

Multi-Platform Benchmarking of the MPI Communications Interface

ABSTRACT

Ralf Reussner
Universität Karlsruhe
reussner@ira.uka.de

Peter Sanders
Max-Planck-Institut für Informatik
sanders@mpi-sb.mpg.de

Jesper Larsson Träff
C&C Research Laboratories, NEC Europe Ltd.
traff@ccrl-nece.de

January 19, 2001

MPI is arguably the most widespread communication interface for dedicated applications on (primarily) distributed memory machines [4]. Vendor implementations of MPI are available for most parallel platforms, be they either distributed memory, shared memory or hybrids like clusters of SMP nodes, as is the widely portable MPICH implementation [1]. The programming model of MPI is a distributed memory model with explicit communication among processes, coupled with powerful collective operations over sets of MPI processes. There is no performance model or guarantees coming with MPI (apart from liveness and other basic semantic properties), and although applications using MPI are semantically portable across different platforms, “performance portability” may not be ensured. To guide the design of efficient and *performance portable* parallel applications with MPI it is of the utmost important to have available reliable figures for performance characteristics of relevant MPI implementations for as many different platforms as possible. Performance characteristics include the “raw” performance of MPI communication primitives, both for message-passing and collective communication for varying parameters (message lengths, number of processes), performance under “load” (eg. bisection bandwidth) or with typical communication patterns (eg. master-slave), as well as comparative measurements of different realizations of collective operations (eg. alltoall communication via message passing compared to collective `MPI_Alltoall` primitive). Such information allows the application programmer to tune his application for a specific platform by choosing the appropriate communication primitives, and for tuning for good performance across different platforms.

There are several benchmarks for MPI which partly address these issues. In the talk we will discuss the Special Karlsruher MPI (*SKaMPI*) benchmark [2, 3], which in particular addresses the issue of cross platform performance portability by maintaining a *public performance data base* of performance measurements for different platforms. The main characteristics of *SKaMPI* are:

- It covers (almost) all of the MPI-1 standard, including the MPI collective operations, and the MPI derived data types [3] typically not addressed by other benchmarks.
- It assesses the performance under different communication patterns, eg. ping-pong and master-slave.

- *SKaMPI* does automatic parameter refinement for accuracy, reliability and speed (of benchmarking).
- The operation of *SKaMPI* is controlled by scripts which allows for detailed and flexible planing of experiments. The benchmark comes with a default suite of measurements.
- It contains a report generator which allows for automatic preparation of measurements into a readable form.
- Last, but not least (and what we will especially focus on in the talk) is a *public performance data base* available on the www, <http://liinwww.ira.uka.de/~skampi/>, which allows for interactive comparison of performance characteristics of various aspects of MPI across different implementations and platforms

SKaMPI was originally developed by Ralf Reussner and Peter Sanders at the University of Karlsruhe. The www-data base was implemented by Gunnar Hunzelmann.

References

- [1] W. Gropp, E. Lusk, N. Doss, and A. Skjellum. A high-performance, portable imlementation of the MPI message passing interface standard. *Parallel Computing*, 22(6):789–828, 1996.
- [2] Ralf Reussner, Peter Sanders, Lutz Prechelt, and Matthias Müller. *SKaMPI: A detailed, accurate MPI benchmark*. In *Recent Advances in Parallel Virtual Machine and Message Passing Interface. 5th European PVM/MPI Users' Group Meeting*, volume 1497 of *Lecture Notes in Computer Science*, pages 52–59, 1998.
- [3] Ralf Reussner, Jesper Larsson Träff, and Gunnar Hunzelmann. A benchmark for MPI derived datatypes. In *Recent Advances in Parallel Virtual Machine and Message Passing Interface. 7th European PVM/MPI Users' Group Meeting*, number 1908 in *Lecture Notes in Computer Science*, pages 10–17, 2000.
- [4] Marc Snir, Steve Otto, Steven Huss-Lederman, David Walker, and Jack Dongarra. *MPI -The Complete Reference*, volume 1, The MPI Core. MIT Press, second edition, 1998.