Chapter 5.5

Evaluating Heuristics for Scheduling Dependent Jobs in Grid Computing Environments

Geoffrey Falzon
Brunel University, UK

Maozhen Li
Brunel University, UK

ABSTRACT

Job scheduling plays a critical role in the utilisation of grid resources by mapping a number of jobs to grid resources. However, the heterogeneity of grid resources adds some challenges to the work of job scheduling, especially when jobs have dependencies which can be represented as Direct Acyclic Graphs (DAGs). It is widely recognised that scheduling m jobs to n resources with an objective to achieve a minimum makespan has shown to be NP-complete, requiring the development of heuristics. Although a number of heuristics are available for job scheduling optimisation, selecting the best heuristic to use in a given grid environment remains a difficult problem due to the fact that the performance of each original heuristic is usually evaluated under different assumptions. This paper evaluates 12 representative heuristics for dependent job scheduling under one set of common assumptions. The results are presented and analysed, which provides an even basis in comparison of the performance of those heuristics. To facilitate performance evaluation, a DAG simulator is implemented which provides a set of tools for DAG job configuration, execution, and monitoring. The components of the DAG simulator are also presented in this paper.

INTRODUCTION

The past few years have witnessed a rapid development of grid computing systems and applications (Li & Baker, 2005). A grid is a heterogeneous

computing environment in that resources may have various computing capacities. Job scheduling which is a process of mapping jobs to resources plays a crucial role in resource utilisation in grid environments. Jobs can be generally classified into two classes - independent jobs and depen-

dent jobs. The problem of scheduling m jobs to n resources with an objective to minimise the total execution time (makespan) has been shown to be NP-complete requiring the development of heuristics (Ibarra & Kim, 1977). Braun et al (Braun et al., 2001) evaluated eleven heuristics for mapping independent tasks to heterogeneous computing environments which helps select the best heuristic to use in a given environment. Mapping dependent jobs to heterogeneous computing environments such as grids poses more challenges in that the dependencies of jobs have to be taken into account in the scheduling process.

Jobs with dependent tasks can be represented by Directed Acyclic Graphs (DAGs) in which each node represents an executable task and each directed edge represents data transfers between two tasks. We assume that DAGs always have a single entry node (i.e., a node with no parents) and a single exit node (i.e., a node with no children). To compute a schedule in grid environments, scheduling algorithms require the following information:

- Information on computing resources available in a grid environment.
- An Expected Completion Time (ECT) matrix in which the expected time to execute a task on each machine is provided.
- The network bandwidth of the communication link connecting any two computers.

This paper evaluates 12 heuristics for scheduling dependent jobs in grid environments. To facilitate performance evaluation, a DAG simulator is implemented which provides a set of tools for DAG job configuration, execution and monitoring. The following assumptions are made when scheduling a job with a number of dependent tasks:

- A computer can execute only one task at a time.
- Task execution can only start after all data required by the task is available.

- Data transfer can only begin when a task is completed.
- Data transfer is scheduled according to the order of the tasks requiring the data. This is referred to as in-order scheduling (Wang, Siegel, Roychowdhury, & Maciejewski, 1997).
- A computer can transfer only one set of data at a time. Data transfer and task execution can be performed in parallel.
- Task execution and data transfer are non-pre-emptive.
- Task execution is computed according to the order defined by a scheduler.
- An ECT matrix is provided. This can be generated by dividing the workload of a job by the MIPS rating of the processor where the job is executed.
- For list scheduling heuristics (Sih & Lee, 1993; Topcuoglu, Hariri, & Wu, 2002), partial makespan (the time required to execute the first n jobs in the workflow) is calculated by ignoring data transfer to jobs that are not yet scheduled.

The remainder of the paper is organised as follows. It briefly describes the twelve heuristics used in the evaluation, and presents the components of the DAG simulator for DAG jobs configuration, execution and monitoring. Then it evaluates the 12 heuristics and analyses the performance results. Finally it concludes the paper.

HEURISTICS FOR SCHEDULING DEPENDENT JOBS

The 12 heuristics that were evaluated are Min-min (Maheswaran, Ali, Siegel, Hensgen, & Freund, 1999), Max-min (Maheswaran et al., 1999), Sufferage (Maheswaran et al., 1999), XSufferage (Casanova, Legrand, Zagorodnov, & Berman, 2000), Baseline (Maheswaran & Siegel, 1998), Levelised Min-Time (LMT) (Iverson, Ozguner, &

Related Content

Plant Disease Detection Using Sequential Convolutional Neural Network
Anshul Tripathi, Uday Chourasia, Priyanka Dixitand Victor Chang (2022). *International Journal of Distributed Systems and Technologies (pp. 1-20).*
www.irma-international.org/article/plant-disease-detection-using-sequential-convolutional-neural-network/303672

Face Recognition: A Tutorial on Computational Aspects
Alexander Alling, Nathaniel R. Powersand Tolga Soyata (2016). *Emerging Research Surrounding Power Consumption and Performance Issues in Utility Computing (pp. 405-425).*
www.irma-international.org/chapter/face-recognition/139855

Introduction to Serverless Computing
Radhika Kavuriand Satya kiranmai Tadepalli (2024). *Serverless Computing Concepts, Technology and Architecture (pp. 1-19).*
www.irma-international.org/chapter/introduction-to-serverless-computing/343717

Opposition-Based Deer Hunting Optimization-Based Hybrid Classifier for Intrusion Detection in Wireless Sensor Networks
Mohandas V. Pawarand Anuradha J. (2022). *International Journal of Distributed Systems and Technologies (pp. 1-29).*
www.irma-international.org/article/opposition-based-deer-hunting-optimization-based-hybrid-classifier-for-intrusion-detection-in-wireless-sensor-networks/300356

Sharing Experience Around Component Compositions: Application to the Eclipse Ecosystem
Grégory Bourguin, Arnaud Lewandowskiand Myriam Lewkowicz (2013). *International Journal of Distributed Systems and Technologies (pp. 15-28).*
www.irma-international.org/article/sharing-experience-around-component-compositions/104715