

Suitability of Data Intensive Application in a Bandwidth Constrained Global Grid Network

T.R. Srinivasan and R. Shanmugalakshmi

Department of IT, Vidyaa Vikas College of Engineering and Technology,
Tiruchengode, India

Department of CSE and IT, Government College of Technology, Coimbatore, India

Abstract: Problem statement: To check the suitability of real time data intensive applications in a global grid network. **Approach:** With improved bandwidth availability, taking the grid to the internet is becoming a reality. However issues' regarding the security and bandwidth utilization needs to be understood as these factors become crucial to the success of the grid as a commercial model. **Results:** Here in this research work we investigate the effect of data intensive applications running in a grid with different number of nodes under constrained bandwidth. **Conclusion/Recommendations:** Our simulation shows that increasing node count does not guarantee improved quality of service.

Key words: Grid computing, quality of service, global grid network, computing environments, data intensive application

INTRODUCTION

Grid computing is gaining popularity in wide areas including academic, research and industrial environments (Mache and Apon, 2007). Grid takes the shape of utilizing the resources that are idle most of the time in a day. The resources that can be part of the Grid are available processor capacity, Storage memory and any hardware or software that can be shared across the globe. The grid is an integrated infrastructure that can play the dual roles of not only a resource consumer but also as a donator in distributed computing environments (Roy and Das, 2009). Service-oriented Grid technologies are increasingly being utilized for the realization of Bioinformatics, Nuclear physics experiments and astronomical computations to name a few for the integration of advanced analysis and simulation applications as well as distributed heterogeneous data sources and information systems (Arbona *et al.*, 2007; Vishwanathan *et al.*, 2007).

Although great advances have been made in the field of Grid computing, QoS remains a major issue as Grid systems cannot be scaled proportionately as expected by the user. A computational grid (Syan and Harnarinesingh, 2010) works in a highly dynamic environment with the resources including bandwidth and processor time availability changing continuously and thus not guaranteeing QoS. Grid applications in a global network also needs to compete for shared

resources which again leads to degradation of QOS (Wang *et al.*, 2009).

MATERIALS AND METHODS

Performance provided by grid service providers is directly related to the collective workload to be executed on a large number of processors scattered globally and on all participating grid sites. Predicting the time required for completing the workload is a very challenging task (AuverGrid Workload Report, 2009; Schopf and Berman, 1998). Scaling the number of processors to complete the collective work load need not be an option as bandwidth also plays a crucial role especially for data intensive workloads where a huge amount of data transfer is involved. In this study we investigate the performance of collective workload with different communication size of task and with different number of distributed processors. This study is organized into the following sections, section II briefly describes issues in grid computing and resource allocation, section III describes our experimental setup and section IV discusses the results obtained.

Issues in grid computing and resource allocation: A Grid solution provider has to take care of the resource discovery, resource selection, resource monitoring, job scheduling, job submission and job monitoring to ensure quality of service. Since Grid itself is virtual and

Corresponding Author: T.R. Srinivasan, Department of IT, Vidyaa Vikas College of Engineering and Technology,
Tiruchengode, India

the resources belong to many different organizations, the above said issues are a hard nut to crack when it comes to implementation (AL-Khateeb *et al.*, 2010).

Mechanisms for auditing and tracing of the end-to-end grid access and transactions are critical and the following process has to be ensured for security:

- Application for tracing and assessing the available resources
- Safe storage and validation of the certificates need to be a continuous process
- Implementation and use of proxy certificates on an end to end basis
- Grid systems should also consider the following issues (Couloris *et al.*, 2001; Tanenbaum and Steen, 2002)
- Heterogeneity
- Transparency
- Fault tolerance and recovery from failure
- Scalability
- Shared access to resources - Concurrency
- Extensibility
- Load balancing
- experimental setup

RESULTS AND DISCUSSION

Experimental setup: In this study we investigate a grid architecture in a global network using available free resources in a global environment. We use the Simgrid toolkit framework for implementing our master/slave simulation. Master distributes tasks to the slave, based on its availability. The primary inputs to the master is the number of tasks to distribute, computation size of each task, communication size of each task and size of the files associated with each task (Boukerram and Azzou, 2006; Poornaselvan *et al.*, 2010). The master also maintains the list of slaves to whom work can be allotted.

In the experimental setup we create a global network with multiple subnets (Rahmat *et al.*, 2010) interconnected using low bandwidth and high bandwidth networks. The total number of tasks and communication size were fixed and only the total number of slaves was varied by removing sub networks.

Three scenarios were created with varying number of nodes and hops. A total of 200 tasks were assigned in each scenario. In the first scenario four subnets consisting of 20 systems were formed with varying bandwidths between the subnets. In the second scenario five subnets consisting of 30 systems were formed and the third scenario consists of 40 systems distributed over 10 sub networks. Figure 1 and 2 shows the experimental test bed.

