Building Cost Effective High Performance Computing Environment via PCI Express

A. Nishida
Chuo University / CREST, JST
nishida@kc.chuo-u.ac.jp
Tel:+81-3-3817-7412, Fax:+81-3-3817-7413

Abstract

Building a scalable and low cost parallel computing environment is becoming a critical matter for scientific computing that requires the repetitive solutions of large linear systems. In this study, a PCI Express based PC cluster is built for the scalable implementation of parallel iterative linear solvers, and the performance and bottlenecks are evaluated. The results show that the communication bandwidth of the internode network is the most important factor for the scalable implementation of parallel sparse linear algebra operations.

Keywords : sparse linear algebra operations, PCI Express, InfiniBand, Opteron processor, NUMA systems

1 Introduction

The recent wide use of commodity off-the-shelf (COTS) technologies for computer hardware has made cluster computing a practical choice for various fields of large-scale scientific calculations [6, 11]. However, compared with the proprietary systems with high speed interconnects, it is known to be much more diffucult task to get scalable results with sparse linear algebra operations[1] on COTS clusters, because of its limited network performance. It affects the performance of libraries such as Lis, a parallel library of iterative solvers that we are developing [1].

As we can see from the algorithm of the conjugate gradient method, shown in Figure 1, the linear algebra operations such as matrix-vector products with indirect references are the most time consuming computations in the iterative solvers with large-scale sparse coefficient matrices. In the conjugate gradient method, most of the operations consist of vector sums, dot products, and matrix-vector products, as shown in Figure 2. To implement them efficiently in parallel, a powerful interconnect is indispensable for frequent global communications. In this study, we have built a COTS cluster with low cost and high performance based on the PCI Express technology, and shown that the bandwidth between the cluster nodes is the most critical factor for the scalable parallel implementation of sparse matrix operations. From the results obtained from the experiments, we can see that the appropriate combination of the COTS technologies can beat the performance of proprietary parallel computer systems, even for the real applications that requires high bandwidth.

```
1. Choose $x_0$
2. $p_0 = r_0 = b - Ax$
$k = 0$
3. $\alpha_k = (r_k, p_k)/(p_k, Ap_k)$
4. $x_{k+1} = x_k + \alpha_k p_k$
5. $r_{k+1} = r_k - \alpha_k Ap_k$
6. $\beta_k = (r_{k+1}, r_{k+1})/(r_k, r_k)$
7. $p_{k+1} = r_{k+1} + \beta_k p_k$
Increment $k$ and if not convergent goto 3
```

Figure 1: Algorithm of Conjugate Gradient Method.

2 Background

PCI Express, the new standard of PC bus architecture that is compatible with PCI bus, is beginning to replace PCI-X with its higher performance and lower cost. Various technologies to connect computers with high speed interconnects such as Myrinet, Quadrics, and InfiniBand have been proposed for cluster com-

[1] One of the most typical examples is NAS Parallel Benchmark CG kernel. Some results on massively parallel COTS environments are reported [4].

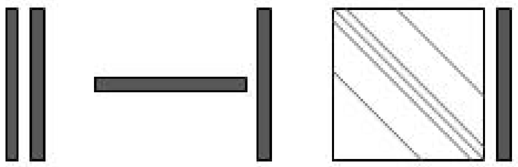

Figure 2: Various linear algebra operations for iterative solvers.

puting. Especially, InfiniBand is growing recently as one of the key technologies for developing high performance and low cost clusters. In this study, we combine these two emerging technologies with Opteron based low cost NUMA architecture, and discuss its scalability for large scale sparse matrix operations. We note our backgrounds below:

Bus standards

PCI Express is a standard of the bus architecture with the serial communication interface compatible with the existing PCI bus technology. It has been put into practical use in 2004. It can connect devices directly, and by linking up to 32 lanes with 2.5Gb/s unidirectional bandwidth per lane, it has bidirectional bandwidth of up to 16GB/s with 8B/10B coding[2]. See Figure 3. The integration with AGP (Accelerated Graphic Port) bus for graphic cards has enabled us to build a cluster environment with high performance interconnects using low cost commodity motherboards.

