Performance Impact of the Interconnection Network on MareNostrum Applications

Alex Ramirez Oriol Prat Jesus Labarta Mateo Valero

Universitat Politecnica de Catalunya
HiPEAC European Network of Excellence
and
Barcelona Supercomputing Center

1 Abstract

Interconnection networks are one of the fundamental components of a supercomputing facility, and one of the most expensive parts. They represent one of the main differences between two supercomputers built from the same processor, and have a significant impact on how the applications should be developed. However, very little is known about how those expensive interconnection networks are used by the real applications running on supercomputing facilities. Furthermore, in the near future, chip multiprocessors offering near supercomputing capabilities, with 64 to 256 processor per chip, will be readily available. On-chip interconnection networks offer the possibility of new designs with lower latencies and much higher bandwidths.

In this paper we present an analysis of the impact of the interconnection network for some of the most representative applications running on MareNostrum, at the Barcelona Supercomputing Center. We have collected traces of real runs of the applications, and verified that our performance model (Dimemas) accurately predicts the real machine performance. Then, we present hypothetical situations where we change the network's latency, bandwidth, number of simultaneous connections, and CPU speed in order to quantify their importance on the final application performance in the context of future on-chip interconnenctions.

Our results show that the CPU speed proves more important than the interconnection network, and that among the network's parameters, interconnection bandwidth is far more important than latency (with a very low impact), or the connectivity (only relevant for low connection bandwidth).

2 Related Work

The usual way to measure performance of a supercomputer is the Linpack benchmark, used to obtain the Top500 list of supercomputing sites [1]. While most supercomputers have been designed and optimized in one way or another to improve their Linkpack performance, Linpack itself is not representative of the real world applications that run on those computers. Instead of evaluating the interconnection network for Linpack, we have used real applications consuming a significant fraction of MareNostrum's time.

In a related paper from Los Alamos supercomputing center, Kamil et al. [3] present a thorough study of ultrascale applications in terms of their communication requirements. They only present trace analysis of the applications in terms of MPI usage, type of communications, message sizes, and communication patterns. Using that information one could determine the kind of impact each of the network parameters may have on the applications, but it is still difficult to quantify such impact. In addition to a trace analysis tool (Paraver), the CEPBA tools include Dimemas, a performance model that allows quantitative analysis of such changes, as we present in this work.

3 Selected applications

This extended abstract shows performance predictions for three of the most representative applications running on MareNostrum, at the Barcelona Supercomputing Center: AMBER, CPMD, and WRF.

AMBER refers to two things: a set of molecular mechanical force fields for the simulation of biomolecules (which are in the public domain, and are used in a variety of simulation programs); and a package of molecular simulation programs which includes source code and demos.

Pmemd is an extensively-modified version (prepared by Bob Duke) of the sander program, optimized for periodic, PME simulations, and for GB simulations. It is faster, and scales better on parallel machines,

than sander; hence it is generally the program of choice, unless you need options that it does not support. In the code model we are now following, sander is the vehicle to explore new features, and pmemd is a production code that implements sander's most-used features in a well-tested fashion that performs well in high-performance environments.

The CPMD code is a parallelized plane wave/pseudopotential implementation of Density Functional Theory, particularly designed for ab-initio molecular dynamics.

The Weather Research and Forecasting (WRF) Model is a next-generation mesocale numerical weather prediction system designed to serve both operational forecasting and atmospheric research needs. It features multiple dynamical cores, a 3-dimensional variational (3DVAR) data assimilation system, and a software architecture allowing for computational parallelism and system extensibility. WRF is suitable for a broad spectrum of applications across scales ranging from meters to thousands of kilometers.

The effort to develop WRF has been a collaborative partnership, principally among the National Center for Atmospheric Research (NCAR), the National Oceanic and Atmospheric Administration (the National Centers for Environmental Prediction (NCEP) and the Forecast Systems Laboratory (FSL), the Air Force Weather Agency (AFWA), the Naval Research Laboratory, Oklahoma University, and the Federal Aviation Administration (FAA). WRF allows researchers the ability to conduct simulations reflecting either real data or idealized configurations. WRF provides operational forecasting a model that is flexible and efficient computationally, while offering the advances in physics, numerics, and data assimilation contributed by the research community.

4 Simulation study

4.1 Methodology

MareNostrum is composed of JS20 blades, with 2 PowerPC970FX processors at 2.1 GHz, 4 GB of DDR2 memory, and a bidirectional 256 MB/s optical link to the Myrinet interconnection network. The traces were obtained using the CEPBA Tools on benchmarking runs of the application, having the machine for exclusive use.

