Modeling Performance of Many-to-One Collective Communication Operations in Heterogeneous Clusters

Alexey Lastovetsky
Is-Haka Imkwawa
Maureen O'Flynn
School of Computer Science and Informatics
University College Dublin
Belfield, Dublin 4, Ireland
email: maureen.oflynn@ucd.ie
Technical Report UCD-CSI-2006-4

30th May 2006

ABSTRACT
This paper presents a performance model for Many-to-One type communications on a dedicated heterogeneous cluster of workstations based on a switched Ethernet network. This study finds that Many-to-One communication is more complex than One-to-Many and Point-to-Point communications as it does not show a linear or even continuous dependence of the execution time on message sizes. It displays a very high jump in execution time for a significant range of message sizes. As a result, the proposed model is divided into three parts. The first part is for small sized messages whose model is linear, the second part models the congestion region, and the last part is for large message sizes where linearity resumes. The proposed model is validated for accuracy by the experiments on various platforms with different MPI implementations.

KEY WORDS
Performance, Collective Communication, Heterogeneous Networks, MPI.

1 Introduction

In our previous work [8] we proposed an accurate model for Point-to-Point communication which was used as a building block to model One-to-Many, multiple independent Point-to-Point and Broadcast communication models for a dedicated heterogenous cluster based on a switched-enabled Ethernet network. Due to the nature of a switched network, there was no traffic interference between any communicating pair of nodes in our previous models and hence no congestion occurred. In contrast, in Many-to-One communication type congestion occurs at the receiving node. This behaviour make it more complex than other type of collective communication operations and hence extra parameters needs to be introduced as well as those of the previous proposed Point-to-Point communication model.

Numerous tests on various MPI implementations and platforms showed the persistent existence of a continuous non-linear behavior for a significant range of message sizes (c.f. Fig.1). The experiments use the same message sizes for all sending nodes to make the model simple and easy to implement.

The rest of the paper is organised as follows. Section 2 gives an overview of the related work in the field. Section 3 summarises our previous work on collective communication models. Section 4 describes the new propose Many-to-One communication model. Experimental setup and results follows in Section 5 and 6, respectively. We conclude in Section 7.

2 Related Work

Previous research for performance measurement such as the PRAM model [7, 9] found execution time estimations for Shared Memory Multiprocessor architectures. It is not suited to a heterogeneous network environment as there are extra contingencies such as processor variation and network issues. The bulk-synchronous parallel model (BSP) and BSP-like models [9, 11, 4] allow for the asynchronous network influences such as latency and bandwidth, but they depend on restrictive programming methodologies. Culler et al [4] present the LogP model based on four parameters of Point-to-Point communications. The model is limited to small message sizes. Many adaptations extend this model, such as LogGP [1] for long messages. HLogGP model addresses heterogeneous issues in [3]. Zhang and Yan [15] describe models for heterogeneity for a network of workstations (NOW). Williams presents the HBSP model [14] for performance on heterogeneous platforms, with similar restrictions to those of BSP.

Sinnen and Sousa [12] show how processor heterogeneity is involved in communication. Kielmann et al [6] extend the LogP model for a wide area network distributed system. Casanova, Marchal, Roberts et al [10] propose models for WAN platforms to include issues such as network latency, bandwidth and topology.

Some authors e.g., Vadhiyar et al in [13] noticed the non-linearity behaviour of Many-to-One communication type. Other studies such as [5] restrict message sizes to below where there is observance of non-linear behaviour.

3 Communication Models

A switched Ethernet network means that each node communicates directly with every other node in a network. In this section we present a summary of our previous work that forms a basis for the proposed model [8].

3.1 Point-to-Point, One-to-Many and Multiple Point-to-Point Communication

The Point-to-Point communication is composed of the transmission delay, the source and destination node delays. For message size M the communication execution time model from node i to node j is given by;

$$T_{p2p} = C_i + t_i M + C_j + t_j M + M/\beta ij \tag{1}$$

where C_i and C_j are the fixed processing delays at node i and j, respectively. t_i and t_j are times to process a byte at node i and j, respectively. The transmission rate of the link connecting nodes i and j is β_{ij}.