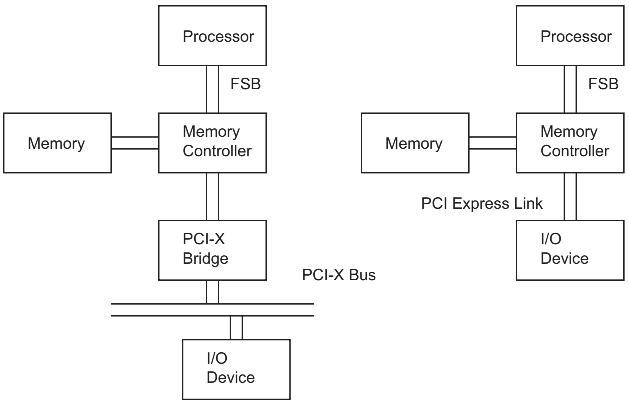

Figure 3: Comparing PCI-X with PCI Express.

Networking technologies

InfiniBand is one of the earliest technologies that has begun supporting PCI Express in 2004. An InfiniHost host channel adapter (HCA) developed by Mellanox Technologies, Inc. has two 4x ports (each port has 4 lanes of 2.5Gbps unidirectional bandwidth). See Figure 4). Using an 8x PCI Express slot, a dual port host channel adapter (HCA) achieves 40Gbps bidirectional bandwidth with 8B/10B coding.

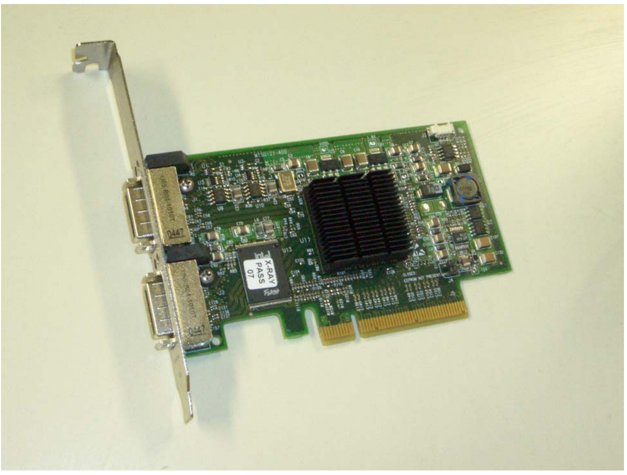

Figure 4: InfiniHCA HCA MHEA28-XT for PCI Express bus from Mellanox Technologies, Inc.

The data rate of the InfiniBand is to be expanded further in the near future. Mellanox Technologies, Inc. has released a new generation of InfiniHost chip with 20Gb/s unidirectional bandwidth per port.

Processor architectures

The Opteron Processor from Advanced Micro Devices, Inc. is a 64bit processor that supports distributed shared memory architecture. It includes a memory controller on the chip, which realizes scalable memory bandwidth. We show the results of a some control experiments of blobal memory bandwidth using Opteron and Intel's Xeon Processors in Section 3.1.

Operating systems

Since the Opteron architecture supports NUMA technology, you can choose an operating system that supports memory affinity to get scalable global memory bandwidth, or the sustained memory bandwidth of a parallel computer that is measured as the sum of the

[2] It adopts 8B/10B coding to keep redundancy.

local memory bandwidth of each processor [3]. In this study, we used Linux operating system with kernel 2.6.4 (SuSE Linux 9.1) for our nodes, which supports memory affinity for NUMA systems.

3 Performance Evaluation

Based on the backgrounds, we have decided to use two kinds of dual port InfiniHost HCAs for 8x PCI Express, MHEL-CF128-T with 128BM DDR memory, and MHEA28-XT that omits local memory by utilizing the main memory of the host computer. Considering the cost, we have introduced four MHEL-CF128-T HCAs and eight MHEA28-XT HCAs, and combined these HCAs with a 24-port InfiniBand switch MTS2400. MTS2400 is a switch based on the tree topology.

As a preliminary experiment, we have measured the performance of the InfiniBand HCAs using two 64bit Xeon servers (Dell PowerEdge SC 1420 with Intel E7520 chipset), and two Athlon64 self-made servers (with Asus A8N-SLI Deluxe motherboard using NVIDIA's nForce4 SLI chipset), since the Opteron motherboads with PCI Express slots have not been available in 2004. Both E7520 and nForce4 SLI support PCI Express.

