A Power-Aware, Application-Based Performance Study Of Modern Commodity Cluster Interconnection Networks

Torsten Hoefler, Timo Schneider, and Andrew Lumsdaine
Open Systems Laboratory
Indiana University
Bloomington IN 47405, USA
$\{$*htor,timoschn,lums*$\}$*@cs.indiana.edu*

Abstract

Microbenchmarks have long been used to assess the performance characteristics of high-performance networks. It is generally assumed that microbenchmark results indicate the parallel performance of real applications. This paper reports the results of performance studies using real applications in a strictly controlled environment with different networks. In particular, we compare the performance of Myrinet and InfiniBand, and analyze them with respect to microbenchmark performance, real application performance and power consumption.

1. Introduction

The architecture and the resulting network characteristics of interconnection networks are critical to achieving high performance and scalability for parallel applications on HPC systems. It is generally assumed that the parallel performance of HPC applications can be predicted based on the parameters obtained by a small set of microbenchmarks. Typical network parameters are latency and bandwidth which can be measured easily with well-understood benchmarks [1], [2], [3].

However, there are few studies that actually compare microbenchmark results with real application runs. We show that simply relying on the two parameters latency and bandwidth often leads to inaccurate predictions. With this work, we want to promote the investigation of better network models and microbenchmarks that are able to give a more accurate prediction of application performance.

Another particular network characteristic that has not historically been monitored or studied is power consumption. The energy to run HPC systems is getting more and more expensive and is a potentially limiting factor in HPC design. Such concerns gave rise to the Green 500 list. Our work isolates the influence of different network architectures to the power consumption of HPC systems.

We study the two most widely used high-performance interconnection networks for cluster computing, Myrinet and InfiniBand, and analyzed them with regards to their microbenchmark performance, real application performance, and power consumption.

InfiniBand. InfiniBand [4] is the most-used commodity high-performance network in cluster computing. Its Single Data Rate (SDR) offers 8 Gbit/s data-transfer rate. Double Data Rate (DDR) and Quad Data Rate (QDR) offer 16 and 32 Gbit/s respectively. The latency can be lower than $1\,\mu$s if it is measured in a tight communication loop. It offers different modes of data transmission and features like RDMA or remote atomic operations. User-level messaging (kernel bypass) is supported by InfiniBand. The InfiniBand Architecture (IBA) has been analyzed in many research works [5], [6] and is well understood. There also exist two major Message Passing Interface (MPI) [7] implementations for InfiniBand, MVAPICH and Open MPI. Both implementations currently use RDMA by default to implement message passing semantics.

Myrinet 10G. Myrinet [8] is the 4th generation Myricom hardware. It offers a bandwidth of 10 Gbit/s for either Myrinet Express (MX) or Ethernet protocols. Latencies down to $2.3\,\mu$s are possible. Its physical layer is 10 Gigabit Ethernet. It also offers kernel bypass features to communicate directly from the user application. The MX communication layer is highly optimized for MPI point-to-point messaging and offers dynamic routing [9]. Myrinet is able to perform tag matching in the NIC firmware which further offloads communication functionality from the main CPU.

We compare two versions of Myrinet 10G (fiber- and copper-based) to copper-based InfiniBand ConnectX under identical environments. We use the same machines and the same Message Passing Interface (MPI) implementation in order to guarantee a direct comparison of the interconnection networks. The detailed hardware configuration and microbenchmark are presented in the next section. Section 3 presents application benchmark results for four carefully chosen real-world datasets and applications.

1.1. Related Work

Different research groups compared the performance of communication networks. Liu et al. [10] compares the characteristics of several high-performance interconnection networks with microbenchmarks. Another study by the same author [11] presents also some application comparisons. Other studies, like [12], [13], [14] limit themselves to microbenchmarks and the NAS parallel benchmarks. Bell et al. [15] discuss several parameters of modern high-performance networks such as full LogGP [16] parametrization. However, to the best of the authors knowledge, none of those works compare the power consumption with different interconnection networks.

