Performance Parameters for Load Balancing Algorithm in Grid Computing

Pardeep Kumar[1], Ravi Rastogi[1],
Shiv Kumar Gupta[1], and Krishna Kumar Verma[2]

[1] Computer Science and Engineering & Information Technology,
Jaypee University of Information Technology,
Waknaghat, Solan, H.P, India
{pardeepkumarkhokhar,ravikumarrastogi,shivku2003}@gmail.com
[2] Computer Science, Awadhesh Pratap Singh University, Rewa, M.P., India
krishnamscit.verma@gmail.com

Abstract. Grid computing is a new parallel and distributed computing paradigm that provides resources for large scientific computing applications. This paper describe the new analysis of parameters for load balancing in Grid that is responsible for performance of load balancing in Grid Computing. At end of paper we show a table of comparison of various load balancing algorithms based on different parameters.

Keywords: Grid Computing, Load Balancing, Performance Parameters, Grid topology.

1 Introduction

The Grid is a hardware/ software infrastructure that enables heterogeneous geographically separated clusters of processors to be connected in a virtual environment [5], [8]. A system [3], [14] of distributed computers with tens or hundreds of computers connected by high speed networks of the system [13]. Many researchers have been carried out on load balancing for many years with the aim is to find the load balancing schemes.

2 Load Balancing

Load Balancing is the assignment of work to processors and is critical in parallel simulations [1]. The problem of load balancing is much more difficult in large distributed systems. Consider a network of n processors, described by a connected undirected graph $G = (N, A)$ where $N = \{1, \dots, n\}$ and A is the set of arcs connecting different processors. These processors are cooperating in the execution of some computational task. Let $x_i(t) \geq 0$ be the load handled by processor i at time$t$, where t is a non-negative integer time variable. Let L be the total load. Let $A(i)$ be the set of neighbors of the ith processor. We assume that

V.V. Das and J. Stephen (Eds.): CNC 2012, LNICST 108, pp. 333–336, 2012.

$$x_j^i(t) = x_j(\tau_j^i(t)) \quad \ldots (1)$$

Where $\tau_j^i(t)$is an integer variable satisfying $0 \leq \tau_j^i(t) \leq t$

For each neighboring processor $j \epsilon A(i)$if$x_i(t) > x_j^i(t)$, then a nonnegative amount of load, denoted by$s_{ij}(t)$, is transferred from i to$j$; no load is transferred if$x_i(t) \leq x_j^i(t)$, in which case, we let$s_{ij}(t) = 0$. For notational convenience, we also let $s_{ij}(t)$ =0 if t$\in T^i$. We assume that a load transfer can take some time to be completed. We use $v_{ij}(t)$ to denote the amount of load that has been sent from processor i to processor j before time$t$, but has not been received byprocessor j before time t. Let $r_{ij}(t)$be the load received by processor$j$ from processor$i$ at time t. We then have

$$x_i(t+1) = x_i(t) - \sum_{j \in A(i)} s_{ij}(t) + \sum_{j \in A(i)} r_{ij}(t), \quad \ldots (2)$$

And

$$v_{ij}(t) = \sum_{\tau=0}^{t-1} (s_{ij}(t) - r_{ij}(\tau)) \quad \ldots (3)$$

Where we are making the implicit assumption that $v_{ij}(0) = 0$. Since no load is assumed to be in transit at time zero, we have $\sum_{i=1}^{n} x_i(0) = L$, and using Eqs. (b) and (c), we easily obtain the load conservation equation

$$\sum_{i=1}^{n} (x_i(t) + \sum_{j \in A(i)} v_{ij}(t) = L, \forall t \geq 0 \quad \ldots (4)$$

3 Performance Parameters for Load Balancing

Computer system performance depends on load balancing algorithm in computational grid [15]. Reference [13] and [6] also discusses job arrival rate and CPU processing rate etc. Load balancing algorithms is measured by the following parameters:-

A. Fault Tolerant: Fault tolerance is an important property in Grid computing. Also in large-scale grids, the probability of a failure is much greater than in traditional parallel systems [10], [12].

B. Stability: Stability gains in the load balancing algorithm by obtaining faster performance by a specified amount of time. In reference [11] there are several factors which affect the stability of the load balancing in the grid.

C. Nature of Load Balancing Algorithms

Static load balancing assigns load to nodes probabilistically or deterministically without consideration of runtime events. A Dynamic load balancing policy [7] can use either local or global information.

D. Process Migration: Process migration [4] parameter provides when a system decides to export a process.

3.1 Some Others Performance Parameters

Some other parameters are Communication delay, grid topology, workload and negotiation protocol, and complexity [9] that is also responsible they are:

A. Complexity: Complexity is a measure of the performance of an algorithm in term of CPU time and memory usage in the system [2].Following Fig.1 shows different aspect of complexity.

B. Security in Grid: Security is needed in the form of secure logins, authentication and authorization, access rights and privileges their surroundings [8].

C. Grid Topology: As topological point of view, grid G is considered as a collection of C number of clusters. An example of such topology is shown in Fig. 2 & Fig. 3 [13].