We have attached MHEL-CF128-T HCAs to the four servers, and measured communication performance. The unidirectional bandwidth between two Athlon64 servers with 1.8GHz and 2.2GHz processors has been 970MB/s, while the bandwidth between two 2.8GHz Xeon servers (with no hyperthreading) was 800MB/s. MPI latency has been less than 4us with both of them, which is close to the official performance measured by Mellanox of 3.7us.

Based on the preliminary results, we have decided to use the motherboards with NVIDIA's nForce chipset for AMD's processors. The first 2-way motherboard for Opteron with PCI Express slots is HDAM Express that has been released from Arima Computer Corporation in March 2005. We have built an 8-node cluster system named IBD with 16 Opteron 246 2GHz processors with 1MB cache, connected with InfiniBand via the 16x PCI Express slots for graphic cards on HDAM Express. See Figure 5 for the diagram.

We have some MPI libraries for InfiniBand, such as MVAPICH [9], LAM MPI [5], and MPICH/SCore [12]. MVAPICH 0.9.5 or later has the striping function to split messages for muitiple ports and average the load if neccessary [10]. Since the InfiniHost HCA for PCI Express has two 4x ports, we have used MVAPICH

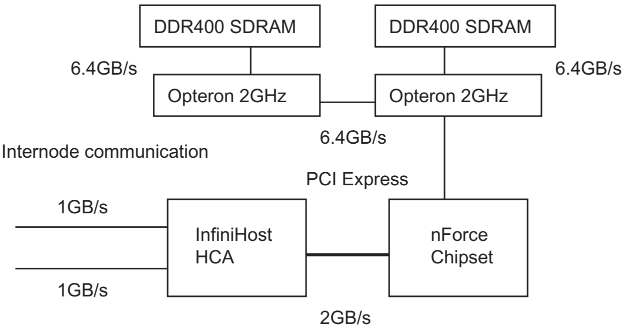

Figure 5: Diagram of 2P Opteron Server with nForce Chipset.

Figure 6: HDAM Express based Opteron Cluster IBD connected with InfiniHost HCAs and MTS2400.

0.9.5 built with gcc for our evaluation. We have also investigated the cases with single 4x InfiniBand port and gigabit ethernet. For gigabit ethernet, we have used Realtek's RTL8169 based NICs with 32bit and 66MHz PCI bus. Also, we have measured the performance of MPI on a cc-NUMA server Altix 3700, for the comparison with its typical proprietary networking technology. See Figure 7 to see the diagram of Altix 3700 [8] used in the experiment.

We show the latency and the bandwidth of MPI with MHEL-CF128-T and MHEA28-XT in Figure 8 and 10. The latency of InfiniHost HCA is 3.99us, and the bidirectional bandwidth is 2907MB/s with MHEL-CF128-T, while the latency is 28.70us and the bidirectional bandwidth is 58MB/s with gigabit ethernet. The striping function works only for the messages of more than a certain size, and has no effect on small messages. It also works to reduce the latency.

The intranode latency is lower than the internode latency, but the bandwidth is also lower. It seems to be due to the implementation of MPI communication on shared memory architectures [7].

The characteristics of the bandwidth of MHEA28-XT are similar to those of MHEL-CF128-T. For bidirectional bandwidth, however, as shown in Figure 10, we have observed small performance degradation for large messages of more than 4MB. Although is is under investigation with Mellanox Technology, it can be considered to be due to the overflow of some memory buffer of the chipset. In this study, we have used MHEL-CF128-T for node 0 to 3 and MHEA28-XT for node 4 to 7, but we had not experienced the side effects caused by the degradation.

For the case with PCI-X bus, it is reported that the latency is about 6us, and the bidirectional bandwidth is 1877MB/s [10]. Therefore, we can see that the adoption of PCI Express brings significant performance enhancement. Furthermore, the results published by the Ohio State University [3] shows that the maximum bidirectional bandwidth achieved by MVAPICH is 2704MB/s obtaind with 3.4GHz EM64T Xeon servers. Our result with 2GHz Opteron servers outperforms it about 7%, which shows that our cluster records the highest interconnect bidirectional bandwidth ever achieved on PC clusters.