2. Testbed

For our testbed we used 14 IBM iDataPlex dx360 nodes with SuSE Linux Enterprise Server 10 Service Pack 2 for x86_64 as operating system. We used the default SLES 10 SP 2 kernel version 2.6.16.60-0.21-smp. Our systems have two 2.5 GHz quad core Intel Xeons L5420 and the Intel 5400 chipset and are equipped with 32 GiB RAM (4 GiB per core). The host hardware was identical for all benchmarks, only the network interface cards were swapped. We also used Open MPI 1.2.8 (with the network specific transports) for all benchmarks to avoid perturbations of the benchmarks (e.g., by different collective algorithms).

We used a 24-port Cisco TopSpin SFS7000D switch and ConnectX IB cards (MT26418) for our InfiniBand tests. The kernel driver and userspace tools were included in the OFED 1.3 packages supplied by Novell. We used Open MPI 1.2.8 with the optimized *openib* Byte Transport Layer (BTL), which uses Remote Direct Memory Access (RDMA), for all tests. All parameters were left at their default values.

For our Myrinet over copper test, we used Myri-10G Dual-Protocol NICs (10G-PCIE-8A-C) and a switch from Myricom with a single line card (10G-SW32LC-16M). We used the MX driver version MX 1.4.3 and Open MPI 1.2.8 for all application tests. For Myrinet over fiber, we used the 10G-PCIE-8B-QP Myricom cards with a single linecard (10G-SW32LC-16QP). The QP fiber network cards run at a slightly higher clock-rate and are thus slightly faster. We used Open MPI 1.2.8 with the MX-optimized *mx* Matching Transport Layer (MTL), which enables tag-matching in hardware, for all tests with Open MPI.

2.1. Microbenchmark Results

In this section, we discuss several microbenchmarks to assess network performance. We used the well-known benchmark NetPIPE [3] to measure basic parameters such as latency and throughput. Figure 1 shows the latency for small messages for all investigated transport types. We analyzed the different Open MPI transport layers and TCP performance. The minimum (zero-byte) latency for InfiniBand (IB-C) with MPI was 1.38 μs for Myrinet with Copper (MX-C) 2.76 μs and for Myrinet with Fiber (MX-F) 2.53 μs. TCP latencies were 22.77 μs for InfiniBand (IP over IB) and 19.33 μs for Myrinet (native). Figure 2 shows the throughput that NetPIPE reported for the different networks. The sudden drops in this diagram are due to protocol changes. The highest throughput with 8 MiB messages was achieved with InfiniBand at about 14.2 Gbit/s which is 88 % of the peak performance. Myrinet reached up to 8.3 Gbit/s which is 83 % of the peak bandwidth. Again, the TCP performance was significantly lower. InfiniBand TCP (IP over IB) reached only 3.6 Gbit/s (22.5 %) and TCP over Myrinet reached 4.5 Gbit/s (45 %) for 8 MiB data transfers.