One-to-Many communication is when a source node sends a message to arbitrarily number of nodes $n \leq N$, for a cluster of N nodes. One-to-Many communication model has two parts (c.f., Eq. 2). The first part is made up of small sized messages $(M \leq S)$, where by the nature of standard MPI communication mode, the buffered mode send is used. A buffered mode send operation sends the message whether or not a matching receive has been posted. It may complete before a matching receive is posted and hence One-to-Many model exhibits parallelism.

$$C_0 + t_0 nM + \begin{cases} \max\limits_{j}^{n}\Big\{C_j + t_j M + M/\beta_{0j}\Big\} & M \leq S \\ \sum\limits_{j=1}^{n}(C_j + t_j M + M/\beta_{0j}) & M > S \end{cases} \tag{2}$$

The second part is of the large sized messages $(M > S)$, the source must wait for a matching receive to be posted before sending the next message. In this context, this part is a serialised communication scheme (c.f., Eq. 2).

The communication execution time for a set of K multiple independent Point-to-Point communications is given by the maximum of the Point-to-Point communication time connections established;

$$T_{mp2p} = \max_{K}\{T_{ij}\} \tag{3}$$

Due to the nature of the full duplex Ethernet switch, disjoint pairs of Point-to-Point communication will be independent to each other and hence this communication type exhibits parallelism.

4 Many-to-One Communication

Many-to-One communication is when arbitrarily number of source nodes $n \leq N-1$ send the same message to the destination node.

The proposed Many-to-One communication model uses the flat tree and is made up of three parts (c.f., Fig. 1):

1. **Small message sizes ($0 \leq M \leq M_1$):** In this part, the proposed model is linear. Source nodes are sorted in ascending order of the sending overheads. The node with the least sending overhead will reach the destination node first assuming the transmission rate is the same for all links on the same message size. This is an analogue to a FIFO queueing system with multiple sources where the receiving node acts as server node with fixed service time. Provided the server utilisation is 100% (i.e., there are no idle times), the communication time is given by;

$$\Delta_1(C_0 + t_0 M + M/\beta_{0j} + \sum_{i=1}^{n-1}(C_i + t_i M)); \tag{4}$$

where Δ_1 is an adjustment factor to improve the accuracy of the proposed model.

The part with 0 subscript is the arrival time of a message from the source node with the least overhead. The summation part is the total receiving overhead for all messages coming from $n-1$ source nodes.

In the case were there are idle times, the total idle time T_{idle} is taken into consideration and Eq. 4 becomes;

$$\Delta_1(C_0 + t_0 M + M/\beta_{0j} + T_{idle} + \sum_{i=1}^{n-1}(C_i + t_i M)) \tag{5}$$

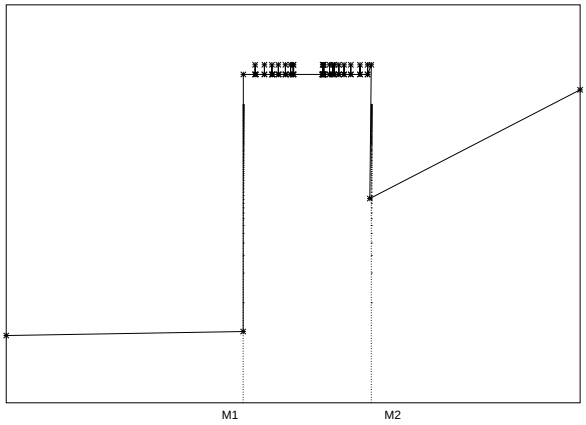

Figure 1. Many-to-One communication pattern

2. **Congestion region ($M_1 < M \leq M_2$):** This part experiences a congestion that leads to a sharp increase in communication execution times that reaches an upper bound of several constant discrete levels. As our first approximation the Eq. 6 accurately predicted execution time with a constant C for our cluster;

$$T_{mto}(M, N) = C; \tag{6}$$

3. **Large message sizes ($M > M_2$):** This is the last part of the proposed model where the linearity resumes. There is synchronisation between a sender and receiver as the data is transmitted with guarantees of system resources that are acknowledged to be available. This means that resources are allocated and reserved with sending and so congestion does not occur. The reservation mode has greater latency as there is an extra system overhead. The communication time is therefore given as;

$$\Delta_2(C_0 + t_0 M + M/\beta_{0j} + \sum_{i=1}^{n-1}(C_i + t_i M)) \tag{7}$$

The additional scaling factor Δ_2 represents the increased overhead for synchronous sending in reservation mode for larger messages.