For the comparison, we show the results with Altix 3700 in Figure 11 and 12. We have used Intel Compiler 9.0 with -fast option, combined with the built-in MPI library.

Although it shows lower latency, the bandwidth is relatively small.

[3] http://nowlab.cis.ohio-state.edu/projects/mpi-iba/

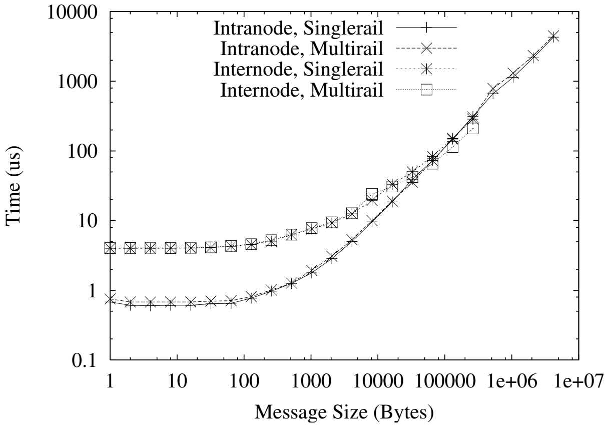

Figure 8: MPI Latency of InfiniHost HCA (MHEL-CF128-T) with different number of communication ports.

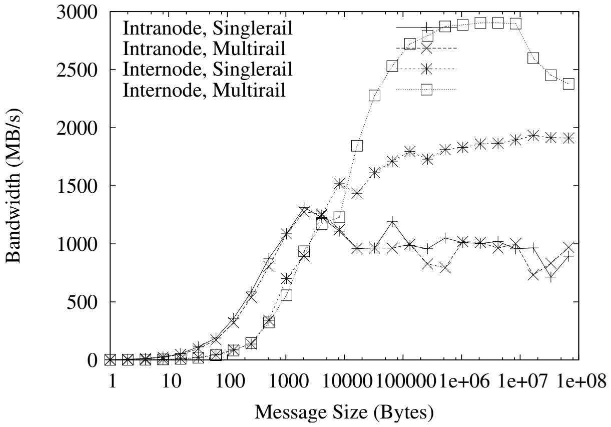

Figure 9: MPI Bidirectional Bandwidth of InfiniHost HCA (MHEL-CF128-T) with different number of communication ports.

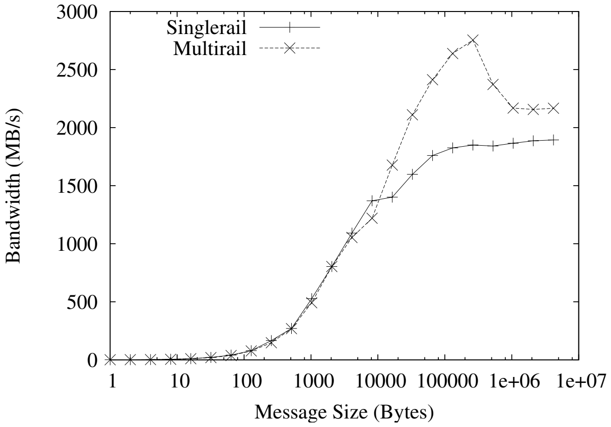

Figure 10: MPI Bidirectional Bandwidth of InfiniHost HCA (MHEA28-XT) with different number of communication ports.

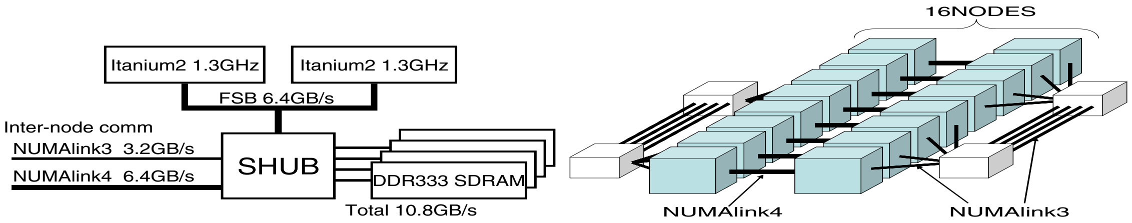

Figure 7: Diagram of Altix 3700 with SHUB Chipset.