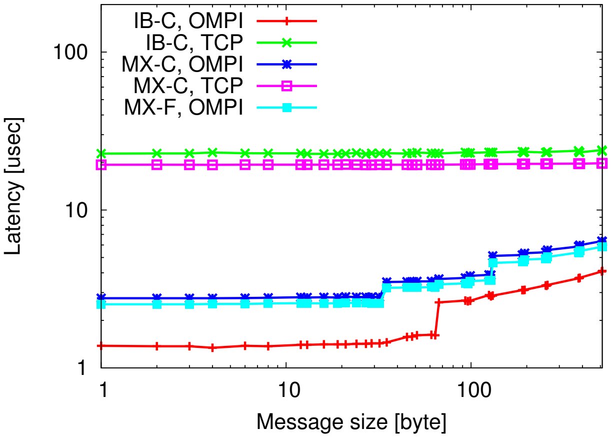


Figure 1. Latency for small messages

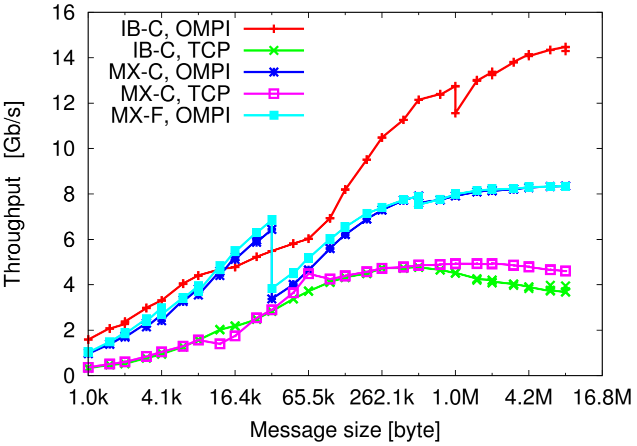


Figure 2. Throughput for large messages

In the next section, we discuss real application performance in comparison to the presented microbenchmarks.

3. Application Communication

In this section, we compare the different networking options for each application. We use the same input problem and an identical system configuration (except the communication network) for each application run. We record the total time to solution and the different MPI overheads. We also show the most significant sources of communication overheads for each application and each network in order to explain the difference precisely. We ran each application three times and report minimal values in order to eliminate operating system noise effects. However, the variance of all runs was very low (<5 %). We used the PMPI [7] interface to intercept and profile all MPI calls and report the average overheads over all processes.

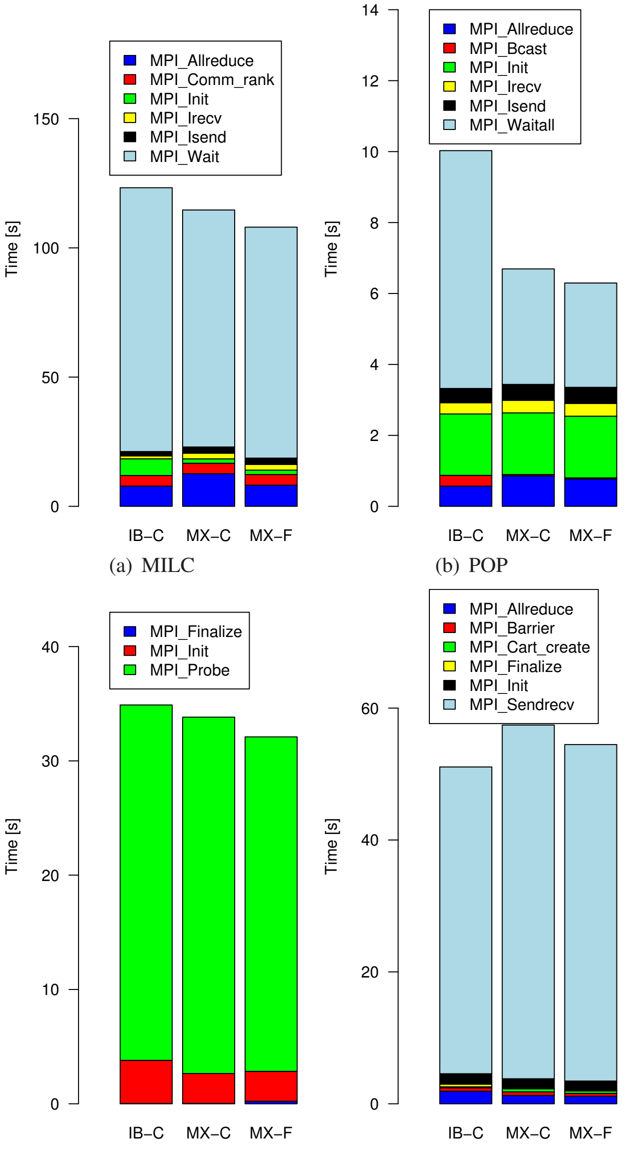
3.1. MILC

The MIMD Lattice Computation (MILC) code is used to study quantum chromodynamics, the theory of strong interactions of subatomic physics, such as used in high energy and nuclear physics. MILC consists of a multiple codes for specific tasks. We used the "medium" NERSC MILC benchmark for the `su3rmd` code. With InfiniBand (IB-C), the benchmark ran for 444.29 s and showed an MPI overhead of 123.34 s (27.76 %). With Copper Myrinet (MX-C), the same calculation ran for 435.27 s and exposed an MPI overhead of 115.35 s (26.50 %). With Fiber Myrinet (MX-F), the benchmark completed in 426.07 s with an MPI overhead of 106.63 s (25.03 %). The MPI overheads of MILC on 64 cores on 14 nodes comparing all three networks are shown in Figure 3(a). The main source of overhead are calls to MPI_Wait which are caused by a three-dimensional nearest neighbor communication pattern.