Dimemas [4, 2] is a performance analysis tool for message-passing programs. It enables the user to develop and tune parallel applications on a workstation, while providing an accurate prediction of their performance on the parallel target machine. The Dimemas simulator reconstructs the time behavior of a parallel application on a machine modeled by a set of performance parameters. Thus, performance experiments can be done easily. The supported target architecture classes include networks of workstations, single and clustered SMPs, distributed memory parallel computers, and even heterogeneous systems.

We have varied the following system configuration parameters: network latency (this includes the MPI call time), bandwidth, number of buses (this measures the number of simultaenous connections that can happen through the network), and CPU speed (relative to the original PowerPC 970FX 2.1GHz). The number of buses parameter allows Dimemas to avoid detailed simulation of the interconnection network. Instead of closely modelling the entire network for conflicts, usage, etc. Dimemas allows only a fixed number of simultaneous communications. Past studies have shown that this simplification is still very accurate for a number of network topologies [2].

We have verified that Dimemas indeed predicts the performance for the traces obtained using the actual configuration settings for MareNostrum. In all cases, the prediction falls within 5% of the actual execution time. The simulation values for MareNostrum are 230 MB/s interconnection bandwidth, 8 microseconds latency, 6 simultaneous communications, using 2 processors per node connected through a 600 MB/s memory bus.

4.2 Simulation results

Figure 1 shows the performance impact of latency and bandwidth. We have fixed the number of buses to infinite connectivity, and varied the latency between 1 and 32 microseconds, and bandwidth between 32 MB/s and infinite.

The results show that interconnection bandwidth is far more important than latency. All 3 applications increase performance as we increase bandwidth up to 256 MB/s on 64 processors. However, latency has only a minor impact on AMBER, causing a degradation of approximately 10% when going from 1 to 16 microseconds, and has no impact at all for CPMD and WRF. In any case, increasing bandwidth from 2 GB/s to infinite bandwidth hardly provides any performance benefits even when we increas ethe number of processors.

Figure 2 shows the performance impact of the number of interconnection buses, that is, the number of simultaneous communications that can happen through the network. We have fixed the latency to 8 microseconds, varied bandwidth between 32 MB/s and infinity, and connectivity between 1 and 36. The

value for 0 buses is a special simulator value, and actually represents infinite connectivity (a fully connected crossbar network).

The results show that, again, bandwidth is the primary performance factor for the interconnection network, specially for limited connectivity environments (1 or 2 simultaneous communications). The connectivity still proves relevant when the bandwidth is under 1 GB/s, but increasing connectivity beyond 4-5 simultaneous transfers makes little difference even for current commercial bandwidth connections (such as 512 MB/s). However, connectivity becomes more important as we increase the number of processors, but still it's only clearly visible for the slowest networks.

In terms of MareNostrum, which has a 250 MB/s interconnection, these results showthat it could still obtain a 5-15% performance improvement going from its current network to a fully connected crossbar.

Figure 3 shows the performance impact of the CPU speed as we increase the interconnection bandwidth. We have fixed latency to 8 microseconds, and assume a fully connected network. CPU speed changes relative to the original MareNostrum PowerPC 970FX from 50% slower (factor 0.5) to 15x faster (factor 15).

The results clearly show that CPU performance is far more important than interconnection bandwidth. For nominal CPU performance (factor 1) increasing bandwidth from 1 GB/s to 16 GB/s hardly obtains a 10% speedup. However, doubling CPU performance immediately translates to almost double performance. Very low interconnection bandwidth (under 512 MB/s) proves to be a limiting factor, and bandwidth starts to become a bottleneck as we increase CPU speed by factors of 10 or more, but still, even is such extreme scenarios, an interconnection bandwidth of 4 GB/s proves to be more than enough for all 3 applications under study.

5 Conclusions

We have presented performance projections for 3 real applications currently using the MareNostrum supercomputer, changing the interconnection latency, bandwidth, the network connectivity, and the CPU speed. Our results show that the importance of the interconnection network is not as high as that of the individual processor performance. Future massive on-chip multiprocessors may offer ideal latency, bandwidth, and connectivity values, and obtain performance improvements around 20%.