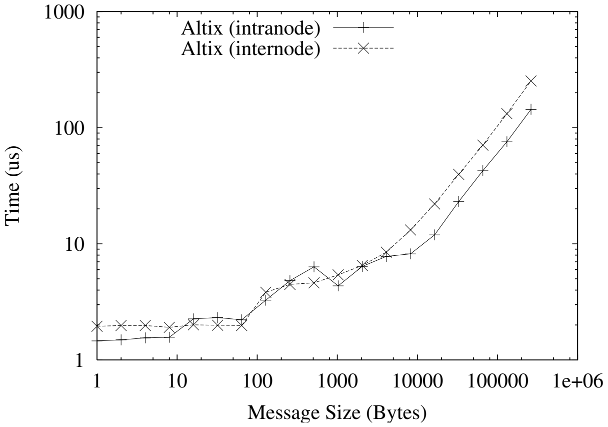

Figure 11: MPI Latency on SGI Altix 3700.

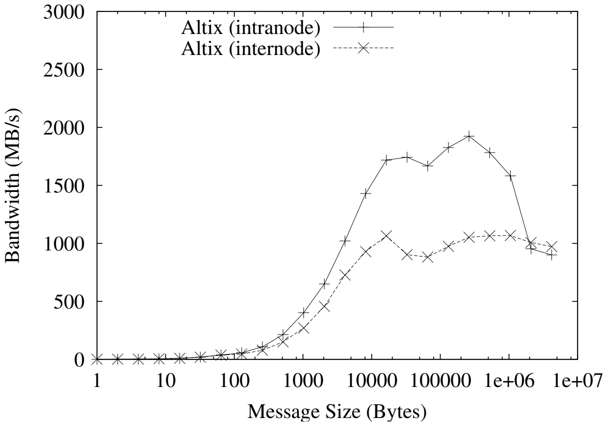

Figure 12: MPI Bidirectional Bandwidth on SGI Altix 3700.

3.1 STREAM Benchmark

For the next step, we have measured the memory bandwidth of both the Opteron cluster and the Altix 3700 using the STREAM benchmark [2] to measure the local memory bandwidth required for vector operations. Memory bandwidth is one of the most important measures for vector operations, and the systemwide global memory bandwidth is the barometer of the performance of parallel vector operations on scalar architectures. The STREAM benchmark evaluates the sustained bandwidth of a system using the following operations:

Table 1: STREAM benchmark types.

Benchmark	Operation	Bytes per iteration
`Copy`	a[i] = b[i]	16
`Scale`	a[i] = q * b[i]	16
`Add`	a[i] = b[i] + c[i]	24
`Triad`	a[i] = b[i] + q * c[i]	24

We have evaluated the performance of the MPI version using the multiple networks listed above. The results are shown in Figure 13. The problem size is 4M per process, that is about 91.6MB. The average performance is about 2.5GB/s per process, which is better than those of the OpenMP version on Altix 3700 shown in Figure 15.

For the reference, we show the result on a 2.8GHz dual Xeon server (IBM x335 with ServerWorks Grand Champion LE Chipset) based cluster connected with gigabit ethernet in Figure 16. Because of the lower local memory bandwidth, we cannot achieve sufficient global bandwidth on the Xeon clusters. See Figure 14 to see the diagram of x335.

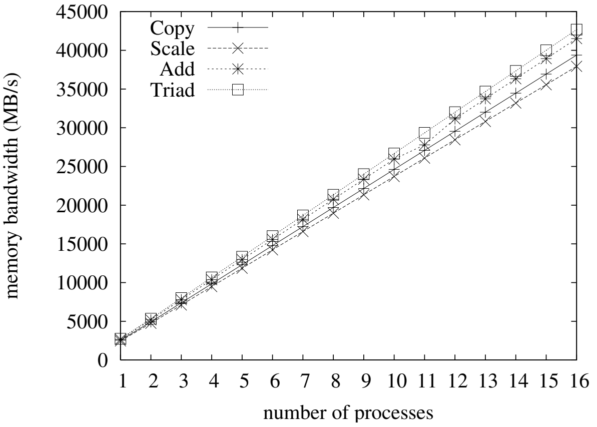

Figure 13: Speedup of STREAM benckmark results with two processes per node.