3.2. POP

The Parallel Ocean Program (POP) performs an ocean circulation simulation based on models described in [17]. It solves three-dimensional equations for fluid motions on the sphere using hydrostatic and Boussinesq approximations where spatial derivatives are calculated with finite difference (FD) discretizations. A preconditioned conjugated gradient solver is used to calculate the two-dimensional surface pressure.

For our test run we computed the x1 POP benchmark input file [18] with 32 cores on 14 nodes. The computation took 66.30 s with InfiniBand and had an MPI overhead of 10.06 s (15.17 %). With Copper Myrinet, the runtime was 62.97 s with 6.72 s (10.68 %) communication overhead. The faster Fiber Myrinet lowered the runtime to 60.98 s with only 6.29 s (10.31 %) MPI overhead. The main source of POPs communication overhead is also MPI_Waitall from the FD nearest neighbor communication. A detailed list is given in Figure 3(b).



(a) MILC

(b) POP

(c) RAxML

(d) WPP

Figure 3. MPI communication overheads

3.3. RAxML

The biological simulation Random Axellerated Maximum Likelihood (RAxML) infers phylogenetic trees, a popular method to model evolution, from DNA sequence data. The relationship between different species is determined by comparing parts of their DNA. The DNA sequences in question have to be aligned and the difference between them, i.e., how many mutations had to happen so that sequence A evolved into sequence B, has to be determined. The search space for this problem, the number of possible trees, is large. For $n = 50$ species there are about 10^{80} possible trees (as many as atoms in the universe). This number grows exponentially with n. A good overview of the algorithms used by RAxML and related tools is given in [19].

The MPI parallelization of RAxML is coarse grained, each rank computes different trees and sends the results to rank zero. We calculated 112 phylogenetic trees on all 112 cores with the same random seed on all networks for reproducibility using a database with 50 pre-aligned genome sequences consisting of 5000 base pairs. This search took 746.97 s with InfiniBand and showed an MPI overhead of 34.90 s (4.67 %). The computation took 742.62 s with Copper Myrinet with an MPI overhead of 32.10 s (4.32 %). On Fiber Myrinet, it took 738.35 s with an MPI overhead of 31.60 s (4.28 %). The lion's share of the MPI overhead is MPI_Probe. The MPI overhead for each network is detailed in Figure 3(c).

3.4. WPP

The Wave Propagation Program (WPP) simulates the time-dependent elastic and viscoelastic propagation of waves. WPP uses a Cartesian grid to solve the governing equations using a node-based finite difference approach. WPP is used for three dimensional seismic modeling and is capable of simulating a large variety of materials and is able to output synthetic seismogram as well as two dimensional slices through the model. The mathematical foundations of WPP are described in [20]. Our benchmark simulated an $30000 \times 30000 \times 17000$ grid with a spacing of 20 and a single wave source in it on all 112 cores. The layout of the grid was the same as in the LOH1 example distributed with WPP. All output was written after the last timestep was computed. The MPI overhead of WPP is dominated by MPI_Sendrecv. On InfiniBand, the calculation took 701.60 s and had an MPI overhead of 51.08 s (7.28 %). On Copper Myrinet, WPP needed 705.83 s with an MPI overhead of 57.43 s (8.14 %). On Fiber Myrinet, the computation time was 700.95 s with 53.37 s (7.61 %) MPI overhead. An overview of the functions that contribute to WPPs MPI overhead can be found in Figure 3(d).

