An Accurate Communication Model of a Heterogeneous Cluster Based on a Switch-Enabled Ethernet Network

Alexey Lastovetsky alexey.lastovetsky@ucd.ie
Is-Haka Mkwawa ishaka.mkwawa@ucd.ie
Maureen O'Flynn maureen.oflynn@ucd.ie

School of Computer Science and Informatics, University College Dublin, Belfield, Dublin 4, Ireland

1 Abstract

The paper presents a communication model of a set of heterogeneous processors interconnected via a switch-enabled Ethernet network. The goal of the model is to accurately predict the contribution of communication operations into the total execution time of parallel applications running on the platform. The presented model takes into account the impact of the heterogeneity of processors on the performance of communication operations. In this paper, we give analytical models for a single point-to-point communication, multiple independent point-to-point communications, multiple one-to-many point-to-point communications, and for a broadcast. Experimental results are presented demonstrating the accuracy of the analytical models.

2 Introduction

In this paper we aim at an accurate performance model of communications of parallel applications on a heterogeneous cluster. The ultimate purpose of the model is to predict the execution time of any given combination of communication operations. In the paper, the model is built step-by-step from a single point-to-point communication to a combination of point-to-point communications and collective communication operations. We look at the standard MPI operations on a heterogeneous platform of workstations with different processors and operating systems with each node running MPI.

Research in this area has its origins in parallel computing on large distributed memory parallel machines with many identical processors and can make the assumption of a homogeneous environment. In recent times the requirements of heterogeneity reflect the evolution of parallel processing from dedicated parallel computing to the harnessing of power of the networks of computers today. Shared Memory Multiprocessor architectures [6] consisting of a number of identical processors have a memory bus connection rather than a network. The PRAM model [7] can be used for this architecture but is unrealistic for a cluster of workstations such as ours as it assumes that all processors work synchronously and that there are no communication costs between processors. The bulk-synchronous parallel model (BSP) and BSP-like models [4, 12, 10] allow for asynchronous processors and also for latency and bandwidth but they have restrictive programming methodologies. Culler et al [3] use the model as a starting point for their LogP model. This model is based on four simple parameters representing aspects of point-to-point communications. The model assumes homogeneous processor and communication conditions for small message sizes, and can be applied to communication operations as a simple summation of inter-processor communications. Many adaptations of this model were presented to extend it, such as LogGP [1] to account for long messages and LogGPC [9] to account for network contention delay. None of these models address the issues due to heterogeneous conditions of a network such as those of different processors, operating systems or network idolization. Zhang and Yan [14] present models for heterogeneity for a network of workstations (NOW) with general definitions of a network. The HBSP model [13] by Williams supports a combination of parallel machines and machine clusters, their experiments show increased performance on heterogeneous platforms, but with the restrictions of following a synchronous programming methodology. Kielmann et al [5] studies the LogP model for collective communications with a computational grid application over a geographically distributed system. New directions considering processor involvement in task scheduling were proposed by Sinnen and Sousa [11], making the point that processor heterogeneity can be characterized by the links for sending and receiving communications.

Models are also proposed by Casanova et al [8] for

large-scale WAN platforms for scheduling for heterogeneous networks. Network latency, bandwidth sharing and network topology are all new areas that are addressed. Their experimental work indicates the possibilities of further exploration to take into account processor variation or local bandwidth fluctuation.

In this paper we describe a model for point-to-point heterogeneity and then extend it for collective communication operations. We seek to have a simple and adaptable model that is independent of processor power, operating system or application type. It is our strategic goal to extend this model to all types of network to accurately predict execution times of all types of communication operations and applications. In a similar approach to Beaumont et al [2] with examinations of scheduling problems for heterogeneous clusters, we make assumptions of dual communication over single port communications with a switched Ethernet topology. We look at point-to-point and collective communications operations in particular. The operations are examined to simplify them to the serialization of communications, and represented algebraically. Our model is built from a few parameters that can be found experimentally with little performance costs. These are important considerations when exploring the practical viability of the model.

The rest of the paper is structured as follows. In Section 3 we describe the model and give formulas for both some point-to-point and collective communication operations. Section 4 describes experimental work, the applications, network and equipment details. Section 5 presents results and deductions made from the experiments. Finally we conclude in Section 6, with suggestions for further studies.