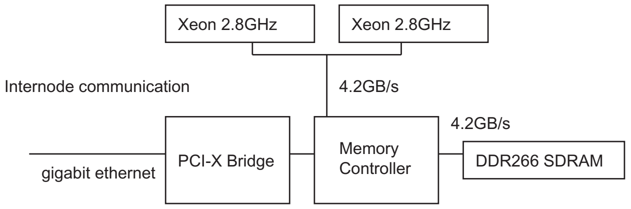

Figure 14: Diagram of Xeon node.

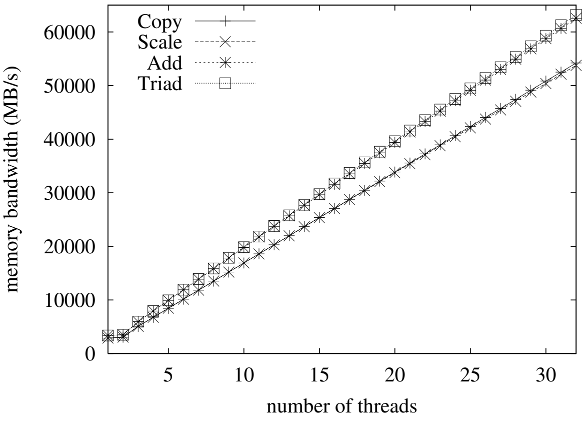

Figure 15: STREAM benchmark performance in MB/s with array size 80,000,000 on SGI Altix with memory affinity

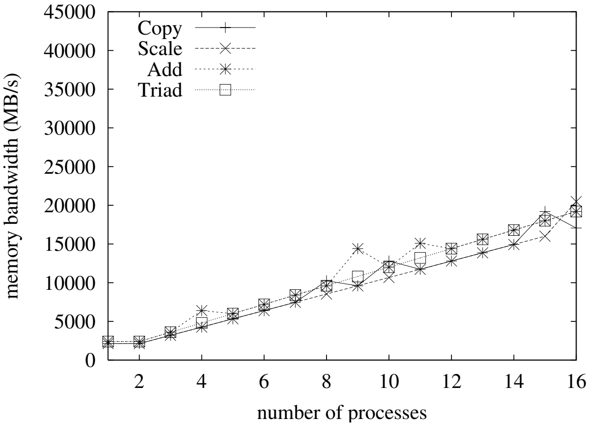

Figure 16: Speedup of STREAM benckmark results with two processes per node. (on a 2.8GHz dual Xeon cluster connected with GbE.)

3.2 NAS Parallel Benchmark

Based on these preparations, we have measured the performance of the conjugate gradient method using Class S to C of NAS Parallel Benchmark [13] 3.2 CG kernel to investigate the correlation between the performance of the iterative methods and the interconnects. The main loop of the CG kernel is shown in Figure 17. CG computes the smallest eigenvalue of a symmetric positive definite sparse matrix using the inverse iteration method and the conjugate gradient method. Although the NAS parallel benchmark is often used to measure the performance of parallel computers, it is a common situation that the CG benchmark cannot achieve sufficient performance, such as the case with Altix in Figure 18.

Although we can presume various reasons, we assume here the restriction of the bandwidth is the cause of the performance degradation with larger number of cluster nodes, and do some experiments based on the assumption. We show below the results with the single port of InfiniHost HCA and the ones with the dual ports.

From the results on the gigabit ethernet, we can see that the bandwidth affects the performance on the same cluster significantly. Furthermore, from the results on the InfiniHost HCA, we see the performance degradation with lower bandwidth, even with the same latency. We can conclude from the results that the bandwidth restricts the performance instead of the latency, and the network performance restricts the performance of the CG benchmark.

```
  vector r,p,q,z; /* working vector */
  /* solve A*x */
  real CGSOLVE(matrix A,vector x, real zeta) {
    int it; real alpha, beta, rho, rho0;
    r = x; p = x; z = 0.0; rho = x * x;
    for(it=0; it < NITCG; it++) {
      q = A*p;
      alpha = rho/(p * q);
      z += alpha * p;
      rho0 = rho;
      r += -alpha * q;
      rho = r*r;
      beta = rho / rho0;
      p += beta*r;
    }
    r = A*z - x;
    zeta = 1.0/x*z;
    x = (1.0/sqrt(z*z))*z;
    return sqrt(r*r);
  }
```

Figure 17: Algorithm of NPB CG Main Loop.