3.5. Message Size Distribution

We also need to analyze the message size distribution in order to draw conclusions from the microbenchmark results. One would expect that bandwidth-limited messages are significantly slower on Myrinet because it has lower bandwidth. Figure 4 shows the accumulated message sizes for each application. A point (x, y) in this graph means that the sum of all messages smaller than x Byte that have been transmitted is y. For example MILC sends a lot of small messages with sizes up to 32 Byte, these messages make up for 163 Kilobyte of transmitted data. MILC also sends many large data-transfers between 10 kiB and 512 kiB. POP mainly sends small messages below 10 byte and 1 kiB messages and should thus be latency-bound. RAxML only exchanges 1 KB of 1 B messages and some large messages in the 100 kiB range. WPP uses mostly very large, bandwidth-limited, message transfers.

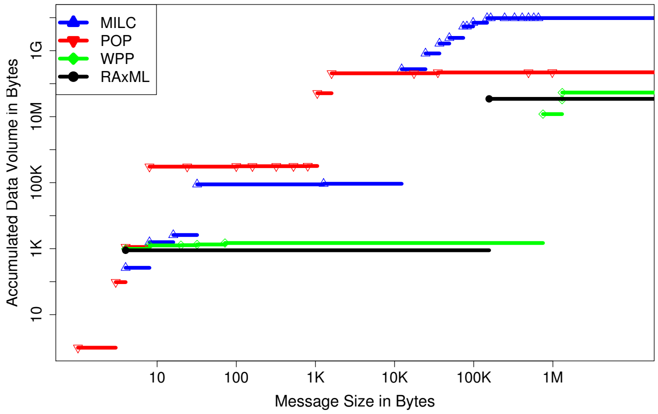


Figure 4. Message Distribution for all Applications
A point (x, y) in this graph means that the sum of all messages smaller than x Byte that have been transmitted is y.

4. Power Measurements

In this section we analyze the power consumption of the different applications. We did this by sampling the root mean square current drawn by the whole cluster (described in Section 2) every second. Our cluster was connected to the 120V power line via two APC 7800 power distribution units (PDUs). We sampled the current through our cluster by querying the PDUs via SNMP. With this method, we are able to compute the total power consumption for the solution of the particular problem for each application. In this case, the power consumption is the discrete integral (sum) over all measurement points. We report the power consumption graphs over time and the total energy needed to compute a particular input.

In a first experiment, we compare the current drawn by our idle system (without the switch) with the three different network configurations. The system equipped with InfiniBand uses 17.7 A when idle. Copper Myrinet lowers the idle-current consumption to 17.3 A and Fiber Myrinet to 16.9 A. We also analyzed the current consumption of the switch. The Cisco InfiniBand switch uses 0.48 A. The 7U Myrinet switch uses 0.75 A with the fan unit. However, removing the fan (which was safe with only a single line-card) decreased the current drawn to 0.48 A. The power consumption of the switches was identical when idle and under full load. We also investigated the power consumption of 4 nodes under full communication load. We used a bidirectional stream of 8 MiB messages to measure the power consumption in a micro-benchmark. Four InfiniBand

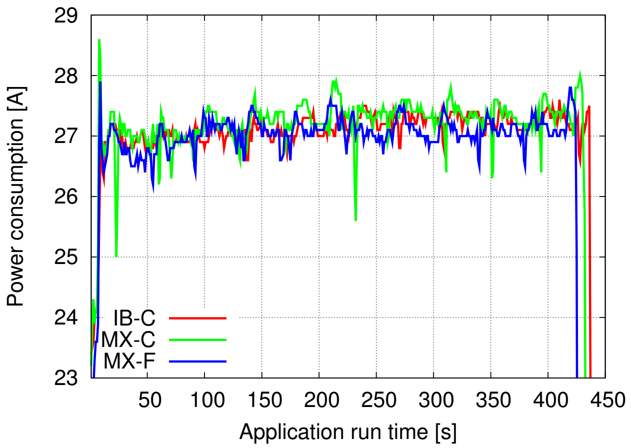


Figure 5. Power usage for MILC

nodes used 3.9 A when idle which increased to 5.0 A under full message load. Four Myrinet (copper) nodes used 3.77 A when idle and 4.95 A under full load with Open MPI using the OB1 PML. However, when we switched to the MX-optimized PML CM with MTL *mx* (our default transport in this article), which enabled matching in hardware, the power consumption was reduced by 4 % to 4.75 A. This interesting observation seems to be a result of packet matching on the specialized NIC processor.

