Chapter 12
Incentive-Based Scheduling for Green Computational Grid

Low Tang Jung
Universiti Teknologi PETRONAS, Malaysia

Ahmed Abba Haruna
Skyline University, Nigeria

ABSTRACT

In the computing grid environment, jobs scheduling is fundamentally the process of allocating computing jobs with choices relevant to the available resources. As the scale of grid computing system grows in size over time, exponential increase in energy consumption is foreseen. As such, large data centers (DC) are embarking on green computing initiatives to address the IT operations impact on the environment. The main component within a computing system consuming the most electricity and generating the most heat is the microprocessor. The heat generated by these high-performance microprocessors is emitting high CO2 footprint. Therefore, jobs scheduling with thermal considerations (thermal-aware) to the microprocessors is important in DC grid operations. An approach for jobs scheduling is proposed in this chapter for reducing electricity usage (green computing) in DC grid. This approach is the outcome of the R&D works based on the DC grid environment in Universiti Teknologi PETRONAS, Malaysia.

INTRODUCTION

Increasing demand for computing resources in recent years has led to the development of grid computing (GC) (Abbas, 2004; Berman, Fox, & Hey, 2003). Computing grid is basically taking the advantage of the idle time of thousands or millions of computers throughout the world to perform computing tasks. In general, grid can be seen as a large-scale dynamic collection of heterogeneous independent computers which are geographically distributed and interconnected by networks for processing massive tasks.

Grid computing originated in the early 1990's, when Ian Foster (Foster & Kesselman, 1999) was promoting a program to elevate shared computing to some global level. Just like the Internet which is a tool for mass communications, grids make computer resources and storage spaces globally available for

DOI: 10.4018/978-1-7998-6709-8.ch012

users. In essence, grid computing is all about getting computer systems operating together to perform computing tasks. The grid has the ability to share heterogeneous distributed assets from programs, data, storage and servers. There are three major factors that have drawn much attention from grid researchers and developers. The first and the prime concern is resource sharing i.e. resource sharing is perhaps the root of grid computing philosophy. Grid is putting computer systems into a prime position in ensuring all grid users are benefiting from the efficiencies of sharing the computing resources. More importantly, grid gives users extra computing power to compute tasks that cannot be computed using just one or few computers. The second factor behind grid is safe access which is directly related to the first factor. It is logically understood that when there are numerous users sharing the computing resources, safe access to those resources is a must with regards to data/information privacy. The third factor is the efficient utilization of grid resources. This includes a mechanism to allocate jobs effectively in real-time among the many users. Consequently, efficient utilization of resources can lessen the jobs' waiting (or completion) time.

In grid environment, scheduling is fundamentally the process of allocating computing jobs with choices relevant to the resources available over the multiple administrative domains. Scheduling may include searching over multiple administrative domains to utilize a single machine or scheduling just one job to make use of multiple resources in a single site or across multiple sites. Alternatively, scheduling can be seen as a process that maps and handles the execution of interdependent tasks on distributed resources. Scheduling comprises two important elements. First, scheduling is a decision making function i.e. to determine a job schedule. Second, scheduling is a collection of principles, models and methods. The issue of mapping tasks on distributed resources goes to a class of problem referred to as NP-complete problems (Galstyan, Czajkowski, & Lerman, 2005). For such problems, no known algorithms can create an optimal solution within polynomial time. Thus, scheduling is a dynamic allocation of resources over time in order to perform a collection of tasks raised in a variety of scenarios.

As large scale distributed computing system tends to grow in size over time, adding more and more computing nodes and storage resources resulting in exponential increase in energy consumption (Orgerie, Lefevre, & Gelas, 2010). Consequently, many large corporate information technology (IT) departments have embarked on green computing initiatives to address their IT operations impact on the environment (Curry, Guyon, Sheridan, & Donnellan, 2012). Particularly in reducing the carbon emission footprint. Green computing and green technology refer to the study and practice of environmentally sustainable computing for information technology (IT). Green computing includes the implementation of best practices, such as energy efficiency central processing units (CPUs), peripherals and servers. In addition, green technology aims to reduce resource consumption and reduce the disposal of electronic waste. Green computing is important for all classes of systems, ranging from small handheld systems to large-scale data centers (Mittal, 2014).

In all computer systems, it is the electrical energy that provides power to the hardware components to perform computation where the dissipated rate of energy is calculated as power (in watts or joules per second) consumed. The ever increasing energy consumption in large-scale distributed computing systems brings along higher operation costs (e.g. electricity bills) and caused negative environmental impacts such as CO_2 emissions. Unless steps are taken to save energy and go green, the global data centers' (DC) share of carbon emission is estimated to rise from 307 million tons in 2007 to 358 million tons in 2020 (Cook, 2011).

Electricity consumption for cooling purpose in DC is known to be the most expensive operational cost. Inefficient cooling may lead to high temperature (hot spot) and this in turn leads to more frequent hardware failures. It is common in most of the DCs nowadays in housing up to tens of thousands of

Related Content