3 Analytical Models

This section proposes parallel communication models for different scenarios in a heterogeneous environment. We start with a single point-to-point communication and then extend this to different types of combined point-to-point communications and a collective communication operation, broadcast. This analysis can then be examined by our experiments to show the validity of our approach in subsequent sections.

3.1 Point to Point Communication

The total communication time T experienced by a message of size M is composed of the transmission delay, T_{net}, that is the time taken by a message of size M to cross the network, the source node delay, L_i and the destination node delay L_j. The transmission delay is the time lapsed between the first bit transmitted and the last bit captured at the destination node. In this model the propagation delay is ignored because of the network nature. The source node delay is due to fixed processing and variable processing delays. Fixed processing delay includes message headers construction while that of variable processing is made up of message copying, hence dependent on message size M.

The following equations summarize the above explanation;

$$T = L_i + L_j + T_{net} \tag{1}$$

$$L_i(M) = C_i + t_i M_i \tag{2}$$

where C_i, t_i and M_i is the fixed precessing delay, time needed to process a byte and message size at source node i, respectively. The term $t_i M_i$ is the variable processing delay at source node i.

$$L_j(M) = C_j + t_j M_{ij} \tag{3}$$

Equation 3 is similar to Equation 2 with the exception that these terms are at destination node j. $M_{ij} > M_i$ is the message size at the destination node (M_{ij} is bigger due to headers).

$$T_{net}(M) = M_{ij}/\beta ij \tag{4}$$

where β_{ij} is the transmission rate of the link connecting source node i and destination node j.

It can easily be shown that the difference between Mi and M_{ij} is very minimal, hence it is assumed that $Mi \approx M_{ij} = M$.

Since a node is capable of receiving and sending messages, one should associate it with C_i and C_j, but empirical results show that their difference is very minimal and can be ignored in order to have a good model with fewest possible number of parameters, but still can capture the complexity of a system under investigation.

Therefore the communication time for a point to point communication in a heterogeneous environment from node i to node j is given by;

$$T_{ij}(M) = C_i + t_i M + C_j + t_j M + M/\beta ij \tag{5}$$

3.2 One to Many Communication

For an Ethernet switched cluster of nodes, each node can directly communicate to every other node. One to many communication is the process where by a source node disseminates the same/different message to an arbitrarily number of nodes $n \leq N$, where N is the cluster size. The source

node delay is much bigger due to large data size destined for other n nodes, $\sum_{j=1}^{n} M_j$. By the nature of a standard MPI communication implementation, for small messages (in our case ($\leq$1KB), the source node must necessarily not wait before it can start sending the next message. Therefore, the source node can start sending next message as soon as the first byte of the current message has reached the destination node. For large message size (>1KB), the source node can wait before sending the next message. In this context, for small messages, one to many communication can be seen as parallel communication scheme while that of large message as a serialized communication scheme. This type of communication is modelled as a flat tree and hence the communication time is given by;

$$C_0 + t_0 \sum_{j=1}^{n} M_j + \begin{cases} \sum_{j=1}^{n} (C_j + t_j M_j + M_j/\beta_{0j}) & M_j > S \\ \max_{n} \left\{ C_j + t_j M_j + M_j/\beta_{0j} \right\} & M_j \leq S \end{cases} \quad (6)$$

where C_0 and t_0 are parameters at the source node 0, S is a message size threshold (in our case is $1KB$) categorizing small and large message size. S can vary with a cluster in use.

3.3 Multiple Point to Point Communication

This type of communication occurs when several disjoint pairs in a cluster start communicating in a point to point communication fashion of the same message at the same time.

As this is running under a full duplex Ethernet switch, disjoint pairs of point to point communication will be independent to each other and hence the communication time for a set K of disjoint ordered pairs, multiple point to point communications is given by;

$$\max_{K} \{ T_{ij}(M) \} \quad (7)$$

In this communication type, there is no traffic interference, the full duplex Ethernet switch has made this possible.

3.4 Broadcast Communication

In this type of communication, a source node sends the same message to all other nodes in the network including itself. MPI_BCAST Library implements a binomial tree for broadcast process and hence the proposed model will take into consideration the characteristics and the properties of binomial trees. The proposed model has two more parameters due to the overhead contributed by the broadcast process. The broadcast overhead includes, broadcasting message to itself (source node), MPI_BCAST Library, physical message copying and routing operations.

