Performance Analysis of Interconnection Networks for Multi-cluster Systems

Bahman Javadi[1], J.H. Abawajy[2], and Mohammad K. Akbari[1]

[1] Department of Computer Eng. and Information Technology,
Amirkabir University of Technology, Hafez Ave.,Tehran, Iran
{javadi, akbari}@ce.aut.ac.ir
[2] School of Information Technology, Deakin University,
Geelong, VIC 3217, Australia
jemal@deakin.edu.au

Abstract. With the current popularity of cluster computing systems, it is increasingly important to understand the capabilities and potential performance of various interconnection networks. In this paper, we propose an analytical model for studying the capabilities and potential performance of interconnection networks for multi-cluster systems. The model takes into account stochastic quantities as well as network heterogeneity in bandwidth and latency in each cluster. Also, blocking and non-blocking network architecture model is proposed and are used in performance analysis of the system. The model is validated by constructing a set of simulators to simulate different types of clusters, and by comparing the modeled results with the simulated ones.

1 Introduction

Advances in computational and communication technologies has made it economically feasible to conglomerate multiple clusters of heterogeneous networked resources leading to the development of large-scale distributed systems known as multi-cluster systems. Performance analysis and evaluation of multi-cluster systems in general and interconnection networks in particular is needed for understanding system behavior and the analysis of innovative proposals. However, performance analysis in such systems has proven to be a challenging task that requires the innovative performance analysis tools and methods to keep up with the rapid evolution and ever increasing complexity of such systems.

This paper addresses the network interconnects performance analysis problem for multi-cluster computing systems. The motivation for considering this problem is that multi-cluster systems are gaining more importance in practice and a wide variety of parallel applications are being hosted on such systems as well [1]. Also, many recent cluster builders are concerned with two primary factors: cost and performance [17].

While cost is easily determined and compared, performance is more difficult to assess particularly for users who may be new to cluster computing. Moreover, with the current popularity of cluster computing, it is increasingly important to understand the capabilities and potential performance of various network interconnects for cluster

V.S. Sunderam et al. (Eds.): ICCS 2005, LNCS 3516, pp. 205 – 212, 2005.

computing systems [18]. In addition, performance analysis in such systems has proven to be a challenging task that requires the innovative performance analysis tools and methods to keep up with the rapid evolution and ever increasing complexity of such systems [16].

In this paper, we present a new methodology that is based on Jackson network technique to analytically evaluate the performance of network interconnects for multi-cluster systems. The model takes into account stochastic quantities as well as network heterogeneity in bandwidth and latency in each cluster. Bandwidth is the amount of data that can be transmitted over the interconnect hardware in a fixed period of time, while latency is the time to prepare and transmit data from a source node to a destination node. Also, blocking and non-blocking network architecture model is proposed and are used in performance analysis of the system. The message latency is used as the primary performance metric. The model is validated by constructing a set of simulators to simulate different types of clusters, and by comparing the modeled results with the simulated ones.

The rest of the paper is organized as follows. In Section 2, related work is discussed. In Section 3, we describe the proposed analytical model. We present the model validation experiments, in Section 4. Finally, Section 5 summarizes our findings and concludes the paper.

2 Related Work

Generally, multi-cluster systems can be classified into *Super-Cluster* and *Cluster-of-Cluster*. A good example of Super-Cluster systems is DAS-2 [5], which is characterized by large number of homogenous processors and heterogeneity in communication networks. In contrast, Cluster-of-Clusters are constructed by interconnecting multiple single cluster systems thus heterogeneity may be observed in communication networks as well as processors. The LLNL multi-cluster system which is built in by interconnecting of four single clusters, MCR, ALC, Thunder, and PVC [6] is an example of cluster-of-cluster system. In this paper, we will focus our discursion on the Super-Cluster system with homogenous processors and heterogeneous communication networks.

