

Chapter 5.6

Balanced Job Scheduling Based on Ant Algorithm for Grid Network

Nikolaos Preve

National Technical University of Athens, Greece

ABSTRACT

Job scheduling in grid computing is a very important problem. To utilize grids efficiently, we need a good job scheduling algorithm to assign jobs to resources in grids. The main scope of this paper is to propose a new Ant Colony Optimization (ACO) algorithm for balanced job scheduling in the Grid environment. To achieve the above goal, we will indicate a way to balance the entire system load while minimizing the makespan of a given set of jobs. Based on the experimental results, the proposed algorithm confidently demonstrates its practicability and competitiveness compared with other job scheduling algorithms.

INTRODUCTION

The early efforts in Grid computing started as a project to link supercomputing sites, but have now grown far beyond the original intent. The popularity of the Internet, the availability of powerful computers and the high-speed network technologies as well as low cost commodity components, are changing the way we use computers

today. These technology opportunities have led to the possibility of using distributed computers as a single, unified computing resource, leading to what is popularly known as Grid computing.

Today's scientific problems are very complex and need huge computing power and storage capability. Computational Grids are emerging as a new paradigm for solving large scale problems in science and engineering. A Grid scheduler acts as an interface between the user and the distributed resources hiding the complexities of Grid comput-

DOI: 10.4018/978-1-4666-0879-5.ch5.6

ing (Abramson, Giddy, & Kotler, 2000; Buyya, Abramson, & Giddy, 2000). It is also responsible for monitoring and tracking the progress of application execution along with adapting to the changes in the runtime environment of the Grid, variation in resource share availability and failures. A Grid scheduler balances the entire system load while completing all the jobs at hand as soon as possible according to the environment status. The existing job scheduling algorithm such as First Come First Serve (FCFS), Shortest Job First (SJF) may not be suitable for the grid environment. These algorithms seize a lot of computational time due to soar waiting time of jobs in job queue reducing the entire system performance (Somasundaram & Radhakrishnan, 2009; Yan, Wang, Wang, & Chang, 2009).

The users who interact with the Grid can manually assign jobs to computing resources in grids. Thus, grid job scheduling is a very important issue in Grid computing. Referring to the Berkeley Open Infrastructure for Network Computing (BOINC, 2009) project, we will find out an open source platform for volunteer computing and grid computing. Here, job scheduling is one of the most important key factors for achieving Teraflops performance (Kondo, Anderson, & McLeod, 2007). We can also notice that grid scheduling concentrates in improving response times in an environment containing autonomous resources whose availability dynamically varies with time. The Grid scheduler has to interact with the local schedulers managing computational resources and adapt its behavior to changing resource loads. Thus, the scheduling is conducted from the perspective of the user rather than that of the system. Because of its significant importance many job scheduling algorithms for Grids have been proposed in order to obtain a load balancing distribution of processes to computational resources, but open issues still exist (Boyera & Hura, 2005; Buyya, Cortes, & Jin, 2001; Collins & George, 2001; De Ronde, Schoneveld, & Sloot, 1997; Dong & Akl, 2004; Dorigo & Gambardella, 1997; Foster &

Kesselman, 2003; Maghraoui, 2006; Nabrzyski, Schopf & Weglarz, 2004; Salari & Eshghi, 2005; Karatza & Hilzer, 2003; Sonmez & Gursoy, 2007). When things come to practice it is impossible to find a tool for the automatic load balancing of a parallel distributed application.

A Grid scheduler differs from a scheduler for conventional computing systems in lack of full control over the grid because the local resources are in general not controlled by the grid scheduler, but by the local scheduler. Also, the Grid scheduler cannot assume that it has a global view of the grid. Therefore, the demand for scheduling is to achieve high performance computing. The heuristic algorithm Ant Colony Optimization (ACO) with efficient local search can be a useful tool for finding an optimal resource allocation for specific job (Dorigo & Blum, 2005; Lorpunmanee, Sap, Abdullah, & Inwai, 2007). This minimizes the schedule length of jobs. Many research projects, which are concerned with load balancing techniques, use ACO to solve NP-hard problems (Dorigo & Gambardella, 1997; Salari & Eshghi, 2005; Zhang & Tang, 2005).

This paper applies the ACO algorithm to job schedule problems in Grid computing. The resource scheduling in Grid is a NP complete problem. We compare the proposed ACO algorithm with various algorithms which have been designed to schedule the jobs in computational grid. The next step is to compare the proposed ACO algorithm with the most commonly used algorithms. The goal of this work is to describe the methodology and to define the required parameters according with the experimental results and thus to make use of real measurements which show that the proposed ACO is capable of achieving system load balance better than other job scheduling algorithms.

In the following, we will provide the details of related work about various kinds of ACO algorithm and job scheduling in grids. Also, we will describe the methodology, the design and the analysis of the proposed ACO algorithm in job scheduling.

16 more pages are available in the full version of this document, which may be purchased using the "Add to Cart" button on the publisher's webpage:

www.igi-global.com/chapter/balanced-job-scheduling-based-ant/64532

Related Content

Methodology for Information Management and Data Assessment in Cloud Environments

Mariam Kiran, Gregory Katsaros, Jordi Guitartand Juan Luis Prieto (2014). *International Journal of Grid and High Performance Computing* (pp. 46-71).

www.irma-international.org/article/methodology-for-information-management-and-data-assessment-in-cloud-environments/127374

Cloud Standards: Security and Interoperability Issues

Fabio Bracci, Antonio Corradiand Luca Foschini (2014). *Communication Infrastructures for Cloud Computing* (pp. 465-495).

www.irma-international.org/chapter/cloud-standards/82551

Adaptive Threshold Based Scheduler for Batch of Independent Jobs for Cloud Computing System

TAJ ALAM, PARITOSH DUBEYand ANKIT KUMAR (2018). *International Journal of Distributed Systems and Technologies* (pp. 20-39).

www.irma-international.org/article/adaptive-threshold-based-scheduler-for-batch-of-independent-jobs-for-cloud-computing-system/211209

Introduction to Dataflow Computing

Nenad Korolija, Jovan Popoviand Miroslav M. Bojovi (2021). *Handbook of Research on Methodologies and Applications of Supercomputing* (pp. 96-105).

www.irma-international.org/chapter/introduction-to-dataflow-computing/273396

Load Balancing to Increase the Consistency of Replicas in Data Grids

Ghalem Belalem, Naima Belayachi, Radjaa Behidjiand Belabbes Yagoubi (2010). *International Journal of Distributed Systems and Technologies* (pp. 42-57).

www.irma-international.org/article/load-balancing-increase-consistency-replicas/47426