Journal of Network and Computer Applications 88 (2017) 99–111

Contents lists available at ScienceDirect

Journal of Network and Computer Applications

journal homepage: www.elsevier.com/locate/jnca

Review

Load balancing in grid computing: Taxonomy, trends and opportunities

Sumair Khan[a], Babar Nazir[a], Iftikhar Ahmed Khan[a], Shahaboddin Shamshirband[b,c,*], Anthony T. Chronopoulos[d,e]

[a] Department of Computer Science, COMSATS Institute of Information Technology, University Road, Tobe Camp, 22060 Abbottabad, Pakistan
[b] Department for Management of Science and Technology Development, Ton Duc Thang University, Ho Chi Minh City, Vietnam
[c] Faculty of Information Technology, Ton Duc Thang University, Ho Chi Minh City, Vietnam
[d] Department of Computer Science, University of Texas at San Antonio, San Antonio, TX, USA
[e] (Visiting Faculty) Department of Computer Science, University of Patras, Patra, Greece

ARTICLE INFO

Keywords:
Grid computing
Load balancing
Task migration
Resource allocation

ABSTRACT

Grid computing is used to provide different services to users through resources that are geographically dispersed, dynamic, and heterogeneous in nature. In grid computing, load balancing plays a vital role in the re-allocation of user jobs when the grid resources become overloaded. In the past few years, scores of load balancing strategies have been proposed by researchers to improve response time, communication overhead, throughput, and resource utilization. In this paper, surveyed load balancing strategies are divided into two broad categories, some supporting task migration and some of them having no support for task migration during the load balancing process. Each category is further categorized based on the basis of grid resource topology, that includes the flat resource topology and hierarchical resource topology. We discussed and compared the number of dynamic load balancing strategies related to these categories on the basis of different load balancing features and performance metrics.

1. Introduction

Grid computing is an environment that utilizes computing resources that are geographically distributed. Grid computing provides a virtual environment to users, integrating data, and computing resources to accomplish solutions for various types of issues (Blatecky, 2002). Users can utilize computing resources transparently without considering location and operating environment constraints (Klous et al., 2006). With the emergence of grid computing technology, sequential applications can be parallelized and executed on unutilized heterogeneous resources that are possessed by organizations all over the world (Krauter et al., 2002).

Grid computing can be used to solve problems related with computation intensive applications related with different areas. Ecommerce applications can also benefit from grid computing in the form of faster decision making and accurate forecasting (Manjula and Karthikeyan, 2010). Grid computing permits researchers to use the grid resources spread across the world to solve problems related to the earth observation, marine, and environmental sciences (Ascione et al., 2006). In the field of medical sciences, grid computing can give benefits in the form of decision making at remote sites (The Future of Healthcare, 2009). Specialized practitioners with their expertise can take decisions based on computing intensive operations and analysis through grid computing resources. In the field of bio-informatics, the day by day need of grid computing is growing as the distributed information related to bio-informatics is increasing (Manjula and Raju, 2010).

Resource management is a challenging issue in grid environment as it involves the management of grid resources distributed across the globe (Krauter et al., 2001). Resources are allocated so that they collectively solve a problem that has high- performance requirements during the processing (Krauter et al., 2002). Due to the dynamic nature of grid resources, resource management handles the complex problem of resource re-allocation, thus the user's job continues to get an improved response time in a grid environment. Load balancing mechanism is a suitable solution that reallocates the grid resources efficiently according to the changes in the grid resource usage (Izakian et al., 2010).

Load balancing plays a vital role in the grid computing in the utilization of grid resources that are globally distributed. Grid computing shares resources, among the users for execution of the tasks. However, the amount of resources assigned to a specific user application may vary from time to time. Therefore, the overall performance of the grid computing systems can be affected. Without the use of efficient

* Corresponding author at: Ton Duc Thang University, Ho Chi Minh City, Vietnam.
E-mail address: shahaboddin.shamshirband@tdt.edu.vn (S. Shamshirband).

http://dx.doi.org/10.1016/j.jnca.2017.02.013
Received 29 October 2016; Received in revised form 18 January 2017; Accepted 20 February 2017
Available online 21 February 2017
1084-8045/

load balancing mechanism, some resources can be overloaded and some resources can be idle for a long period of time, resulting in degraded system performance (Foster et al., 2001). For resource balancing in grid computing, efficient load balancing mechanism is very critical through which grid resource utilization can be improved. This ultimately improves throughput and response time (Venkatesan and Solomi, 2011).

The process of load balancing involves four basic policies, namely (i) Information, (ii) selection, (iii) transfer and (iv) location (Mukhopadhyay et al., 2010).

(i) *Information policy:* This policy plays a vital role in load balancing in terms of which task information is to be collected, from which resource and the time of its collection.
(ii) *Location policy:* This policy is used to determine a suitable resource or receiver that will become part of load balancing process and where the task should be transferred.
(iii) *Transfer Policy:* Such a policy deals with the decision making pertaining to the load balancing process. It continuously checks the grid resources and initiates the load balancing process if it finds the overloaded resource.
(iv) *Selection policy:* This policy will select a task from an overloaded grid resource and will transfer it to an idle (or receiver) resource that is in a position to accept the transferred task (Yagoubi et al., 2006).