Currently, there are three possible ways to address this problem – simulation, prediction and analytical modeling. The limitations of simulation-based solutions are that it is highly time-consuming and expensive. Similarly, techniques based on predictions from measurements on existing clusters would be impractical. An alternative to simulation and prediction approaches is an analytical model, which is the focus of this paper. An accurate analytical model can provide quick performance estimates and will be a valuable design tool. However, there is very little research addressing analytical model for interconnects in multi-cluster systems. The few results that exist are based on homogenous cluster systems and the evaluations are confined to a single cluster [2, 3, 4, 19]. With all probability, multiple cluster systems would be configured from heterogeneous components, rendering exiting optimization solutions unusable in heterogeneous multi-cluster environment. In contrast, our work focuses on heterogeneous multi-cluster computing systems. To this end, we present a

generic model to analytically evaluate the performance of multi-cluster systems. We believe that our work is the first to deal with heterogeneous multi-cluster environments.

3 Proposed Analytical Model

The architectural model of the system assumed in this paper similar to [12]. It is made up of C clusters, each cluster i is composed of N_i processors of type T_i, i=1,…,C. Also, each cluster has two communication networks, an Intra-Communication Network ($ICN1_i$), which is used for the purpose of message passing between processors, and an intEr-Communication Network ($ECN1_i$), which is used to transmit messages between clusters, management and also for the expansion of system. Note that, ECN can be accessed directly by the processors of a cluster without going through the ICN.

The proposed model is based on the following assumptions that are widely used in similar study [3, 4, 12, 14]:

1. Each processor generates packets independently which follows a Poisson process with a mean rate of λ and inter-arrival times are exponentially distributed.
2. The arrival process at a given communication network is approximated by an independent Poisson process. This approximation has often been invoked to determine the arrival process in store-and-forward networks [13]. In this paper we apply the store-and-forward network, e.g., Ethernet-based networks. Therefore, the rate of process arrival at a communication network can be calculated using Jackson's queuing networks formula [7].
3. Each processor granted the network as a packet transmission.
4. The destination of each request would be any node in the system with uniform distribution.
5. The processors which are source of request must be waiting until they get service and they cannot generate any other request in wait state.
6. The number of processors in all clusters are equal ($N_1=N_2=\ldots=N_C=N_0$) with homogenous type of ($T_1=T_2=\ldots T_C=T_0$).
7. Message length is fixed and equal to *M* bytes.

A packet is never lost in the network. Also, the terms "request" and "packet" are used interchangeably throughout this paper.

3.1 Queuing Network Model

Based on the characteristics of the system model, each communication network can be considered as service center. The queuing network model of system is shown in Fig. 1, where the path of a packet through various queuing centers is illustrated. As is

shown in the model, the processor requests will be directed to service center ICN1 and ECN1 by probability *1-P* and *P*, respectively. According to assumption 1, the request rate of a processor is λ, so the input rate of ICN1 and ECN1 which feed from that processor will be $\lambda(1-P)$ and $\lambda.P$, respectively. The additional inputs at these service centers, γ_{I1} and γ_{E1}, are due to the requests generated by other processors of the same cluster. The output of ICN1 is feedback to the same processor, and also ε_{I1} represents the response to other processors in the same cluster.

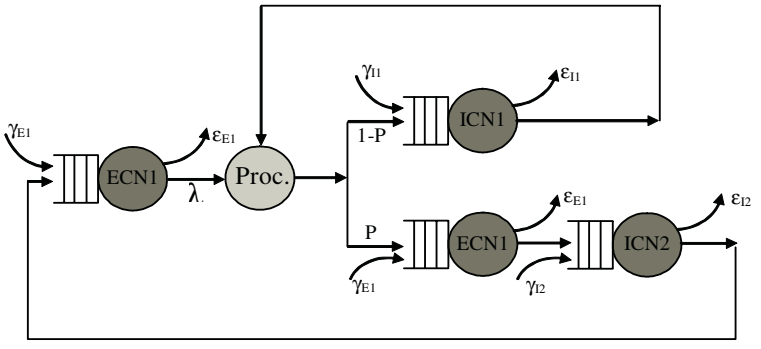

Fig. 1. Queuing Network Model of a Super-Cluster System

The external request (out of cluster) of a cluster goes through the ECN1 with probability *P* and then ICN2. In the return path, it again accesses the ECN1 to get back to the node, which initiated the request. As mentioned before, ε_{E1} and ε_{I2} are responses to the other requests except the one under consideration. So, the total requests of the processors received by service centers in the first stage can be calculated as follows:

