International Journal of Innovative
Computing, Information and Control
Volume **7**, Number **5(A)**, May **2011**

 ISSN 1349-4198
pp. **2099–2108**

RESOURCE ALLOCATION ANALYSIS MODEL BASED ON GRID ENVIRONMENT

Huey-Ming Lee, Jin-Shieh Su and Chia-Hsien Chung

Department of Information Management
Chinese Culture University
No. 55, Hwa-Kung Road, Yang-Ming-San, Taipei 11114, Taiwan
hmlee@faculty.pccu.edu.tw; jonguser@gmail.com

Received December 2009; revised June 2010

Abstract. *In this paper, we propose a performance analysis of resource allocation model for grid resources under the grid computing environment. By the proposed model, we can make all grid nodes be load-balancing. The proposed model can detect the grid nodes resource status about CPU usage and memory usage, analyze the grid nodes resources load grades, and make the plans and allocations of the resources of collaborative nodes optimize. Via implementing this model, we can enhance the grid environment performance efficiently.*
Keywords: Grid computing, Resource allocation

1. **Introduction.** The term "Grid" was coined in the mid 1990s to denote a proposed distributed computing infrastructure for advanced science and engineering [3]. In grid environment, user may access the computational resources at many sites [2,5]. The functions of information systems based on grid computing architectures are resources (e.g., CPUs, memory, storages, etc.) sharing, collaborative processing, reliable and secure connection, etc [3,7-9]. However, each resource of coordinate nodes in the grid environment, (e.g., CPU loading, memory usage, etc.) changes dynamically. Therefore, how to optimize these resources usages is an important issue.

Foster *et al.* [3] presented grid resource allocation and management (GRAM). GRAM simplifies the usage of remote systems by providing a single standard interface for requesting and using remote system resources for the execution of "jobs". The most common usage of GRAM is a remote job submission and control. This is typically used to support distributed computing applications. GRAM is designed to provide a single common protocol and application programming interface (API) for requesting and using remote system resources, by providing a uniform, flexible interface to local job scheduling systems. Miller *et al.* [12] proposed Paradyn parallel performance measurement tools which can identify the heaviest loading process by heuristic method and find out the bottleneck point. Lee *et al.* [7] proposed a dynamic supervising model which can utilize the grid resources, e.g., CPU, storages, etc., more flexible and optimal. Lee *et al.* [8] presented an optimal analyzing resources model that can receive the information about CPU usage, number of running jobs of each grid resource node to achieve load balancing and make the plans and allocations of the resources of collaborated nodes optimize. Iosup *et al.* [6] proposed synthetic workloads to evaluate performance on grid. Lee *et al.* [9] proposed a process schedule analyzing model, when a new job is entered to be processed and the load of this node is heavy, the model can schedule some jobs to the other nodes to process by the data base of grid information. Sanjay and Vadhiyar [13] presented a performance modeling of parallel applications for grid scheduling that developed a comprehensive set