Among the 3 studied interconnection elements, latency proves to be the less relevant one, with 2 of the 3 applications being fully insensitive to changes in that parameter. Our results show that for high speed networks (over 1 GB/s), a limited connectivity (4-5 simultaneous connections) is more than enough. The need for high connectivity is lower as we increase the interconnection bandwidth. Finally, the network bandwidth is by far the most relevant factor, but offers only minimal performance improvements over 1 GB/s at realistic CPU speeds.

However, these results depend on the applications, and we must bear in mind that message passing applications are often designed for a specific interconnection network architecture. Given a different environment, some applications may be re-designed in order to obtain higher speedups from the enhanced interconnection network, canging the conclusions of this work.

References

[1] Top500 supercomputer sites. *http://www.top500.org/*.

[2] Sergi Girona, Jesus Labarta, and Rosa M. Badia. Validation of dimemas communication model for mpi collective operations. *EuroPVM/MPI 2000*.

[3] Shoaib Kamil, John Shalf, Leonid Oliker, and David Skinner. Understanding ultrascale application communication requirements. *IEEE International Symposium on Workload Characterization*, 2005.

[4] Jess Labarta, Sergi Girona, Vincent Pillet, Toni Cortes, and Luis Gregoris. Dip: A parallel program development environment. *Second International Euro-Par Conference on Parallel Processing-Volume II*, pages 665–674, 1996.