$$\lambda_{I1} = (1- P)\lambda + \gamma_{I1} = (1-P)\lambda + (N_0-1)(1-P)\lambda$$
$$= N_0 (1-P)\lambda \quad (1)$$

$$\lambda_{E1(1)} = P\lambda + \gamma_{E1} = P\lambda + (N_0-1)P\lambda$$
$$= N_0 P\lambda \quad (2)$$

where $\lambda_{E1(1)}$ is the input rate of ECN1, the one which is feed by the processor.

In the second stage, the input request rate of ICN2 in forward path and ECN1in feedback path can be computed by following equations:

$$\lambda_{I2} = \lambda_{E1(1)} + \gamma_{I2} = N_0 P\lambda + (C-1)\lambda_{E1(1)}$$
$$= N_0 P\lambda + (C-1)\, N_0 P\lambda = C\, N_0 P\lambda \quad (3)$$

$$\lambda_{E1(2)} = \lambda_{I2}/C = N_0 P\lambda \quad (4)$$

where $\lambda_{E1(2)}$ is the input rate of ECN1 from feedback path. According to equations (2) and (4), the input rate of ECN1 is:

$$\lambda_{E1} = \lambda_{E1(1)} + \lambda_{E1(2)} = 2N_0 P\lambda \quad (5)$$

The average number of waiting processors in each service center can be computed through queue length of each center. So, the average of total waiting processors in the system will be:

$$L = C\,(2.L_{E1} + L_{I1}) + L_{I2} \tag{6}$$

which L is denoting the queue length of each service center. As mentioned in the assumption 5, the waiting processors would not be able to generate new requests, so the effective request rate of the processor would be less than λ. Applying the method described in [8] to find the effective request rate of a processor, it is directly dependent to the ratio of number of active processors to total number of processors. Therefore, L and λ are computed iteratively based on following equation, until no considerable change is observed between two consecutive steps:

$$\lambda_{eff} = \frac{N-L}{N} \times \lambda \tag{7}$$

As it can be seen in the previous equations, the probability P has been used as the probability of outgoing request within a cluster. According to assumption 4, this parameter is computed base on structure of system [20] by the following equation:

$$P = \frac{(C-1) \times N_0}{(C \times N_0) - 1} \tag{8}$$

In this paper, message latency is selected as a primary performance metric. However, most of the other performance metrics for the queuing network model of a multi-cluster system are related to the message latency with simple equations [12]. To model the mean message latency, we consider effective parameters as follows. In such systems, the mean network latency, that is the time to cross the network, is the most important part of the message latency. Other parameters such as protocol latency can be negligible.

Since the system under study is symmetric, averaging the network latencies seen by message generated by only one node for all other nodes gives the mean message latency in the network. Let S be the source node and D denotes a destination node such that $D \; \varepsilon \; A\text{-}\{S\}$ where A is the set of all nodes in the network. The network latency, T_C , seen by the message crossing from node S to node D consist of two parts: one is the delay due to the physical message transmission time, T_W, and the other is due to the blocking time in the network, T_B. Therefore, T_C can be written as:

$$T_C = T_W + T_B \tag{9}$$

These parameters are strongly depended on the characteristics of the communication network which is used in the system. Of this, we take into account two different networks in our model as following.

3.2 Blocking and Non-blocking Network Model

For non-blocking architecture, we use the *Multi-Stage Fat-Tree* topology, which is used in some cluster systems such as Thunder [9]. For modeling blocking interconnect architecture, a *Linear Array* of switches is used. Due to space limitation,

we have not included the details of blocking and non-blocking analysis here. Interested reader can refer to [20].