Demand for efficient load balancing strategies is also becoming evident to fulfil the user needs (Narkhede and Khandare, 2013). Researches are facing great challenge in proposing load balancing strategies that can give better performance according to the needs of the application (Yagoubi et al., 2006; Eager et al., 1986). A load balancing strategy whether it is flat or hierarchical (Reddy and Roy, 2012), is designed to spread the load among heterogeneous resources in a fair manner so that the maximum utilization of resources can be achieved. In the design of a load balancing strategy, some basic features that can improve the performance of load balancing must be considered. In this survey, the following such features are taken into consideration while comparing the different load balancing strategies.

(1) *Scalability:* It allows the provision to grid resources to join the grid system without degrading the system performance (Östberg, 2009). Scalability is an important feature because in the grid environment, the numbers of resources are not limited; hence a load balancing strategy must cater for any number of resources.
(2) *Dynamic grid resources:* In a Grid environment, the computational resources exhibit dynamic behavior in terms of their availability (Meddeber et al., 2011). The Grid resources can join and leave the grid environment, or they may not be available at any time due to network failure.
(3) *Fine grained and Coarse-grained Jobs (*Engler et al., 1993*):* The fine-grained jobs have subtasks that are dependent on each other, and they communicate many times during the execution process. In the case of coarse-grained jobs, sub tasks have no dependency on each other and without any communication they complete their execution.
(4) *Flat and hierarchal grid topology:* In the load balancing based on hierarchical topology, resources at the higher-level controls the group of resources at the lower level, and this hierarchy ends at the control of individual resources (Krauter et al., 2000). The Flat-based topology is further subdivided into two sub types namely centralized and decentralized topology. In the centralized approach, a single controller has the responsibility to do the load balancing process between the grid resources. In a decentralized approach, there are a number of controllers designated to control the load balancing process.
(5) *Homogenous and heterogeneous grid resources:* The definition of homogeneous/heterogeneous computing environments is dependent on the application (Chen, 2011). The resources in the homogenous system have similar hardware and software configurations resulting in generating similar results for similar problems. Whereas in the case heterogeneous systems, there is no guarantee that the resources have the similar characteristics.
(6) *Task dependencies:* In general, load balancing in grid computing is a challenging issue even when the tasks are not dependent on each other (Singh et al., 2005). The dependencies between the tasks raise the difficulty in achieving higher level of load balancing.
(7) *Fault Tolerance:* The fault tolerance provides the functionality which enables the correct working of a grid system in the presence of faults (Townend and Xu, 2009). The need of fault tolerance features also becomes more evident because resources may have different organizational boundaries thus there is no control on resource availability.
(8) *Consideration of Resource Processing Capacity:* This feature in load balancing strategy is used to improve the performance of load balancing decisions by examining the processing capability of each resource of the grid environment (Sharma and Bhatia, 2013).

We also considered the performance metrics for the comparison of load balancing strategies. The selected metrics are:

(PM1) Average Response Time: Performance of any load balancing strategy is affected by the average response time of jobs submitted to the grid. It can be defined as the total time taken from the job submission, until the user receives the response (Shan et al., 2004). The response time is directly affected by the communication delay and the processing capability of the resources on which the job is running. To reduce average response time of a grid system, it is necessary to have an efficient load balancing strategy.

(PM2) Communication Overhead: In the design of a load balancing strategy, a major goal is to decrease the communication overhead generated by the information exchange between the resources to achieve load balancing (El-Zoghdy, 2011). The minimization of the communication overhead is necessary because the network bandwidth is shared by the grid resources, and it directly affects the throughput of the system.

(PM3) Resource Utilization: One of the important design goals of a load balancing policy is to reduce the idle time of the grid resources in order to achieve a better resource utilization. It can be accomplished by assigning jobs to the best matching grid resources; consequently, minimizing the response time (Bindu et al., 2011).

(PM4) Communication Delay: Communication delay is an important factor that must also be considered during the design of load balancing strategies, which affects the average response time (Heiss and Schmitz, 1995).

In a given grid environment, the communication delay varies because the network devices and channels are shared by the grid resources,

(PM5) Throughput: The ultimate goal to achieve by any load balancing strategy is to maximize the throughput of the grid system. The incorporation of different design goals in load balancing can help achieving better throughput in the system (Budhani et al., 2010).

The rest of the paper is organized as follows. Section 2 presents the existing survey, followed by the Section 3 that presents a comprehensive study of load balancing strategies, and a comparison of strategies based on different features and performance metrics. Open research issues related with load balancing are discussed in Section 4. Finally, in Section 5 conclusions from the survey are drawn.

2. Existing surveys

The load balancing mechanism plays a vital role in the timely execution of the tasks and efficient utilization of the grid resources. This survey provides a detailed overview and comparison of various