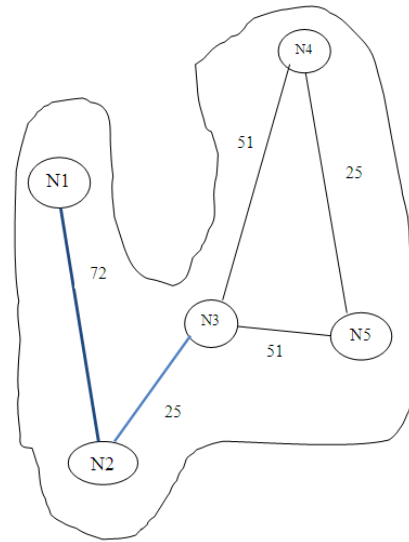


Fig. 1: Experimental test bed for 5 networks with their bandwidth

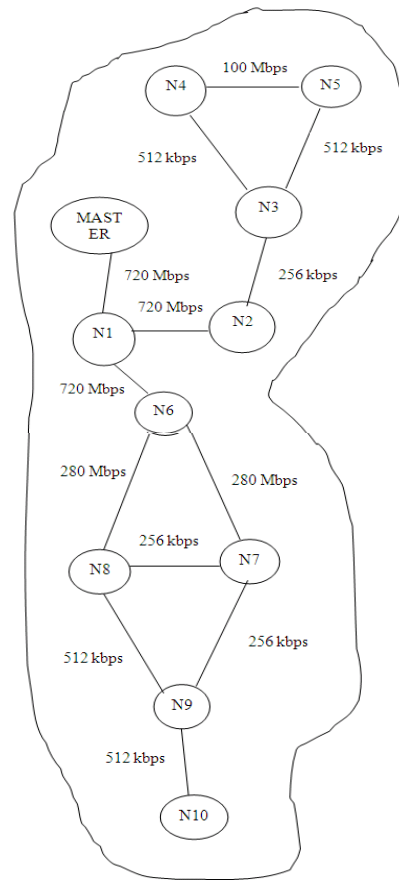


Fig. 2: Experimental test bed for 10 networks with their bandwidth

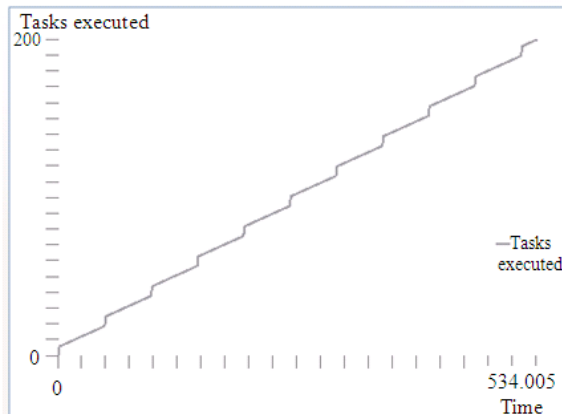


Fig. 3: Time taken by 20 nodes to complete 200 tasks

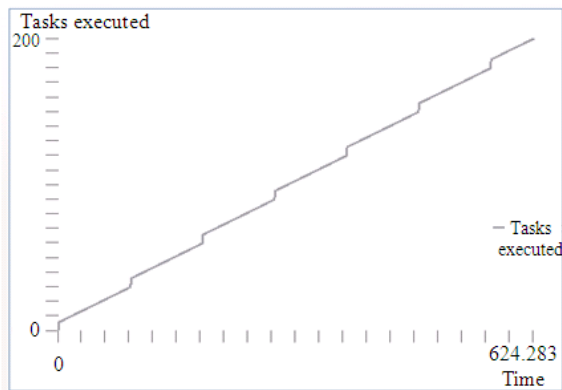


Fig. 4: Time taken by 30 nodes to complete 200 tasks

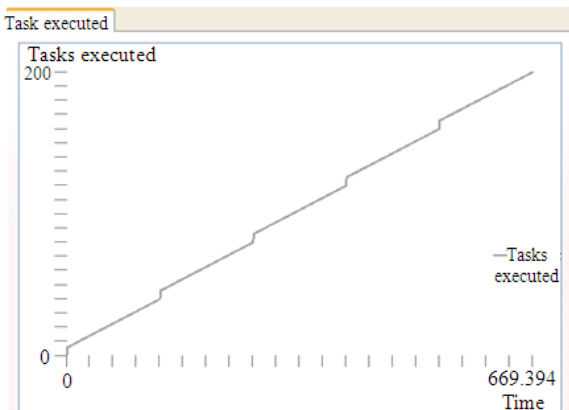


Fig. 5: Time taken by 40 nodes to complete 200 nodes

The simgrid test bed master process created to assign tasks to available slaves in round robin mode is given below (Meligy and Al-Khatib, 2009).