Since there are $\lfloor N/2 \rfloor$ leaf nodes in a binomial, the number of paths from the source node to these leaf nodes is $\lfloor N/2 \rfloor$. Let $\Pi = \{\pi_1^{\Omega_1}, \pi_2^{\Omega_2}, \pi_3^{\Omega_3}, \ldots, \pi_\omega^{\Omega_\omega}\}$ be a set of these paths, where $\omega = \lfloor N/2 \rfloor$. Each path π_k has its total point to point communication time $\Omega_k = \sum_{i=0}^{P} T_{ij}(M)$, where $j = i + 1$ and $1 \leq P \leq log_2 N$. Therefore the broadcast communication time is given by,

$$\max_{\Pi} \left\{ \pi_1^{\Omega_1}, \pi_2^{\Omega_2}, \pi_3^{\Omega_3}, \ldots, \pi_\omega^{\Omega_\omega} \right\} + \Gamma + \tau M \quad (8)$$

where Γ is the fixed broadcast overhead and τ is the delay per a message byte of the broadcast overhead.

4 Experiment Setup

The experiments were done on a dedicated heterogeneous cluster in the School of Computer Science and Informatics, University College Dublin (UCD). The cluster is made up of 16 nodes, IBM x306 3.0GHz AMD Processor, two IBM x326 2.2GHz AMD Processors, two Dell PowerEdge SC1425 Xeon processors, 3.0GHz and 2.2GHz. 6 Dell PE750 Pentium 3.4GHz processors. Three HP DL140 Xeon Processors, 2.8GHz,3.4GHz and 3.6GHz. Two HP DL320 Celeron 2.9GHz and 3.4GHz Pentium 4 Processors. Eight nodes are running Fedora Core 4, six nodes have Debian Linux installed, one is running Solaris and the last one is running HP-UX.

The cluster is connected via a Cisco Catalyst 3560 Gigabit Ethernet switch with adjustable bandwidth (from few Kilobytes) on each link.

4.1 Model Parameters

Model parameters were measured using LAM/MPI. The t_i, t_j, C_i and C_j parameters are simply obtained by a ping pong communication. Two homogeneous nodes are set for the experiment and the parameters are calculated by solving simultaneous equations. The two parameters (i.e., C and t) for each node define the characteristics of a particular node, this parameters are fixed unless one or all of the following are changed; operating system, hardware and communication software. These parameters are the building blocks for all other communication operations.

As for broadcast communication, $\Gamma + \tau M$ is obtained by deducting experimental values for a particular message

size to $\max_{\Pi}\left\{\pi_1^{\Omega_1}, \pi_2^{\Omega_2}, \pi_3^{\Omega_3}, \ldots, \pi_\omega^{\Omega_\omega}\right\}$. Then Γ and τM are calculated by simultaneous equation.

5 Experimental Results

Due to limited space, each communication model discussed above is validated against experimental results but only few graphs are shown although the whole experimental setup was done in a cluster of 16 nodes.

Point to Point Communication: Figure 1 shows the comparison between the proposed point to point communication model and the experimental results. Two heterogeneous nodes (hcl10 and hcl15) are selected from the cluster, hcl10 is IBM E-Server 1.8GHz AMD opteron processor and hcl15 is HP Proliant DL140 2.8GHz Xeon Processor, both running Linux Debian.

With GigaEthernet network infrastructure and powerful processing nodes, the communication time for small messages is very small and the difference between them is very minimal.

From 1KB of a message size, the graph shows a sharp increase in communication time. This is due to the point that MPI standard communication requires message buffering. This can be a threshold where the model would have to have two simultaneous equations.

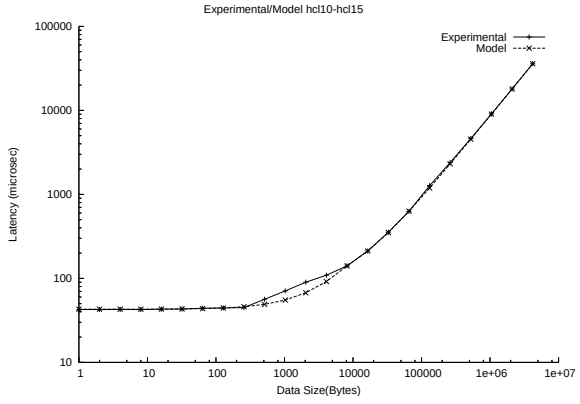

Figure 1. Point to point communication time

Broadcast Communication: Figure 2 shows the comparison between the experimental data and the predicated data obtained from Equation 8. The proposed model for broadcast communication gives a description very close to the experimental data obtained by MPI_BCAST.

