"Improving Task Division Assignment Using Heuristic Algorithms"

Akshay Neharkar, Avinash Kadam, Priya Dhage, Prof. Asmita Mali

Akshay Neharkar, Information Technology, DYPIT, Maharashtra, India

Avinash Kadam, Information Technology, DYPIT, Maharashtra, India

Priya Dhage, Information Technology, DYPIT, Maharashtra, India

Prof. Asmita Mali, Information Technology, DYPIT, Maharashtra, India

ABSTRACT

Scheduling task is one of the core steps to efficiently exploit the capabilities of heterogeneous distributed computing systems. Therefore using heuristic algorithms is a suitable approach in order to cope with its difficulty.

Existing system make use of three heuristics such as minmin, maxmin and sufferage in assigning independent tasks to processors in heterogeneous computing systems. All three heuristics are known to run by assigning N tasks to K processors. Since this three heuristics in the system are further combine, obtaining two hybrid algorithms. The motivation behind the former hybrid algorithm is to address the drawback of MaxMin in solving problem instances with highly skewed cost distributions while also improving the running time performance of MaxMin. Furthermore the hybrid algorithm improves the running time performance of Sufferage without degrading its solution quality. Objectives of heuristic algorithms are increase in system throughput, efficiency and reduction in task completion time.

But there is no need of checking time to complete the task to a processor. Instead of that in this proposed system a new algorithm is developed which is totally based on only load factor. Load on processor means checking how many other tasks the processor is currently executing. This will reduces the time required to complete the task without affecting on the throughput and efficiency.

Keywords: - *Parallel processors, heterogeneous systems, load balancing, independent task assignment, MinMin, MaxMin, Sufferage, constructive heuristics.*

1. INTRODUCTION

1.1 Project Idea

In this paper, we are going to design two basic scheduling algorithms Min-Min and Max-Min which are based on the scheduling algorithms, to use the advantages of these algorithms also, overcome the disadvantages of these algorithms.

1.2 Motivation of the Project

Today, plethora in supercomputer costs and the need for large-scale computational resources on the different way, has led to the development of network of computational resources to remove large-scale problems in different fields as like science, engineering and commerce, etc. Most of the efforts have been made and many projects have been working on it, such as Globus and Condor to provide the needed concepts and tools for tackling shortcomings. The final result of these efforts has led to the emergence of a coming paradigm known as Grid.

1.3 Literature Survey

Most of the scheduling algorithms have been designed for the grid environments, to solve the problem of mapping a set of tasks to a set of machines (scheduling). It has been proved that optimal-solving of this mapping is an NP problem. Many heuristics have been proposed to obtain semi-optimal match. Existing scheduling heuristics can be divided into two categories: on-line mode and batch-mode. In the first mode, a task is mapped to a machine as soon as it arrives at the scheduler. Some heuristic instances of this category are as follow.

MCT (Minimum Completion Time): MCT assigns every different task to the resource that performs it in the minimum amount of completion time, there might be this resource is available or not at that time. This heuristic might cause some load imbalance across the resources of the different machines. However, this is one of the heuristics that is implemented in SmartNet. The time required to map a given task to an expected resource is O(m).

OLB (opportunistic load balancing): OLB assigns each task to the resource that becomes ready next, without considering the execution time of the task on that resource. When more than one resource becomes ready, one resource is arbitrarily chosen. The time complexity of OLB is dependent on the implementation. In the implementation considered in, it takes O(m) time to find the assignment.

In the batch-mode heuristics, the different tasks are collected into a set called meta-task (MT). These sets are mapped at prescheduled times called mapping events. Some instances of this category are as follows:

Min-Min: Min-Min begins with the set MT of all tasks which are not assigned. Firstly it computes minimum completion time all tasks in MT on all resources (lines 1-3). After that two main phases of this algorithm are begins. In the very first phase, the set of minimum expected execution time for each task in MT is found (lines 5-6). In the second phase, the task with the overall minimum expected completion time. Corresponding resource (lines 7-8). Then this task is removed from MT and the process is repeated until all tasks in the MT are mapped (lines 9-11). It is also one of the scheduling algorithms implemented in Max-Min: Max-Min is very similar to Min-Min, except in phase 2. Max-Min assigns task with maximum expected completion time to the corresponding resource.

Suffrage: It is based on the idea that a task should be assigned to a certain resource and if it does not go to that resource, the most it will suffer.

2. PROBLEM DEFINITION AND SCOPE

2.1 Problem Statement

Potential of Job Processing depends on my issues such as security of resources, heterogeneity of resources, fault tolerance & resource discovery and job scheduling. Scheduling is most important to efficiently exploit the capabilities of heterogeneous or the systems having different configurations distributed computing resources and is an NP-complete problem.