In the following, we compare the power consumption of each application and input problem for the different networking technologies. We also compute and compare the total energy consumption that is required to solve the particular real-world problem.

4.1. MILC

The computation of the medium input file from the NERSC MILC benchmark needs 3.879 kWh with InfiniBand. Copper Myrinet uses 3.875 kWh which is approximately 0.1 % less. The power consumption with Fiber Myrinet is 3.819 kWh which is around 1.5 % less than InfiniBand.

4.2. POP

The Parallel Ocean Program uses, depending on the interconnect between 20.5 A and 21.5 A. This big variation is due to the heavy communication in the application (15 % communication overhead). The computation of the X1 pop benchmark [18] needs 0.458 kWh with InfiniBand. Copper Myrinet needs 0.432 kWh which is about 4.6 % less than with InfiniBand. Fiber Myrinet uses about 0.406 kWh, which is about 11.3 % less, to compute the result.

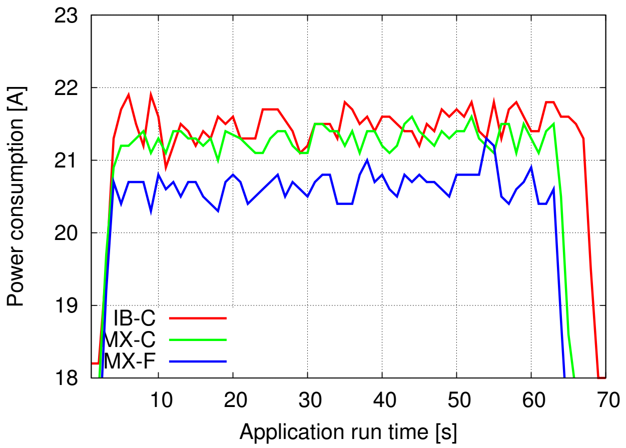


Figure 6. Power usage for POP

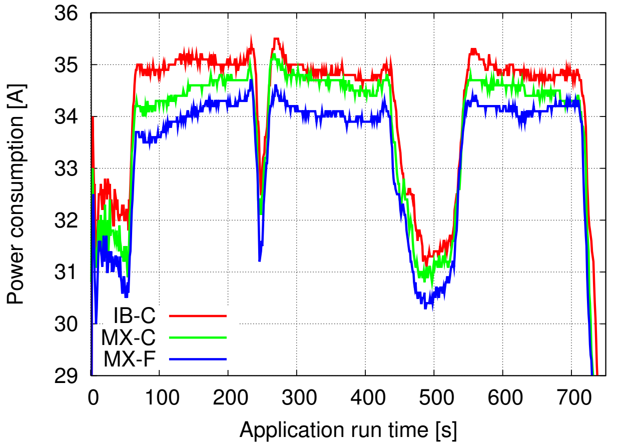


Figure 7. Power usage for RAxML

4.3. RAxML

The RAxML computation has different phases with difference power consumption. This shows nicely that the CPU is used differently in those phases. We would assume that a higher power consumption means more efficient CPU usage. RAxML has a peak with more than 35 A in our measurement.

The generation of the “tree of life” for the 50 species in our input file used 8.315 kWh in InfiniBand. Copper Myrinet used with 8.164 kWh around 1.8 % less energy. Fiber Myrinet uses 8.015 kWh which is around 3.6 % less than InfiniBand. RaxML only exhibits a small communication overhead, thus the energy consumption is only marginally influenced by the network.