4.1 Gather Communication

If $n = N-1$ then the Many-to-One communication is reduced to Gather communication type. It has been found experimentally that MPI_Gather communication time is very close to the one with the proposed Many-to-One communication model and also shows the same pattern. In this context the proposed Many-to-One communication model also accurately predicts the MPI_Gather communication time and hence Eqs. 4-7 are used for MPI_Gather predictions.

5 Experiment Setup

The experiments were done on a dedicated heterogeneous cluster at the School of Computer Science and Informatics, University College Dublin (UCD). The cluster is made up of 16 nodes, nine nodes are running Fedora Linux, 5 nodes have Debian Linux installed and 2 node run SunOs. The processors are as follows, 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. 3 HP DL140 Xeon Processors, 2.8GHz,3.4GHz and 3.6GHz. Two HP DL320 Celeron 2.9GHz and 3.4GHz Pentium 4 Processors. The cluster is connected via an Ethernet switch with adjustable bandwidth (from few Kilobytes) on each link.

5.1 Model Parameters

The t_i, t_j, C_i and C_j parameters were obtained by a simple ping pong communication in our previous work [8].The two parameters (i.e., C_i and t_i) for each node define the characteristics of a particular node, and are the building blocks for all other

experimental work.

M_1, M_2 and C are additional parameters found experimentally which are the characteristics of a given network.

6 Experimental Results

This section presents the experimental results carried out to validate the Many-to-One communication model.

6.1 Many-to-One for Small Message Sizes

Experimental results for Many-to-One communication type for small message sizes ($0 \leq M \leq M_1$) are compared against the Many-to-One communication model in Fig. 2. It can be seen that the model accurately predict the communication execution time. In our experiments M_1 was found to be $3KB$.

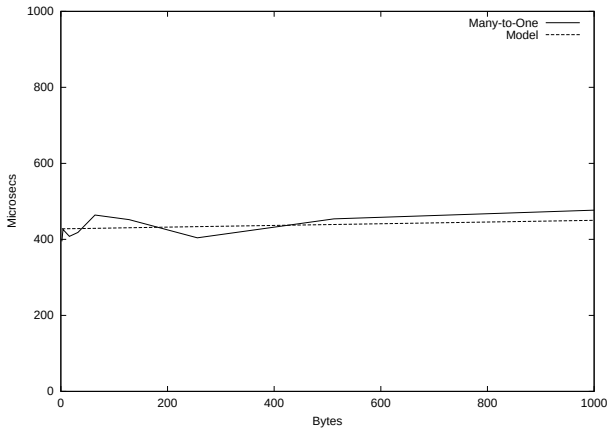

Figure 2. Many-to-One compared to model for small message Sizes

6.2 Congestion Region

Fig. 3 shows Many-to-One communication model validation against the experimental results in the congested region ($M_1 < M \leq M_2$), where M_1 and M_2 where found to be $3KB$ and $64KB$, respectively.

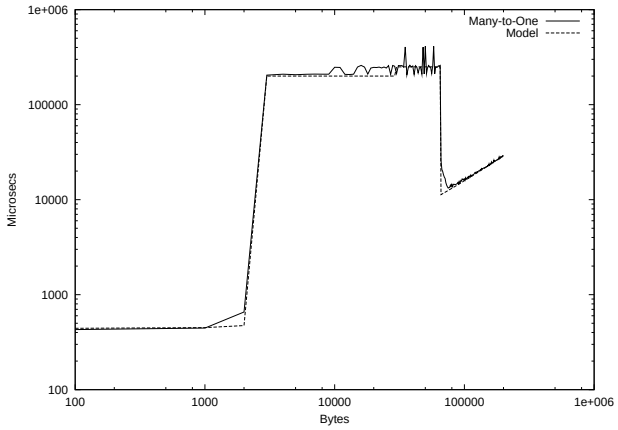

Figure 3. Many-to-One compared to model in congested region

6.3 Many-to-One for Large Message Sizes

The setting of the synchronous send by the system for large messages means that the execution time changes to include extra overhead, but remains linear.