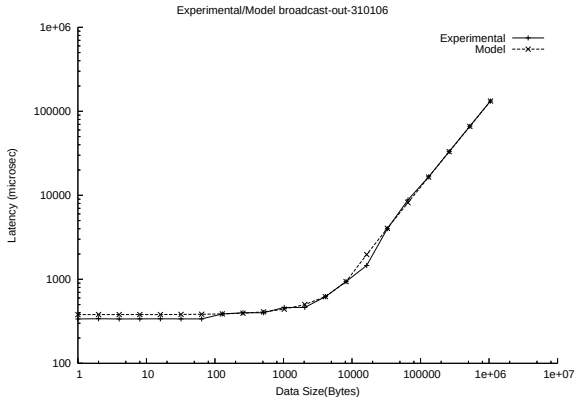

Figure 2. Broadcast communication time

To use of binomial tree in a cluster, the nodes were arranged as described in LAM/MPI. They were arranged according to their nodes rank numbers, the implementation was top bottom and right left.

One to Many Communication: As per one to many communication time, the data obtained from model in Equation 6 is compared to experimental data, the results are very close and are depicted in Figure 3. For small messages i.e., $\leq$1KB the second part of Equation 6 has been used and the rest has used the first part of the equation.

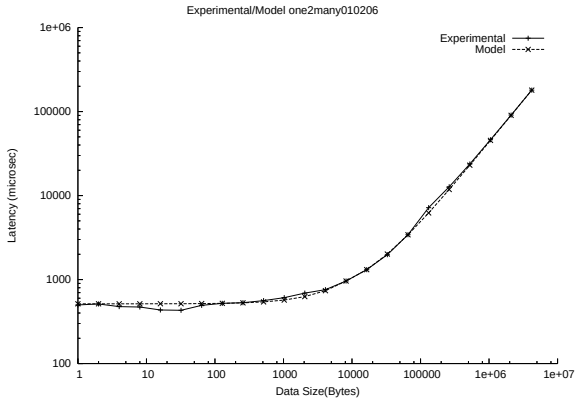

Figure 3. One to many communication time

It has been shown that the difference in the experimental and model communication time when using either part of Equation 6 is an average of 12% compared to an average of 5% when using both parts appropriately. This validates the claim of branching the equation for M_j $\leq$1KB

and M_j >1KB.

Multiple Point to Point Communication: The model for multiple point to point communication time in Equation 7 is validated against the experimental data, results are accurate and are shown in Figure 4.

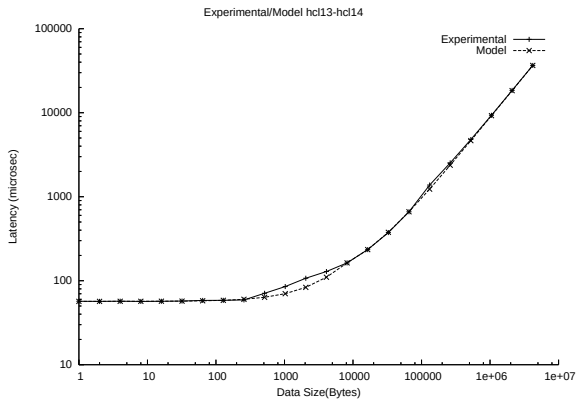

Figure 4. Multiple point to point communication time

One to Many Vs Broadcast Communication: As a point of interest, if $n = N$ in Equation 6 and the source node disseminates the same message to itself, the one to many communication model becomes a broadcast problem and outperforms the MPI_BCAST for bigger message sizes. This can be explained due to the fact that the construction of a binomial tree in LAM/MPI is not optimized and the overheads incurred due to MPI_BCAST library. The differences are depicted in the Figure 5.

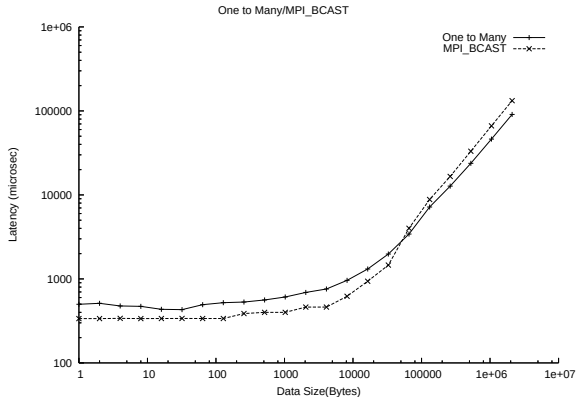

Figure 5. One to many communication model Vs MPI_BCAST

6 Conclusion

In this paper we have considered the challenge of heterogeneity for a switched Ethernet cluster with a variety of processor power and operating systems. We presented an accurate performance model for communications on a heterogeneous switch-enabled Ethernet network. We have examined the problem of collective communications, and propose a new model that can be adapted to represent different forms of communication on parallel systems in a simple and practical way.