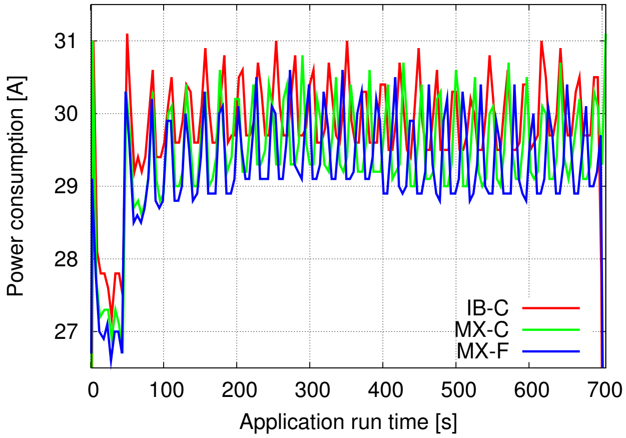


Figure 8. Power usage for WPP

4.4. WPP

The Wave Propagation Program uses between 29 A and 31 A and the power consumption varies highly during the application run. The computation of the seismic properties modeled in the (modified) LOH1 example shipped with WPP uses 6.807 kWh on InfiniBand. Copper Myrinet uses around 6.781 kWh which is about 0.4 % less energy consumption. Fiber Myrinet lowers the energy consumption by 1.4 % to 6.713 kWh.

5. Conclusions and Future Work

The first and most important conclusion is that networking microbenchmarks and simple metrics such as latency and bandwidth do not necessarily reflect the performance of real-world applications. Many other effects such as support for tag matching in hardware, memory registration or remote direct memory access influence the performance of real applications significantly. We show that even though microbenchmarks predict that Myrinet should be slower than InfiniBand, Myrinet performs significantly better than InfiniBand for many investigated applications. We note that those results are tightly bound to our specific setting and allow for little generalization. We also expect that the power consumption of InfiniBand with fiber transmission would be less than InfiniBand's copper version. However, we conclude that current microbenchmark metrics are not sufficient to predict application performance well. Thus, we advise to conduct detailed application studies to assess the performance of network interfaces in a particular setting. We also propose to analyze other more detailed performance assessment schemes, such as the LogGP model.

Power consumption is an important parameter for high-performance networks. We demonstrate that the energy needed to compute a certain result can be decreased by up to 11 % with a power-efficient interconnection network. We also show that the energy consumption of an idle system significantly depends on the networking equipment. Those results will hopefully influence the design of future networks to be more energy efficient. We suspect that the effective use of special hardware support (on our case tag-matching in hardware) can decrease the power consumption significantly.

Acknowledgments

This work was supported by the Department of Energy project FASTOS II (LAB 07-23), a grant from the Lilly Endowment and a gift the Silicon Valley Community Foundation on behalf of the Cisco Collaborative Research Initiative.

References

[1] T. Hoefler, T. Mehlan, A. Lumsdaine, and W. Rehm, "Netgauge: A Network Performance Measurement Framework," in *High Performance Computing and Communications, Third International Conference, HPCC 2007, Houston, USA, September 26-28, 2007, Proceedings*, vol. 4782. Springer, 9 2007, pp. 659–671.

[2] Pallas GmbH, "Pallas MPI Benchmarks - PMB, Part MPI-1," Tech. Rep., 2000.

[3] D. Turner, A. Oline, X. Chen, and T. Benjegerdes, "Integrating New Capabilities into NetPIPE." in *Proceedings of the 10th European PVM/MPI Users' Group Meeting*, ser. Lecture Notes in Computer Science, J. Dongarra, D. Laforenza, and S. Orlando, Eds., vol. 2840. Springer, 2003, pp. 37–44.

[4] The InfiniBand Trade Association, *Infiniband Architecture Specification Volume 1, Release 1.2*, InfiniBand Trade Association, 2003.

[5] G. M. Shipman, T. S. Woodall, R. L. Graham, A. B. Maccabe, and P. G. Bridges, "InfiniBand Scalability in Open MPI," in *Proceedings of IEEE Parallel and Distributed Processing Symposium*, April 2006.

[6] J. Liu, J. Wu, S. P. Kini, P. Wyckoff, and D. K. Panda, "High performance RDMA-based MPI implementation over InfiniBand," in *ICS '03: Proceedings of the 17th annual international conference on Supercomputing*. New York, NY, USA: ACM, 2003, pp. 295–304.