Algorithm / Parameters	Scheduling and Load Balancing Strategy	Load Balancing on Arrival (LBA) for Small-Scale (intraGrid) System	Live Update Information Algorithm (LUIA)	Agent Based Resource Management System
Efficiency	Good	For Small Scale Grid	High Good	Good
Security	No Security	No Security	No Security	No Security
Fault Tolerance	No	No	No	No
Scalability	Not Good	Medium	Not Scalable	High
Stability	Less	Less	No	Instability
Overload Rejection	No	No	Yes	No
Forecasting Accuracy	Less	Less	Less	Medium
Static/ Dynamic	Dynamic	Dynamic	Dynamic	Agent Based
Resource Utilization	High	Less	High	High
Performance	High	High	High	High

TABLE I Comparison of load balancing algorithms

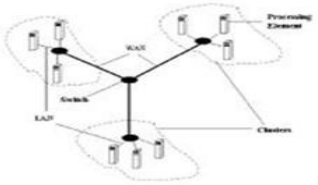

Fig.2 Grid Topology

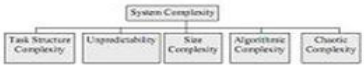

Fig. 1 Different aspect of System Complexity

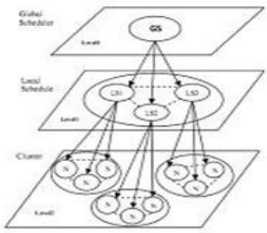

Fig.3 3-Level Grid Topology

4 Comparison among Scheduling and Load Balancing Strategies

This section presents the comparative study of "SLB strategy [13] ", "LBA [6] ", "ARMS[11]" and "live update information algorithm [9]" Scheduling and load balancing strategy: This strategy [13] is divided into three steps first step is Workload Estimation, second step is Decision Making and third step is Job Transferring.

4.1 Load Balancing on Arrival (LBA)

Load Balancing on Arrival (LBA) [6] is proposed for small-scale (intraGrid) systems.

4.2 Agent Based Resource Management System

The ARMS [11] is an Agent-based resource manager for grid computing, it is a scalable and adaptable algorithm.

4.3 Live Update Information Algorithm

Live Update Information Algorithm [9] is a dynamic algorithm. This criterion is expressed by the efficiency function $f_x = totalLoad/speed$. So, for each task, $2 \times n + 1$ messages are exchanged.

5 Conclusion

Through this paper, we have described multiple aspects of Grid Computing and introduced numerous concepts which illustrate its broad capabilities. Grid Computing is definitely a promising tendency to solve high demanding applications and all kinds of problems. This paper presents number of parameters for load balancing like communication delay, security, fault tolerance, efficiency, overload rejection, complexity, grid topology etc. At the end of paper, given table shows performance of various load balancing algorithm based on different parameters.

References

1. Bode, A.: Load Balancing In Distributed Memory Multiprocessors. Invited paper. IEEE (1991)
2. Ranganathan, A., Campbell Roy, H.: What is the Complexity of a Distributed Computing System? National Science Foundation, NSF CCR 0086094 ITR and NSF 99-72884 EQ
3. Eager, D.L., Lazowska, E.D., Zahorjan, J.: Adaptive load sharing in homogeneous distributed systems. IEEE Transactions on Software Engineering 12(5), 662–675 (1986)
4. Milojičić, D.S., Douglis, F., Paindaveine, Y., Wheeler, R., Zhou, S.: Process Migration. ACM Computing Surveys 32(3), 241–299 (2000)
5. Harvey, D.J.: Development of the Grid Computing Infrastructuer, NASA Ames Research Center Sunnyvale, California, http://webpages.sou.edu/~harveyd/presentations/Grid.ppt
6. Prathima, G., Saravanakumar, E.: A novel load balancing algorithm for computational grid. International Journal of Computational Intelligence 1(1), 20–26 (2010)
7. Lin, H.-C., Raghavendra, C.S.: A Dynamic Load Balancing Policy With a Central Job Dispatcher (LBC). IEEE Transactions on Software Engineering 18(2), 148–158 (1992)
8. Alkadi, I., Gregory, S.: Grid Computing: The Trend of The Milleninum. Review of Business Information System 11(2), 33–38 (2007)
9. Psoroulas, I., Anognostopoulos, I., Loumos, V., Kayafas, E.: A Study of the Parameters Concerning Load Balancing Algorithms. IJCSNS 7(4) (April 2007)
10. Jayabharathy, J., Parveen, A.: A Fault Tolerant Load Balancing Model for Grid Enviroment. International Journal of Recent Trends in Engineering 2(2) (Novermber 2009)
11. Salehi, M.A., Deldari, H.: A Novel Load Balancing Method in an Agent-based Grid. IEEE, Iran Telecommunication Research Center (ITRC), 1-4244–0220-4 (2006)
12. Bheevgade, M., Mujumdar, M., Patrikar, R., Malik, L.: Achieving Fault Tolerance in Grid Computing System. In: Proceeding of 2nd National Conference on Challenges & Opportunities in Information Technology (COIT 2008). RIMT-IET (March 29, 2008)
13. Nandagopal, M., Uthariaraj, R.V.: Hierarchical Status Information Exchange Scheduling and Load Balancing For Computational Grid Environments. IJCSNS 10(2) (February 2010)
14. Malik, S.: Dynamic Load Balancing in a Network of Workstation, 95.515 Research Report (Novermber 19, 2000)
15. Sharma, S., Singh, S., Sharma, M.: Performance Analysis of Load Balancing Algorithms. World Academy of Science, Engineering and Technology 38 (2008)