Experiments show how our model for point to point, one to many, many to one and many to many type communications yield comparative evaluation and measurement with graphical displays. The figures show that the model is a close representation of actual communications behavior. The model can then be used to estimate the broadcast operation, as demonstrated by experimental results.

We would like to continue experimental work to further verify our findings in this area. We wish to represent other collective communication operations. The ultimate potential of the model can be extended to apply to a more general network topology such as LANs and WANs. The theoretical basis for this new direction is promising, and it is hoped to be confirmed in further work.

References

[1] A. Alexandrov, M. Ionescu, K.E.Schauser, and C. Scheiman. Loggp: Incorporating long messages into the logp model — one step closer towards a realistic model for parallel computation. *Technical Report: TRCS95-09, University of California at Santa Barbara Santa Barbara, CA, USA*, 1995.

[2] O. Beaumont, A. Legrand, and Y. Robert. The master-slave paradigm with heterogeneous processors. *3rd IEEE International Conference on Cluster Computing (CLUSTER'01)*, page 419, 2001.

[3] D. Culler, R. Karp, D. Patterson, A. Sahay, K. E. Schauser, R. S. E. Santos, and T. von Eicken. Logp: Towards a realistic model of parallel computation. *Proceedings of the Fourth ACM SIGPLAN Symposium on Principles and Practice of Parallel Programming, San Diego, CA*, May 1993.

[4] B. Juurlink and H. Wijshoff. A qualitative comparison of parallel computation models. *ACM Transactions on Computer Systems (TOCS)*, 16(3):271–318, August 1998.

[5] T. Kielmann, H. E. Bal, and K. Verstoep. Fast measurement of logp parameters for message passing platforms. *Proceedings of the Parallel and Distributed Processing IPDPS 2000 Workshops, Cancun, Mexico*, 15:1176, May 2000.

[6] A. Lastovestsky. *Parallel Computing on Heterogeneous Networks*. Wiley Series on Parallel and Distributed Computing, Wiley, 2003.

[7] B. Maggs, L. Matheson, and R. Tarjan. Model of parallel computation: A survey and synthesis. *Proceedings of the 28th Hawaii International Conference on System Sciences (HICSS)*, 2:61–70, January 1995.

[8] L. Marchal, Y. Yang, H. Casanova, and Y. Robert. A realistic network/application model for scheduling divisible loads on large-scale platforms. *19th IEEE International Parallel and Distributed Processing Symposium (IPDPS'05)*, page 48b, 2005.

[9] C. Moritz and M. Frank. Logpc: Modeling network contention in message-passing programs. *IEEE Transactions on Parallel and Distributed Systems*, 12(4):1–100, April 2001.

[10] J. Sineyn and M. Kaufmann. Bsp-like external-memory computation. *Proceedings of the Third Italian Conference on Algorithms and Complexity, ACM Lecture Notes In Computer Science*, 1203:229–240, 1997.

[11] Sinnen and L. Sousa. Task scheduling: Considering the processor involvement in communication. *Proc. 3rd Int. Workshop on Algorithms, Models and Tools for Parallel Computing on Heterogeneous Networks (ISPDC'04/HetroPar'04) IEEE Press, Cork, Ireland*, pages 328–335, July 2004.

[12] L. Valiant. A bridging model for parallel computation. *Communications of the ACM, ACM*, 33(8):103–111, August 1990.

[13] T. Williams and R. Parsons. Exploiting hierarchy in heterogeneous environments. *Proceedings of the 15th International Parallel and Distributed Processing Symposium (IPDPS-01), San Francisco, CA*, page 140, April 2001.

[14] X. Zhang and Y. Yan. Modeling and characterizing parallel computing performance on heterogeneous networks of workstatioedings of the 7th ieee symposium on parallel and distributed processing (spdp'95). *Proceedings of the 7th IEEE Symposium on Parallel and Distributed Processing (SPDP'95)*, October 1995.