Preliminary

- Initiate message_launch_application
- Calculate number of tasks to distribute
- compute size of each task
- compute size of files associated with the task
- Identify slaves available for assigning task

- Initialize variables to zero for slave_count, Number of task, communication size and computation size

Task creation

- Assign computation size, communication size for the task

Process organization

- Identify slave available
- Assign task to slave
- Intimate assigned task to master
- As slave finishes a task, it is ready to accept next task

- Once all task are completed, inform slaves that computation is complete

The results obtained for the three scenarios is shown in Fig. 3-5.

From the figure it is seen that the computation time for executing 200 tasks increases by 17% when the tasks are distributed over networks with low bandwidth though the available nodes were increased by 50%. Similarly the computation time increases by 25.5 % even though the available nodes are increased by 100% but distributed over ten subnets.

CONCLUSION

From the above investigation it is seen that the timing performance degrades with increased distribution of processor over the network even if the number of tasks that needs to be allocated by the master are low. However if the task is purely compute type then the performance is not affected. Resource allocation algorithms need to learn from past experience. Data mining can be considered to improve the efficiency of resource allocation algorithms based on previous experience.

REFERENCES

AL-Khateeb, A., R. Abdullah and N.A. Rashid, 2010. Job type approach for deciding job scheduling in grid computing systems. Am. J. Applied Sci., 7: 1254-1263. ISSN: 1546-9239

- Arbona, A., S. Benkner, G. Engelbrecht, J. Fingberg and M. Hofmann *et al.*, 2007. A service-oriented grid infrastructure for biomedical data and compute services. *IEEE Trans. NanoBiosci.*, 6: 136-141. PMID: 17695748
- AuverGrid Workload Report, 2009. http://gwa.ewi.tudelft.nl/pmwiki/reports/gwa-t-4/trace_analysis_report.html
- Boukerram, A. and S.A.K. Azzou, 2006. Implementation of load balancing algorithm in a grid computing. *Am. J. Applied Sci.*, 3: 1810-1813. DOI: 10.3844/ajassp.2006.1810.1813
- Couloris, G., J. Dollimore and T. Kinberg, 2001. *Distributed Systems-Concepts and Design*. 4th Edn., Addison-Wesley, Pearson Education, UK., ISBN: 8131718409, pp: 340.
- Mache, J. and A. Apon, 2007. Teaching grid computing: Topics, exercises and experiences. *IEEE Trans. Educ.*, 50: 3-9.
- Meligy, A. and M. Al-Khatib, 2009. A grid-based distributed SVM data mining algorithm. *Eur. J. Sci. Res.*, 27: 313-321. ISSN: 1450-216X
- Poornaselvan, K.J., S. Shanmugasundaram and C.D. Preya, 2010. Application of E-commerce for incentive management in grid ambience. *Eur. J. Sci. Res.*, 45: 484-497. ISSN: 1450-216X
- Rahmat, M.F., M.D. Isa, K. Jusoff, T.A. Hussin and S.M. Rozali, 2010. Image reconstruction algorithm for electrical charge tomography system. *Am. J. Applied Sci.*, 7: 1254-1263. ISSN: 1546-9239
- Roy, N. and S.K. Das, 2009. Enhancing availability of grid computational services to ubiquitous computing applications. *IEEE Trans. Parallel Distributed Syst.*, 20: 953-967.
- Schopf, J. and F. Berman, 1998. Performance prediction in production environments. *Proceedings of the 12th International Parallel Processing Symposium*, Mar. 30-Apr. 3, Orlando, FL, USA., pp: 647-653. DOI: 10.1109/IPPS.1998.669995
- Syan, C.S. and R.E.S. Harnarinesingh, 2010. Comparison of pre-processing and classification techniques for single-trial and multi-trial P300-based brain computer interfaces. *Am. J. Applied Sci.*, 7: 1219-1225. ISSN: 1546-9239
- Tanenbaum, A. and M.V. Steen, 2002. *Distributed Systems: Principles and Paradigms*. 2nd Edn., Prentice Hall, Pearson Education, USA., ISBN: 0132392275, pp: 686.
- Vishwanathan, S., B. Veeravalli and T.G. Robertazzi, 2007. Resource-aware distributed scheduling strategies for large-scale computational cluster/Grid systems. *IEEE Trans. Parallel Distributed Syst.*, 18: 1450-1461.
- Wang, L., G.V. Laszewski and M. Kunze, 2009. Grid virtualization engine: Design, implementation and evaluation. *IEEE Syst. J.*, 3: 477-488. DOI: 10.1109/JSYST.2009.2028589