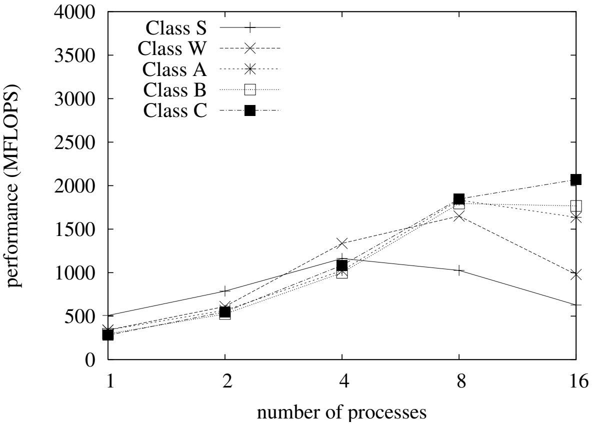

Figure 18: Speedup of NPB CG on SGI Altix 3700.

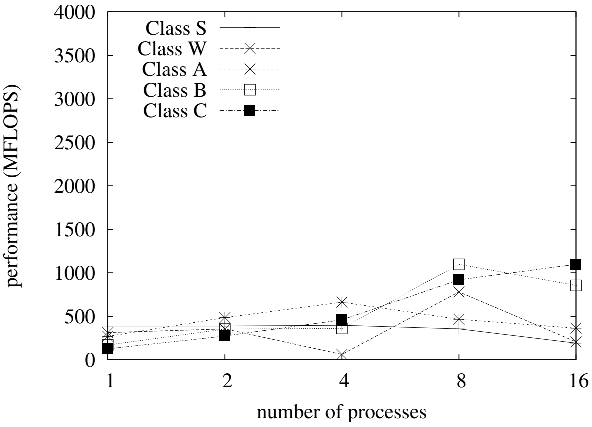

Figure 19: Speedup of NPB CG with one port of GbE.

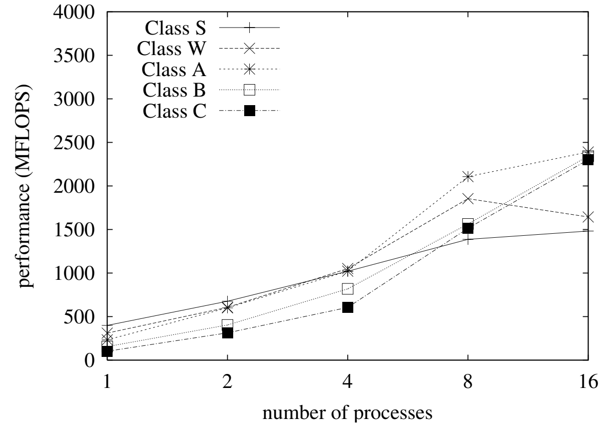

Figure 20: Speedup of NPB CG with one port of InfiniHost HCA.

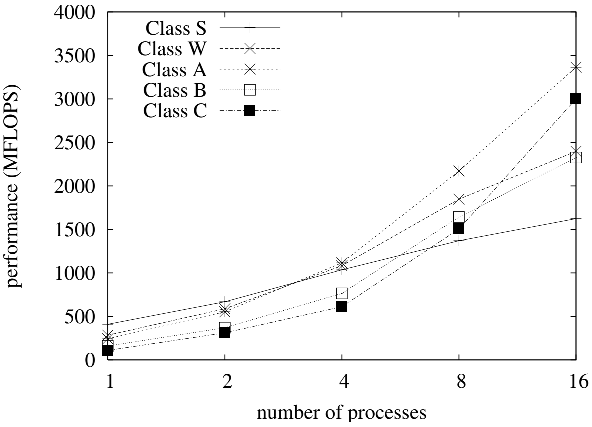

Figure 21: Speedup of NPB CG with two ports of InfiniHost HCA.

