and availability of resources varies over time and a better resource can join at any time. Constructing a schedule for entire workflow before the execution may result in a poor schedule. If a resource is allocated to each task at the beginning of workflow execution, the execution environment may be different at the time of task execution. At this time a best resource may acts as a worst resource. Therefore, the scheduler must be able to adapt the resource dynamics and update the schedule using up-to-date system information. Several approaches have been proposed to address these problems. In the dynamic scheduling, for individual task and batch mode based scheduling, it is easy for the scheduler to use the most up-to-date information, since it takes into account only the current task or a group of independent tasks. The scheduler could map tasks only after their parent tasks become to be executed. For dependency mode and meta hueristics based scheduling, the scheduling decision is based on the entire workflow. Here, it is very difficult to estimate execution performance accurately, since the execution environment may change for the tasks which are late executed.

2. LITERATURE REVIEW

Stefka Fidanova, et.al discuss about the ant colony optimization technique used in the grid computing. Grid computing is a form of distributed computing that involves coordinating and sharing computing, application, data storage or network resources across dynamic and geographically dispersed organizations. The goal of grid task scheduling is to achieve high system throughput and to match the application needed with the available computing resources. This is matching of resources in a non-deterministically shared heterogeneous environment. The complexity of scheduling problem increases with the size of the grid and becomes highly difficult to solve effectively. To obtain good methods to solve this problem a new area of research is implemented. This area is based on developed heuristic techniques that provide an optimal or near optimal solution for large grids. In this paper, author introduces a tasks scheduling algorithm for grid computing. The algorithm is based on Ant Colony Optimization (ACO) which is a Monte Carlo method. The paper shows how to search for the best tasks scheduling for grid computing.

Leyli Mohammad Khanli, et.al discuss about the resource matching problem in grid computing. The grid infrastructure provides a way to execute applications over autonomous, distributed and heterogeneous nodes by secure resource sharing among individuals and institutions. Typically, a user can submit jobs to a grid without necessarily knowing where it will be executed. The grid resource management system used to distribute such jobs among a heterogeneous pool of servers. [5]It tries to optimize the resource usage. It provides the best possible quality of service.
Problem is an important task in the Grid environment which involves assigning resources to tasks in order to satisfy task requirements and resource policies. This contribution presents algorithms, methods, and software for a Grid resource manager, responsible for resource brokering and scheduling in Grids. The broker selects computing resources based on actual job requirements and a number of criteria identifying the available resources, with the aim to minimize the turnaround time for the individual application**.**

Ryan J. Wisnesky, discuss about the heterogeneous computation of grid computing. Now days, computational grids are becoming more prevalent as the cost of bringing together disparate computing resources declines. However, a number of challenges remain before these grids can be utilized efficiently. This paper explores the results of using several well known scheduling algorithms to schedule work on a grid under probabilistic work arrival rates and varying task completion times. This paper presents the results of a simulation study of a heterogeneous computational grid using different scheduling algorithms. After a definition of robustness based on the concept of work completion latency is discussed, a method to simulate grids based on estimated time to compute matrices is presented. Three well known scheduling algorithms are then evaluated against each other, and the highest-performing scheduler is then analyzed in detail. The notion of ETC perturbation is presented, and this high-performing scheduling algorithm is found to be relatively robust against uncertainties in estimated task completion times.

Fangpeng Dong and Selim G. Akl discuss about the grid computing. As popularity of the Internet and the availability of powerful computers and the high speed networks as low-cost commodity components are changing the way we use computers today. These technical opportunities have led to the possibility of using geographically distributed resources to solve large-scale problems in science, engineering, and commerce. Recent research on these topics has led to

INTERNATIONAL JOURNAL FOR ADVANCE RESEARCH IN ENGINEERING AND TECHNOLOGY
WINGS TO YOUR THOUGHTS…..

the emergence of a new paradigm known as Grid computing. Grid computing is used to aggregate the power of widely distributed resources, and provide non-trivial services to users. To achieve this goal, an efficient [6] Grid scheduling system is an essential part of the Grid. In this paper author discuss the lots of things. First, the architecture of components involved in scheduling is briefly introduced to provide an intuitive image of the Grid scheduling process. Then various Grid scheduling algorithms are discussed from different points of view, such as static vs. dynamic policies, objective functions, applications models, adaptation, QoS constraints, and strategies dealing with dynamic behavior of resources, and so on.

3. PURPOSED WORK

Grid computing is dynamic in nature. In the grid computing the load is divides into the several parts. For these divisible loads optimal scheduling is used. The linear theory formulation opens a attractive modeling possibilities for systems incorporating communication and computation issues, as in parallel, distributed, and grid computing. It is defined in the context of a specific scheduling policy and interconnection topology. The linear model formulation produces the optimal solutions through linear equation solutions. In the simpler models, the recursive algebra also produces optimal solutions. The model can take into account heterogeneous processor and link speeds.

A generic grid computing system infrastructure considered here comprises a network of supercomputers and/or a cluster of computers connected by local area networks having different computational and communication capabilities. We consider the problem of scheduling large-volume loads within a cluster system, which is part of a grid infrastructure. We envisage this cluster system as a cluster node comprising a set of computing nodes. Communication is assumed to be predominant between such cluster nodes and is assumed to be negligible within a cluster node. The underlying computing system within a cluster can be modeled as a fully connected bipartite graph comprising sources. The computationally intensive loads to be processed and computing elements, called sinks, for processing loads. This represents the fact that each source can schedule its load on all the sinks. In real-life situations, one of the practical constraints is satisfying the deadline requirements of the loads to be processed while taking into account the availability of the buffer resources at the sink nodes, since the memory available at the processing nodes to store the received load and process them is limited. We consider these combined influences in our proposed algorithm.

4. CONCLUSION AND FUTURE WORK

In this paper, we discussed some scheduling structures that typically occur in meta systems or computational grids. As in real-life situations, we have considered the dynamic arrival of loads, the buffer capacity constraints at the sink nodes, and the deadline requirement of the loads to be processed. As evaluating such structures highly depends on the used algorithms and strategies of the scheduling itself, a selection of them has been presented. Future work will extend the studies to more architecture and include more detailed parameter and configuration variation. This is important as the current results show that the performance of the examined algorithms for the scheduling structure is highly dependent on the parameters, machine configurations and workload.

REFERENCES

[1] European grid forum, http://www.egrid.org

[2] Khanli, L. M. & Analoui, M. (2006a). “Grid-JQA - a new architecture for QoS-guaranteed grid computing system”, Proc. of the 14th Euromicro Conference on Parallel, Distributed and Network-based processing, France, PDP2006, Feb 15-17.

[3] Wang, C.-M.; Chen, H.-M.; Hsu, C.-C. & Lee J. (2010). Dynamic resource selection heuristics for a non-reserved bidding-based Grid environment, Future Generation Computer Systems 26, 183_197.

[4] "New to grid computing." IBM developer Works. Retrieved March 13, 2008. http://www.ibm.com/developerworks/grid/newto/

[5] Leyli Mohammad Khanli et.al, Grid-JQA: A QoS Guided Scheduling Algorithm for Grid Computing.

[6] Foster, I.; Kesselman, C.; et al. (1999). A Distributed resource management architecture that supports advance reservation and co-allocation. In proceedings of the international workshop on Quality of Service, page 27-36.