2.2 Goals and Objectives

To optimized the performance of resources i.e. makespan time & resource utilization. With this, we have classified various tasks scheduling heuristic in grid on the basis of their characteristics.

2.3 Scope of Statement

Task scheduling

Resource utilization

2.4 Software Context

Scheduling the tasks to the resources in grid computing is also complicated due to the distributed, dynamic and heterogeneous nature of the resources. Scheduling is one of the basic steps to efficiently abuse the capabilities of the different heterogeneous distributed resources. A scheduling of the task is a process that maps and manages the execution of inter-dependent tasks or independent task on the distributed resources. It allocates

suitable resources to various tasks to satisfy the agreement between resource producer & resource consumer. Poor scheduling will reduce performance of the nodes participating in grid system. Due to heterogeneous, dynamic nature of resources, the problem of mapping tasks on distributed services belongs to a class of problems known as NP-hard problem. Due to dynamic & heterogeneous nature of resources, neither algorithm can produce optimal schedule. We can only make our best to find the most suitable solution for users.

2.5 Major Constraints

Resource identification and the evaluation of different resources.

2.6 Methodologies of Problem Solving and Efficiency Issues

All discussed heuristics algorithms in previous section are known for reducing makspan time based on set of different inputs. But the utilization of different resources is not up to the mark. Instead of that proposed algorithms reduces makspan and optimizes the different resources utilization & balance the load on all participating nodes.

2.7 Outcome

Load balancing,

Independent task assignment.

2.8 Hardware Resources Required

Table-1: Hardware Required

Sr. No.	Parameter	Minimum Requirement	Justification
1	CPU Speed	I3 or Higher	Software required
2	RAM	3GB	Software Required

2.9 Software Resources Required

Platform:

1. Operating System: Windows or Linux
2. IDE: Eclipse
3. Programming Language: JAVA

3. SOFTWARE REQUIREMENT SPECIFICATION

Today, plethora in supercomputer costs and the need for large-scale computational resources on the different way, has led to the development of network of computational resources to remove large-scale problems in different fields as like science, engineering and commerce, etc. Most of the efforts have been made and many projects have been working on it, such as Globus and Condor to provide the needed concepts and tools for tackling shortcomings. The final result of these efforts has led to the emergence of a coming paradigm known as Grid. According to this algorithm, Grid is a type of parallel and distributed system that enables the sharing, selection and aggregation of geographically distributed autonomous and heterogeneous resources dynamically at runtime depending on their availability, capability, and performance also the cost and users quality of service (QoS) requirements. In this paper, we are going to design a scheduling algorithm which is based on two basic scheduling algorithms which are Min-Min and Max-Min algorithms, to use their advantages and also, to overcome their disadvantages.

3.1 Purpose

Basically the main aim of mapping in both the above cases is to maximize an objective function, which is totally based on quality of service (QoS) attributes such as execution or completion time, response time or those requested by the users of the highly configured systems. To achieve such objective, different heuristics

have been developed for mapping the different tasks. In the following section, various mapping heuristics will be discussed and the different assumptions will make while describing the heuristics.

3.2 Overview of Responsibilities of Developer

To understand exact problem definition, gather requirements of the project. Also analyse the requirements and design model. Also the most important responsibility is to do efficient coding with use of appropriate data structures and different algorithms. Also the test project for set of documents is also important to complete the project successfully and finally scale it depending on the time.

3.3 Data Description

In this section, we are going to discuss five heuristics for scheduling the different tasks to various heterogeneous machines which are basically participating in grid algorithm. After that we will propose an efficient heuristics called as Optimized Assignment of Independent Task.

Algorithm Opportunistic Load Balancing: OLB assigns tasks to the next available machine in grid. If more than one different heterogeneous machine is available, any one machine out of these is chosen randomly. It does not consider expected execution time or completion time of the task on that machine.

Fast Greedy or Minimum Completion Time (MCT): This heuristic assigns each and every task to the machine which completes in minimum or less time. MCT heuristic does not consider execution time of a task on that machine, so some execution time of task will be increase on the machine where the task has to be assigned.

Minimum Execution Time (MET): In this assigning of heuristic task in to heterogeneous machine which give minimum completion time for that tasks execution. Due to this, it can cause load imbalance on the different resources. Simplicity of implementation is the advantage of MET over the MCT.

Min-min: The Min-min heuristic algorithm starts with the set of all unmapped tasks. After that, minimum completion time for each task is found on every heterogeneous machine. Also, the different task with the overall minimum completion time are get selected and assigned to the corresponding machine (hence the name Min-min). At the end, the newly mapped or assigned task is re-moved from the machine, and the process repeats until all tasks are mapped. Min-min & MCT both heuristic produce minimum completion or execution time.

