Communication Performance Measurement and Analysis on Commodity Clusters

A DISSERTATION
SUBMITTED TO THE DEPARTMENT OF COMPUTER SCIENCE
OF THE UNIVERSITY OF ADELAIDE
FOR THE DEGREE OF
DOCTOR OF PHILOSOPHY

By

Nor Asilah Wati Abdul Hamid

March 13, 2008

Table of Content

List of Figures

CHAPTER 2

CHAPTER 3

CHAPTER 4

CHAPTER 5

CHAPTER 6

CHAPTER 7

List of Tables

CHAPTER 1

CHAPTER 2

CHAPTER 3

CHAPTER 5

CHAPTER 6

CHAPTER 7

ABSTRACT

Cluster computers have become the dominant architecture in high-performance computing. Parallel programs on these computers are mostly written using the Message Passing Interface (MPI) standard, so the communication performance of the MPI library for a cluster is very important. This thesis investigates several different aspects of performance analysis for MPI libraries, on both distributed memory clusters and shared memory parallel computers.

The performance evaluation was done using MPIBench, a new MPI benchmark program that provides some useful new functionality compared to existing MPI benchmarks. Since there has been only limited previous use of MPIBench, some initial work was done on comparing MPIBench with other MPI benchmarks, and improving its functionality, reliability, portability and ease of use. This work included a detailed comparison of results from the Pallas MPI Benchmark (PMB), SKaMPI, Mpptest, MPBench and MPIBench on both distributed memory and shared memory parallel computers, which has not previously been done. This comparison showed that the results for some MPI routines were significantly different between the different benchmarks, particularly for the shared memory machine.

A comparison was done between Myrinet and Ethernet network performance on the same machine, an IBM Linux cluster with 128 dual processor nodes, using the MPICH MPI library. The analysis focused mainly on the scalability and variability of communication times for the different networks, making use of the capability of MPIBench to generate distributions of MPI communication times. The analysis provided an improved understanding of the effects of TCP retransmission timeouts on Ethernet networks.

This analysis showed anomalous results for some MPI routines. Further investigation showed that this is because MPICH uses different algorithms for small and large message sizes for some collective communication routines, and the message size where this changeover occurs is fixed, based on measurements using a cluster with a single processor per node. Experiments were done to measure the performance of the different algorithms, which demonstrated that for some MPI routines the optimal changeover points were very different between Myrinet and Ethernet networks and for 1 and 2 proc-

essors per node. Significant performance improvements can be made by allowing the changeover points to be tuned rather than fixed, particularly for commodity Ethernet networks and for clusters with more than 1 process per node.

MPIBench was also used to analyse the MPI performance and scalability of a large ccNUMA shared memory machine, an SGI Altix 3000 with 160 processors. The results were compared with a high-end cluster, an AlphaServer SC with Quadrics QsNet interconnect. For most MPI routines the Altix showed significantly better performance, particularly when non-buffered copy was used. MPIBench proved to be a very capable tool for analyzing MPI performance in a variety of different situations.

DECLARATION

This work contains no material which has been accepted for the award of any other degree or diploma in any university or other tertiary institution and, to the best of my knowledge and belief, contains no material previously published or written by another person, except where due to reference has been made in the text.

I give consent to this copy of my thesis, when deposited in the University Library, being available for loan and photocopying.

_____

NOR ASILAH WATI ABDUL HAMID
Phd Candidate,
Department of Computer Science
University of Adelaide
13 March 2008

LIST OF PUBLICATIONS

The following papers were written based on the work presented in this thesis.

Papers in Refereed Conference Proceedings

1. N. A. W. A Hamid and P. Coddington. "Analysis of Algorithm Selection for Optimizing Collective Communication with MPICH for Ethernet and Myrinet Networks", *Proc. of The* 8^{th} *International Conference on Parallel and Distributed Computing, Applications and Technologies (PDCAT'07), Adelaide, Australia, December 2007.*

2. N. A. W. A Hamid and P. Coddington. "Averages, Distribution and Scalability of MPI Communication Times for Ethernet and Myrinet Networks", *Proc. of Parallel and Distributed Computing and Network (PDCN'07),* 25^{th} *IASTED International Multi-Conference, Congress Innsbruck, Austria, February 2007.*

3. N. A. W. A Hamid, P. Coddington and F. Vaughan. "Comparison of MPI Benchmark Programs on an SGI Altix ccNUMA Shared Memory Machine"**,** *Proc. of Workshop on Performance Modeling, Evaluation, and Optimization of Parallel and Distributed Systems (PMEO-PDS'06), Rhodes, Greece, April 2006.*

4. N. A. W. A Hamid, P. Coddington and F. Vaughan. "Performance Analysis of MPI Communications on the SGI Altix 3700*", Proc. of APAC'05, Gold Coast, Australia, September 2005.*

ACKNOWLEDGEMENT

The work described in this thesis was carried out at the University of Adelaide in the Department of Computer Science under the supervision of Dr. Paul Coddington.

I would like to express my deepest gratitude to Dr. Paul Coddington for his excellent guidance, enthusiastic supervision and tolerance throughout this research. His dedication and contribution gave me tremendous help in completion of this research and the publication of the papers. I also would like to thank him for his commitment to this research and for giving me the chance to present our papers at conferences around the world.

Many thanks are due to the University of Adelaide and in particular the Department of Computer Science, as well as the South Australian Partnership for Advanced Computing (SAPAC) for making supercomputer time available to me. Special thanks to Patrick Fitzhenry and Grant Ward for their professional advice and patience throughout the experimental program. I also would like to give thanks for the help from a team of programmers from the University of Adelaide and the South Australian Partnership of Advanced Computing, Alex Chichowski, Tim Seely and Paul Martinaitis.

Lastly to my family, my husband Dr. Raizal Saifulnaz Muhammad Rashid and my son Muhammad Haziq Raizal Saifulnaz and my parents for their love, encouragement and continual support throughout the years of doing this research.