4 Performance Evaluation

In order to validate the technique and justify the approximations, the model was simulated using the OMNeT++ [15]. Requests are generated randomly by each processor with an exponential distribution of inter-arrival time with a mean of $1/\lambda$ where is λ is fixed to 0.25 msg/sec in all experiments. The destination node is determined by using a uniform random number generator. Each packet is time-stamped after its generation. The request completion time is checked in to compute the message latency in a "*sink*" module. For each simulation experiment, statistics were gathered for a total number of 10,000 messages. In our study, we used two well-known network technologies, Gigabit Ethernet (GE) and Fast Ethernet (FE), which are widely used in cluster systems. We also used the same value for the latency and bandwidth of each network as reported by [10]. Two different communication network scenarios for network heterogeneity were simulated. However, due to space limitations, a subset of the results is presented here.

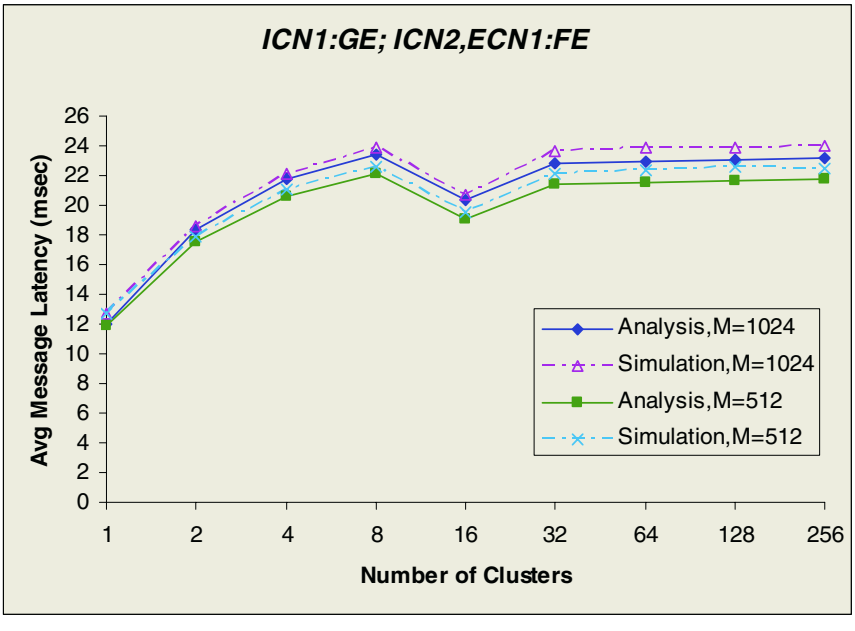

Fig. 2. Average Message Latency vs. Number of Clusters for Non-blocking Networks

Fig. 2 shows the average message latency in a multi-cluster system with *N=256* nodes and non-blocking communication network against those provided by the simulator for the message size of 1024 and 512 bytes. The horizontal axis in the

figures represents the number of clusters in the system. To have a better performance analysis of the system, we used the blocking communication networks, with the same parameters. As we expected, the average message latency in this system is much larger than in the system with non-blocking networks. These results reveal that average message latency in blocking network with uniform traffic pattern is 1.4 to 3.1 times larger than non-blocking network.

5 Conclusion and Future Directions

A performance model is an essential tool for behavior prediction of a system. It is used to analyze intricate details of the system and various design optimization issues. One such model based on queuing networks is presented in this study to predict the message latency of multi-cluster systems. Two different networks, blocking and non-blocking, were used in our modeling of the system. The analysis captures the effect of communication network architecture on the system performance. The model is validated by constructing a set of simulators to simulate different types of clusters, and by comparing the modeled results with the simulated ones. The future works focus on improving the analytical model to tack into account more effective parameters, modeling of communication networks with technology heterogeneity and propose a similar model to another class of multi-cluster systems, Cluster-of-Clusters.