4 Discussion

This study has shown that the appropriate combination of COTS technologies can beat the performance of proprietary parallel computer systems, even for the real applications that requires high bandwidth. Especially, it is a useful observation for future high performance computing that shows the ability of building high performance interconnects using the PCI Express slots for graphic cards on the lowest cost motherboards. Each application has its own characteristics, and in sparse linear algebra operations, you have to choose an appropriate networking technology. In the field of scientific computing that requires the repetitive solutions of various discretized differential equations, we can say that the choice of the interconnect has the vital importance. This study gives a ground of the claim by showing the relation between the performance of interconnect and computing on the state-of-the-art COTS technology.

5 Related Work

Liu et al. [9] reports the merits of InfiniBand interconnect compared with other technologies for COTS clusters. Boku et al. [4] reports the limitations of the scalability of NPB CG kernel [13] on a GbE based large scale cluster, which partly motivates this study.

6 Concluding Remarks

In this study, we have built a PCI Express based PC cluster with COTS technologies, and compared the results with those on SGI Altix 3700. We have shown that the bandwidth between the cluster nodes is the most critical factor for the scalable parallel implementation of the sparse matrix operations from the results obtained from the experiments.

Acknowledgment

The author is grateful for supports and advices from many people at Mellanox Technologies, inc. and Altima Co.,Ltd, and Dr. Panda at Ohio State University. This study is partially supported by MEXT Grant-in-aid for Scientific Research 16016225, and a grant from CREST, JST.

References

[1] *Lis: A Library of Iterative Solvers for Linear Systems.*
http://ssi.is.s.u-tokyo.ac.jp/lis/.

[2] *STREAM: Sustainable Memory Bandwidth in High Performance Computers.*
http://www.cs.virginia.edu/stream/.

[3] D. Bailey, T. Harris, W. Saphir, R. van der Wijngaart, A. Woo, and M Yarrow. The NAS Parallel Benchmarks 2.0. *The International Journal of Supercomputer Applications*, 2858:500–510, 2003.

[4] T. Boku, Y. Hotta, S. Matsuoka, H. Nakamura, H. Nakashima, M. Sato, and D. Takahashi. MegaProto: 1 TFlops/10 kW Rack Is Feasible Even with Only Commodity Technology. In *Proceedings of Supercomputing Conference 2005*, 2005.

[5] Greg Burns, Raja Daoud, and James Vaigl. LAM: An Open Cluster Environment for MPI. In *Proceedings of Supercomputing Symposium*, pages 379–386, 1994.

[6] J. I. Hennessy and D. A. Patterson. *Computer Architecture: A Quantitative Approach, Third Edition.* Morgan Kaufmann, 2003.

[7] H. W. Jin, S. Sur, L. Chai, and D. K. Panda. LiMIC: Support for High-Performance MPI Intra-Node Communication on Linux Cluster. In *Proceedings of the International Conference on Parallel Processing*, 2005.

[8] H. Kotakemori, H. Hasegawa, T. Kajiyama, A. Nukada, R. Suda, and A. Nishida. Performance Evaluation of Parallel Sparse Matrix–Vector Products on SGI Altix3700. In *Proceedings of First International Workshop on OpenMP (IWOMP2005)*, 2005. in press.

[9] J. Liu, B. Chandrasekaran, W. Jiang J. Wu, S. P. Kini, W. Yu, D. Buntinas, P. Wyckoff, and D. K. Panda. Performance Comparison of MPI implementations over Infiniband, Myrinet and Quadrics. In *Proceedings of the International Conference on Supercomputing '03*, 2003.

[10] J. Liu, A. Vishnu, and D. K. Panda. Building Multirail InfiniBand Clusters: MPI-Level Design and Performance Evaluation. In *Proceedings of the International Conference on Supercomputing '04*, 2004.

[11] G. F. Pfister. *In Search of Clusters.* Prentice-Hall, second edition, 1998.

[12] S. Sumimoto, A. Naruse, K. Kumon, K. Hosoe, and T. Shimizu. PM/InfiniBand-FJ: a high performance communication facility using InfiniBand for large scale PC clusters.

[13] Rob van der Wijngaart. The NAS Parallel Benchmarks 2.4. Technical Report NAS-02-007, NASA, 2002.