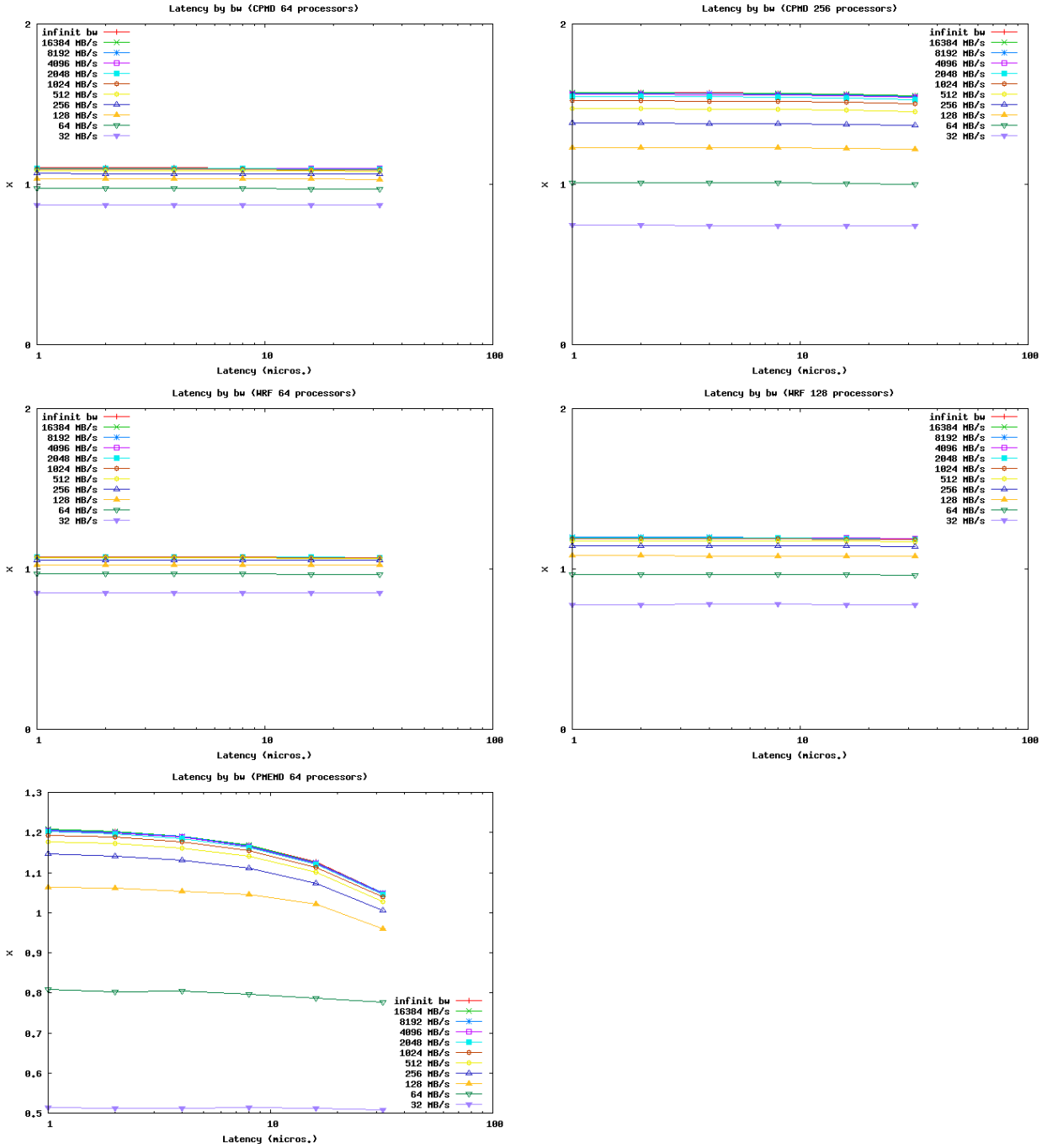

Figure 1: Performance impact of latency and bandwidth on a fully connected network.

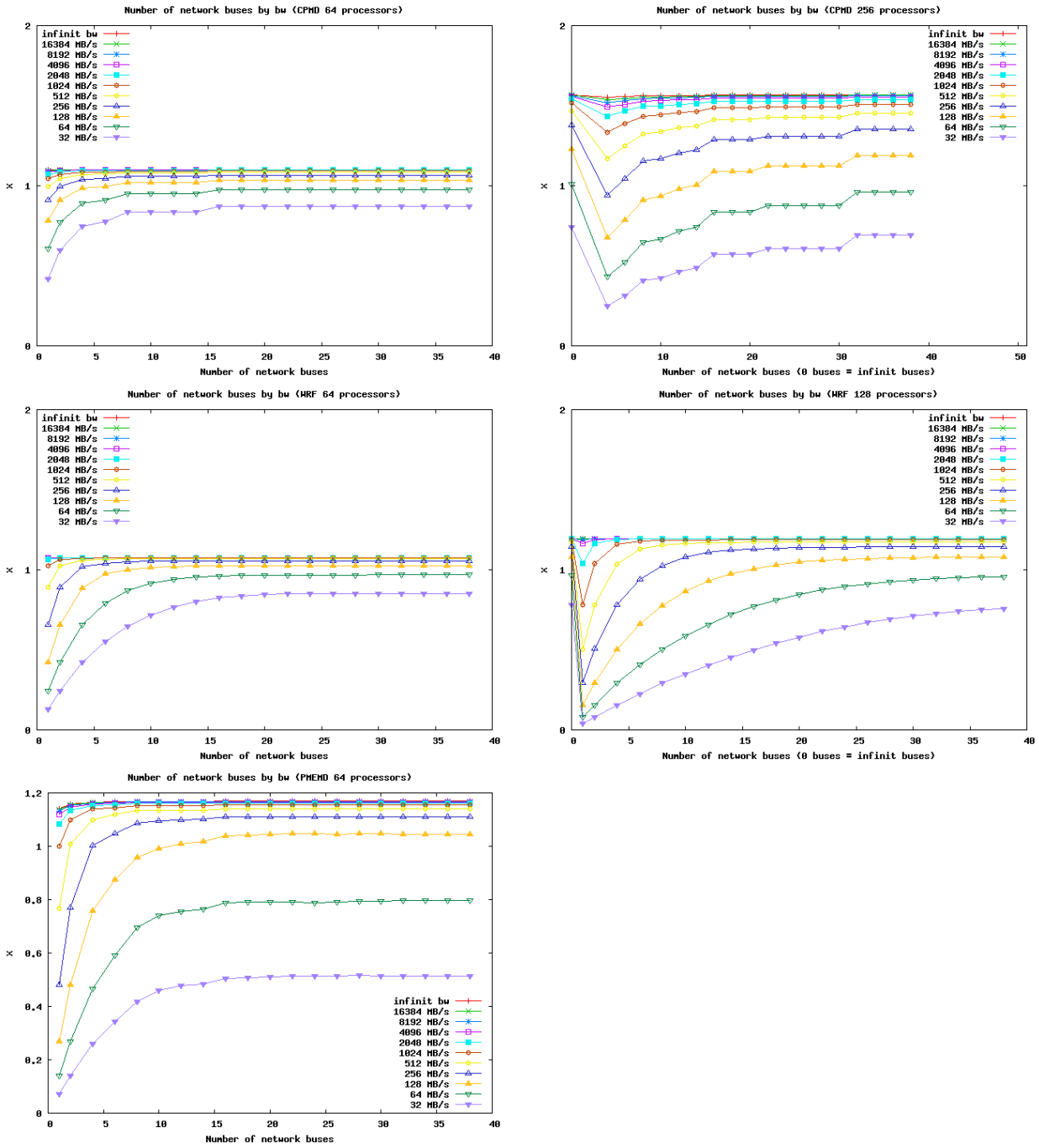

Figure 2: Performance impact of network connectivity and bandwidth.