[7] MPI Forum, "MPI: A message-passing interface standard. version 2.1," September 4th 2008, `www.mpi-forum.org`.

[8] N. J. Boden, D. Cohen, R. E. Felderman, A. E. Kulawik, C. L. Seitz, J. N. Seizovic, and W.-K. Su, "Myrinet: A gigabit-per-second local area network," *IEEE Micro*, vol. 15, no. 1, pp. 29–36, 1995.

[9] P. Geoffray and T. Hoefler, "Adaptive Routing Strategies for Modern High Performance Networks," in *16th Annual IEEE Symposium on High Performance Interconnects (HOTI 2008)*. IEEE Computer Society, Aug. 2008, pp. 165–172.

[10] J. Liu, B. Chandrasekaran, W. Yu, J. Wu, D. Buntinas, S. Kini, D. K. Panda, and P. Wyckoff, "Microbenchmark Performance Comparison of High-Speed Cluster Interconnects," *IEEE Micro*, vol. 24, no. 1, pp. 42–51, January 2004.

[11] J. Liu, B. Chandrasekaran, J. Wu, and W. Jiang, "Performance Comparison of MPI implementations over Infiniband, Myrinet and Quadrics," in *in proceedings of Int'l Conference on Supercomputing, (SC'03*, 2003.

[12] J. Hsieh, T. Leng, V. Mashayekhi, and R. Rooholamini, "Architectural and performance evaluation of GigaNet and Myrinet interconnects on clusters of small-scale SMP servers," in *Supercomputing '00: Proceedings of the 2000 ACM/IEEE conference on Supercomputing (CDROM)*. Washington, DC, USA: IEEE Computer Society, 2000, p. 18.

[13] S. Majumder and S. Rixner, "Comparing Ethernet and Myrinet for MPI communication," in *Proceedings of the 7th workshop on Workshop on languages, compilers, and runtime support for scalable systems*. New York, NY, USA: ACM, 2004, pp. 1–7.

[14] M. Lobosco, V. S. Costa, and C. L. de Amorim, "Performance Evaluation of Fast Ethernet, Giganet, and Myrinet on a Cluster," in *ICCS '02: Proceedings of the International Conference on Computational Science-Part I*. London, UK: Springer-Verlag, 2002, pp. 296–305.

[15] C. Bell, D. Bonachea, Y. Cote, J. Duell, P. Hargrove, P. Husbands, C. Iancu, M. Welcome, and K. Yelick, "An Evaluation of Current High-Performance Networks," in *Proceedings of the 17th International Symposium on Parallel and Distributed Processing*. Washington, DC, USA: IEEE Computer Society, 2003, p. 28.1.

[16] A. Alexandrov, M. F. Ionescu, K. E. Schauser, and C. Scheiman, "LogGP: Incorporating Long Messages into the LogP Model," *Journal of Parallel and Distributed Computing*, vol. 44, no. 1, pp. 71–79, 1995. [Online]. Available: citeseer.ist.psu.edu/article/alexandrov95loggp.html

[17] K. Bryan, "A numerical method for the study of the circulation of the world ocean," *J. Comput. Phys.*, vol. 135, no. 2, pp. 154–169, 1997.

[18] P. Jones, P. Worley, Y. Yoshida, J. White, and J. Levesque, "Practical performance portability in the Parallel Ocean Program(POP)," *Concurrency and Computation Practice and Experience*, vol. 17, no. 10, pp. 1317–1327, 2005.

[19] A. Stamatakis, "Distributed and Parallel Algorithms and Systems for Inference of Huge Phylogenetic Trees based on the Maximum Likelihood Method," *PhD Thesis Technische Universitaet Muenchen*, 2004.

[20] S. Nilsson, A. N. Petersson, B. Sjögreen, and H. O. Kreiss, "Stable Difference Approximations for the Elastic Wave Equation in Second Order Formulation," *SIAM Journal on Numerical Analysis*, vol. 45, no. 5, pp. 1902–1936, 2007.