Max-Min: The max-min mapping heuristic is somewhat similar to the min-min mapping heuristic. The first step of this heuristic is similar to the min-min heuristic algorithm. In the next step, instead of choosing the task having the minimum completion times over all the tasks, max-min algorithm selects the task which having maximum completion times and assigns it to the that machine. The machine availability time is updated and the process is repeated for every task in the Meta task or big task. The max-min mapping heuristic algorithm would generally outperform min-min mapping heuristic algorithm, when the number of short tasks is greater than that of long tasks.

4. CONCLUSION

In this paper we are going to propose the MaxMin+ and Suff+ heuristics, which are hybrid types of MaxMin and Suff algorithms, and acquired by means of combining the latter heuristics with MinMin algorithm. To reap the high computing throughput (means the output get in a particular time slice) in an environment, this new scheduling algorithm was proposed. It selects between two conventional algorithms, Min-Min and Max-Min, each time one acts better than the other founded on the general deviation of minimal completion time of all unassigned duties in a meta-mission. Analysis of our new heuristic was once performed via a simulation environment called Sim. The experimental results show that the Selective algorithm outperforms the natural Min-Min and Max-Min heuristics. We provided designated efficiency upgrades over the widespread unbiased task challenge heuristics.

5. REFERENCES

1. O.H. Ibarra and C.E. Kim, Heuristic Algorithms for Scheduling Independent Tasks on Nonidentical Processors, J. ACM, vol. 24, no. 2, pp. 280-289, 1977.

2. oni, M. Maheswaran, A.I. Reuther, J.P. Robertson, M.D. Theys, B. Yao, D. Hens-gen, and R.F. Freund, A Comparison of Eleven Static Heuristics for Mapping a Class of Independent Tasks onto Heterogeneous Distributed Computing Systems, J. Paral-lel and Distributed Computing, vol. 61, no. 6, pp. 810-837, 2001.

3. H.J. Siegel and S. Ali, Techniques for Mapping Tasks to Machines in Heteroge-neous Computing Systems, J. Systems Architecture, vol. 46, no. 8, pp. 627-639, 2000.

4. R. Duan, R. Prodan, and T. Fahringer, Performance and Cost Optimization for Multiple Large-Scale Grid Workow Applications, Proc. ACM/IEEE Conf. Super-computing, pp. 1-12, 2007.
5. Z. Shi, A Revisit of Fast Greedy Heuristics For Mapping a Class of Independent Tasks onto Heterogeneous Com- puting Systems, J. Parallel and Distributed Com-puting, vol. 67, pp. 695-714, 2007.
6. E. Davis and J.M. Jaffe, Algorithms for Scheduling Tasks on Unrelated Proces-sors, J. ACM, vol. 28, pp. 721-736, 1981.
7. P.C. SaiRanga and S. Baskiyar, A Low Complexity Algorithm for Dynamic Scheduling of Independent Tasks onto Heterogeneous Computing Systems, Proc. 43rd Ann. Southeast Regional Conf., pp. 63-68, 2005.
8. R. Armstrong, D. Hensgen, and T. Kidd, The Relative Perfor- mance of Various Mapping Algorithms is Independent of Sizable Variances in Run-Time Predictions, Proc. IEEE Seventh Heteroge- neous Computing Workshop, pp. 79-87, 1998.
9. M. Maheswaran, S. Ali, H.J. Siegel, D. Hensgen, and R.F. Freund, Dynamic Mapping of a Class of Independent Tasks onto Hetero- geneous Computing Sys-tems, J. Parallel and Distributed Comput- ing, vol. 59, pp. 107-131, 1999.
10. C. Liu and S. Baskiyar, A General Distributed Scalable Grid Scheduler for In-dependent Tasks, J. Parallel and Distributed Com- puting, vol. 69, pp. 307-314, 2009.

Akshay Ranjeet Neharkar, passed HSC from Jai Bhavani Junior College Pachegaon. Now pursuing BE from Dr. DY Patil Institute of Technology Pimpri, Pune-18 in information technology department.

Avinash Sadashiv Kadam, passed HSC from Dadasaheb Rupvate Junior college, A.Nagar. Now pursuing BE from Dr. DY Patil Institute of Technology Pimpri, Pune-18 in information technology department.

Priya Namdev Dhage, passed HSC from BJS Junior college, Pimpri, Pune. Now pursuing BE from Dr. DY Patil Institute of Technology Pimpri, Pune-18 in information technology department.