An Analytical Model for Resource Characterization and Parameter Estimation for DAG-Based Jobs for Homogeneous Systems

Mohammad Sajid, Jawaharlal Nehru University, New Delhi, India

Zahid Raza, Jawaharlal Nehru University, New Delhi, India

ABSTRACT

High Performance Computing (HPC) systems demand and consume a significant amount of resources (e.g. server, storage, electrical energy) resulting in high operational costs, reduced reliability, and sometimes leading to waste of scarce natural resources. On one hand, the most important issue for these systems is achieving high performance, while on the other hand, the rapidly increasing resource costs appeal to effectively predict the resource requirements to ensure efficient services in the most optimized manner. The resource requirement prediction for a job thus becomes important for both the service providers as well as the consumers for ensuring resource management and to negotiate Service Level Agreements (SLAs), respectively, in order to help make better job allocation decisions. Moreover, the resource requirement prediction can even lead to improved scheduling performance while reducing the resource waste. This work presents an analytical model estimating the required resources for the modular job execution. The analysis identifies the number of processors required and the maximum and minimum bounds on the turnaround time and energy consumed. Simulation study reveals that the scheduling algorithms integrated with the proposed analytical model helps in improving the average throughput and the average energy consumption of the system. As the work predicts the resource requirements, it can even play an important role in Service-Oriented Architectures (SOA) like Cloud computing or Grid computing.

Keywords: *Directed Acyclic Graph (DAG), Energy, High Performance Computing, Service Oriented Architectures (SOA), Turnaround Time (TAT)*

DOI: 10.4018/ijdst.2015010103

1. INTRODUCTION

The landscape of computing is changing continuously. The traditional computing paradigms are being replaced by high performance computing paradigms viz. grid computing, cloud computing and internet of things (Foster & Kesselman, 1998; Buyya et al., 2009). According to Intel's info-graphic, "The Internet of Things" (Humprey, 2011), 31 billion devices and four billion people will be connected to the Internet by 2020 i.e. every person will have close to 8 devices connectivity. The data generated by these devices will be a tremendous amount that appeals to deploy high performance computing models having capability of handling diverse workloads. The performance in such a scenario becomes the fundamental key to any technology which in turn depends on the optimized management of resources. The optimization of resource management is considered as one of the indispensable part of any computing paradigm (Husain et al., 2013). It is very essential because if a job is not able to finish its execution due to lack of proper resources, it will be suspended or restarted resulting in an escalated cost. In the worst case, the job may even fail necessitating it to execute on another set of resources selected afresh. These scenarios show wastage of resources and appeal advanced features like resource reservation or prediction of required resources for the given job. The resource requirement prediction model predicts the resource requirements of the job before its execution and it can be done statically or dynamically. Resource prediction tools are employed to help the resource manager in order to use available resource in an optimized manner and guarantees that each of the jobs has always enough resources to meet the agreed Quality of Service (QoS). The feasible prediction of resources leads to optimized resource management that results in many benefits e.g. higher throughput of the system, lower turnaround time of the job, higher utilization, reduction in unnecessary consumption of resources, lower monetary costs and lesser negative effects on the environment (Berl et al., 2009; U.S. Environmental Protection Agency, 2007; Hamilton, 2009; Jarvis et al., 2006; Pamlin, 2008). Resource prediction models are also very helpful in service provider computing models. In the case of cloud based computing with scarce resources, the nature of the jobs is usually heterogeneous i.e. it can range from high performance jobs to various common web services. If the resources are allocated to the jobs without any appropriate consideration, this can lead to inefficient resource consumption. Therefore, resource requirement prediction becomes the key to several crucial system design and deployment decisions such as, workload management, capacity planning and system sizing. The same can be done by the user or the system can generate it on its own by employing knowledge based models and tools. If the resource requirement specification is provided by the user, it may lead to over-estimation or under-estimation. The overestimation results in wastage of resources whereas underestimation does not lead to the desired level of performance of the application. The resource requirement prediction model characterizes the required resources and helps the resource manager to allocate the appropriate number of resources to the submitted jobs. There have been many prediction models proposed in the literature based on the historical information (Ali et al., 2004; Bohlouli & Analoui, 2009; Smith et al., 2004; Gibbons, 1997; Caron et al., 2010; Dinda, 2002). These predictor models raise several significant issues some of them being:

1. How to define similarity? Some work in the literature suggests that if the same user submits the jobs on the same machine employed in past, the jobs may be considered similar (Ali et al., 2004; Bohlouli & Analoui, 2009; Smith et al., 2004; Gibbons, 1997; Caron et al., 2010; Dinda, 2002);
2. The predictor cannot predict how long an application which ran on system A will now take on system B to finish;
3. The input to jobs X and Y may be different that execute on system A and B respectively. The job X in past had two inputs on the same