References

1. J. H. Abawajy and S. P. Dandamudi, "Parallel Job Scheduling on Multi-Cluster Computing Systems," In P*roceedings of the IEEE international Conference on Cluster Computing (CLUSTER'03)*, Dec. 1-4, 2003, Hong Kong.
2. X. Du, X. Zhang, Z. Zhu, "Memory Hierarchy Consideration for Cost-Effective Cluster Computing," *IEEE Transaction on Computers*, Vol. 49, No.5, pp. 915-933, Sept. 2000.
3. B. Javadi, S. Khorsandi, and M. K. Akbari, "Study of Cluster-based Parallel Systems using Analytical Modeling and Simulation" *International Conference on Computer Science and its Applications (ICCSA 2004)*, May 2004, Perugia, Italy.
4. B. Javadi, S. Khorsandi, and M. K. Akbari, "Queuing Network Modeling of a Cluster-based Parallel Systems", *7th International Conference on High Performance Computing and Grids (HPC ASIA 2004)*, July 2004, Tokyo, Japan.
5. The DAS-2 Supercomputer. http://www.cs.vu.nl/das2
6. B. Boas, "Storage on the Lunatic Fringe", Lawrence Livermore National Laboratory, Panel at SC2003, Nov. 2003, Arizona, USA.
7. D. Bertsekas, R. Gallager., *Data Networks*, Prentice Hall Publishers, New Jersey, 1992.
8. H. S. Shahhoseini, M. Naderi, "Design Trade off on Shared Memory Clustered Massively Parallel Processing Systems", *The 10th International Conference on Computing and Information (ICCI '2000),* Nov. 2000, Kuwait.
9. "Thunder Statement of Work", University of California, Lawrence Livermore National Laboratory, Sept. 2003.
10. M. Lobosco, and L. de Amorim, "Performance Evaluation of Fast Ethernet, Giganet and Myrinet on a Cluster", *Lecture Notes in Computer Science*, volume 2329, pp. 296-305, 2002.

11. J. H. Abawajy, "Taxonomy of Job Scheduling Approaches in Cluster Computing Systems," Technical Report, Deakin University, 2004.
12. B. Javadi, M. K. Akbari, J.H. Abawajy, "Performance Analysis of Multi-Cluster Systems Using Analytical Modeling", *International Conference on Modeling, Simulation and Applied Optimization*, Sharjah, United Arab Emirates, Feb. 2005.
13. L. Kleinrock, Queuing System: Computer Applications, Part 2, John Wily Publisher, New York, 1975.
14. H. Sarbazi-Azad, A. Khonsari, M. Ould-Khaoua, "Analysis of k-ary n-cubes with Dimension-order Routing", *Journal of Future Generation Computer Systems*, pp. 493-502, 2003.
15. Nicky van Foreest. Simulation Queuing Networks with OMNet++, in Tutorial of OMNnet++ Simulator, Department of Telecommunications, Budapest University of Technology and Economics, Apr. 2002.
16. J. H. Abawajy, "Dynamic Parallel Job Scheduling in Multi-cluster Computing Systems," *4th International Conference on Computational Science*, Kraków, Poland, pp. 27-34, 2004.
17. Chee Shin Yeo, Rajkumar Buyya, Hossein Pourreza, Rasit Eskicioglu, Peter Graham, Frank Sommers, "Cluster Computing: High-Performance, High-Availability, and High-Throughput Processing on a Network of Computers", Handbook of Innovative Computing, Albert Zomaya (editor), Springer Verlag, 2005.
18. H. Chen, P. Wyckoff, and K. Moor, "Cost/Performance Evaluation of Gigabit Ethernet and Myrinet as Cluster Interconnects," *Proc. 2000 Conference on Network and Application Performance (OPNETWORK 2000)*, Washington, USA, Aug. 2000.
19. J. Hsieh, T. Leng, V. Mashayekhi, and R. Rooholamini, "Architectural and Performance Evaluation of GigaNet and Myrinet Interconnects on Clusters of Small-Scale SMP Servers," *Proc. 2000 ACM/IEEE conference on Supercomputing (SC2000)*, Dallas, USA, Nov. 2000.
20. Bahman Javadi, J. H. Abawajy, Mohammad K. Akbari, "Performance Analysis of Interconnection Networks for Multi-Cluster Systems," Technical paper, School of Information Technology, Deakin University, Geelong, VIC 3217, Australia, 2005.