Fig. 4 compares the prediction model against the experimental results for large messages size $M > M_2$

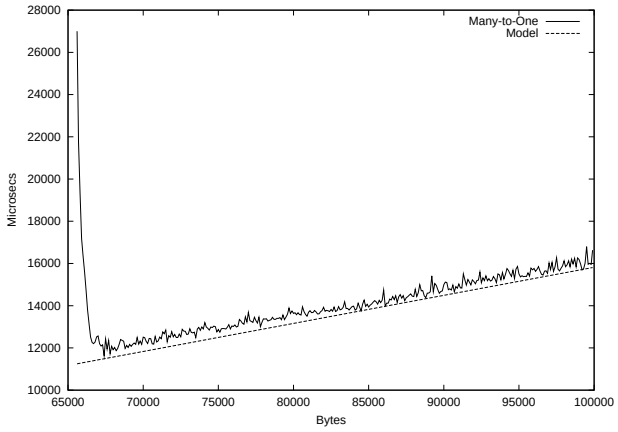

Figure 4. Many-to-One compared to model for large message sizes

6.4 Gather Communication

The propose Many-to-One communication model is used to accurately predict the MPI_Gather communication execution time (c.f., Fig. 5).

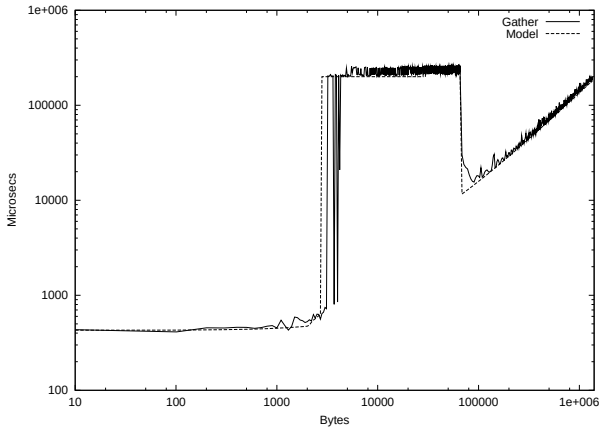

Figure 5. Gather compared to the proposed Many-to-One model

7 Conclusion

This paper present a model for Many-to-One communication execution times on a dedicated heterogeneous cluster base on a switched-Ethernet network. The model is adapted to reflect both linear and the non-linear behaviour caused by congestion. The proposed Many-to-One communication models also accurately predicted the MPI_Gather communication time.

We meet the requirements of heterogeneity with a variety of processor power and operating systems with a simple and practical solution.

Bibliography

[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] J. Bosque and L. Perez. Hloggp: A new parallel computational model for heterogeneous clusters. *4th IEEE/ACM International Symposium on Cluster Computing and the Grid (CCGrid 2004)*, 2004.

[4] 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.

[5] G. Fagg, J. Pjesivac-Grbovic, G. Bosilca, T. Angskun, J. Dongarra, and E. Jeannot. Flexible collective communication tuning architecture applied to open mpi. *submitted to 2006 Euro PVM/MPI*, 2006.

[6] 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.

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

[8] A. Lastovetsky, I. Mkwawa, and M. OFlynn. An accurate communication model for a heterogeneous cluster based on a switch-enabled ethernet network. *The 12th International Conference on Parallel and Distributed Systems (ICPADS 2006)*, 2006.

[9] 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.

[10] 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.

[11] 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.

[12] O. 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.

[13] S. Vadhiyar, G. Fagg, and J. Dongarra. Towards an accurate model for collective communications. *The Internaltional Journal of High Performance Computing Applications*, page 159, 2004.

[14] 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.

[15] X. Zhang and Y. Yan. Modeling and characterizing parallel computing performance on heterogeneous networks of workstations 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.