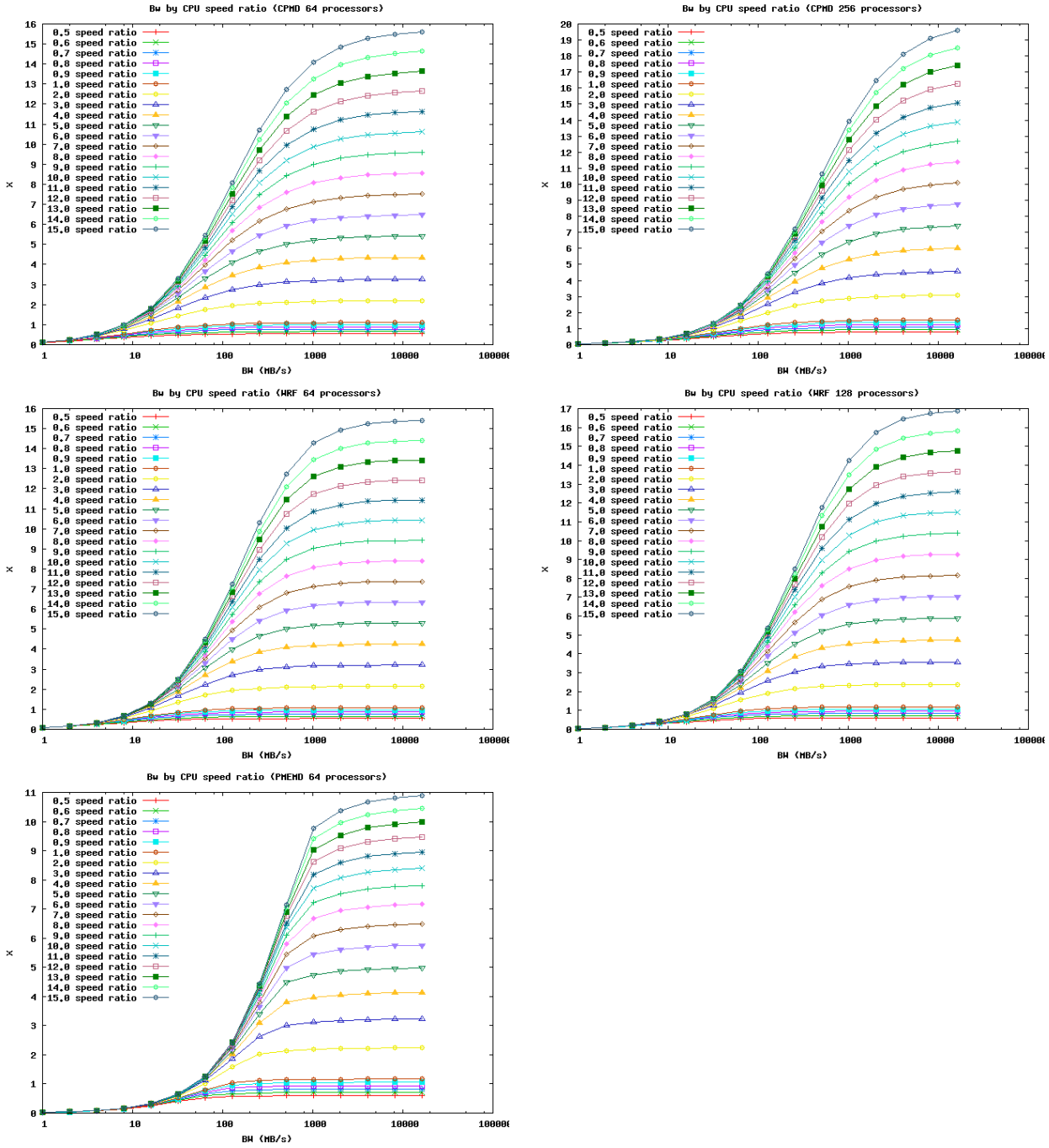

Figure 3: Performance impact of CPU speed and interconnection bandwidth.