Can Memory-Less Network Adapters Benefit Next-Generation InfiniBand Systems?

Sayantan Sur, Abhinav Vishnu, Hyun-Wook Jin, Wei Huang and D. K. Panda

Technical Report
OSU-CISRC-5/05-TR31

Can Memory-Less Network Adapters Benefit Next-Generation InfiniBand Systems? *

Sayantan Sur Abhinav Vishnu Hyun-Wook Jin Wei Huang
Dhabaleswar K. Panda

Network-Based Computing Laboratory
Department of Computer Science and Engineering
The Ohio State University
{surs, vishnu, jinhy, huanwei, panda}@cse.ohio-state.edu

Abstract

InfiniBand is an emerging high-performance interconnect. It is gaining popularity because of its high performance and open standard. Recently, PCI-Express, which is the third generation high-performance I/O bus used to interconnect peripheral devices, has been released. The third generation of InfiniBand adapters allow applications to take advantage of PCI-Express. PCI-Express offers very low latency access of the host memory by network interface cards (NICs). Earlier generation InfiniBand adapters used to have an external DIMM attached as local NIC memory. This memory was used to store internal information. This memory increases the overall cost of the NIC. In this paper we design experiments, analyze the performance of various communication patterns and end applications on PCI-Express based systems, whose adapters can be chosen to run with or without local NIC memory. Our investigations reveal that on these systems, the memory fetch latency is the same for both local NIC memory and Host memory. Under heavy I/O bus usage, the latency of a scatter operation increased only by 10% and only for message sizes 1B - 4KB. These memory-less adapters allowed more efficient use of overall system memory and showed practically no performance impact (less than 0.1%) for the NAS Parallel Benchmarks on 8 processes.

1 Introduction

InfiniBand Architecture [6] is an industry standard which offers low latency and high bandwidth as well as many advanced features such as Remote Direct Memory Access (RDMA), atomic operations, multicast and QoS. Currently, InfiniBand products in the market can achieve a latency of few microseconds for small messages and a bandwidth of up to 4 GB/s aggregate (using dual port InfiniBand 4X Host Channel Adapters (HCAs) or NICs). As a result it is becoming increasingly popular for building high performance clusters.

Recently, PCI-Express [12] has been introduced as the next generation local I/O interconnect. Unlike PCI, PCI-Express uses a serial, point-to-point interface. Compared with PCI, PCI-Express can achieve lower latency by allowing I/O devices to be connected directly to the memory controller. More importantly, it can deliver scalable bandwidth by using multiple lanes in each point-to-point link.

The HCA needs some memory to keep some internal information for operation. This is usually kept in an external attached memory. External attached memory at the HCA has the potential to reduce the local I/O bus traffic significantly. In addition to that, the attached memory might have lower access latency. However, providing the additional memory at the HCA increases the overall cost of the system. Not only the cost of the memory, but the local memory requires on-chip real estate, which leads to costlier fabrication.

The advent of PCI-Express (which has very low access latency) has raised a new and relevant question:

- Can memory-less InfiniBand network adapters deliver the same performance (micro-benchmarks and end applications) compared to adapters with memory?
- Do these network adapters demand more system memory?

In this paper we carry out a detailed performance analysis of the third generation InfiniBand HCAs which can support a Memory Free mode (MemFree) and a mode with memory (Mem) [1]. This mode can actually support operation in the case where there is no local memory attached with the HCA. In this mode,

*This research is supported in part by Department of Energy's grant #DE-FC02-01ER25506, National Science Foundation's grants #CCR-0204429 and #CCR-0311542.

the host memory is used as a replacement for local HCA memory. In both modes of operation, the HCA maintains a cache of recently used entries (called as InfiniHost Context Memory (ICM) Cache).

Our experimental results reveal that for PCI-Express based systems, the memory access times of both local NIC memory and Host memory are similar. In addition, the basic ICM cache miss penalty is the same for both Mem and MemFree modes. Further, the translation entry cache misses do not cause any degradation in bandwidth performance. However, under heavy I/O bus usage, the performance of the MemFree operation can drop up to 10% but only for small messages (1 Byte - 4KB). Finally, our performance evaluation of some of the NAS Parallel Benchmarks [2] suggests practically no performance degradation (less than 0.1%) for Mem and MemFree modes. Hence, our experiments reveal that these memory-less network adapters can achieve the same performance as adapters with memory.

Also, our experiments with memory usage indicate that the MemFree mode consumes up to 18 MBytes more host memory than the Mem mode for 128 connections between two processes.

Since the current Mem mode HCAs typically have 128 MBytes of memory each, these experiments indicate that next generation InfiniBand systems can be designed with MemFree HCAs with a very small increase in host memory size and with an effective reduced memory size (considering both host and nic memory) compared to systems using Mem mode HCAs.

Hence, our experiments reveal that these memory-less network adapters do not demand more overall system memory.

The rest of the paper is organized as follows: In Section 2, we provide a brief overview of InfiniBand, PCI-Express and the third generation Mellanox InfiniBand adapters. In Section 3, we provide a motivating example of the reduction of host memory access time with next generation PCI-Express. In Section 4, we describe our microbenchmark evaluation tests and provide the corresponding results and their analysis. In Section 5 we present performance of NAS Parallel Benchmarks. In Section 6, we describe the related work in this field. Finally, this paper concludes in Section 7.

2 Background

2.1 InfiniBand Overview

The InfiniBand Architecture [6] defines a System Area Network for interconnecting both processing nodes and I/O nodes. In an InfiniBand network processing nodes and I/O nodes are connected to the fabric by Channel Adapters (CA). The Channel Adapters on processing nodes are called Host Channel Adapters (HCAs). A queue-based model is used in InfiniBand. A Queue Pair (QP) consists of two queues: a send queue and a receive queue. Communication operations are described in the Work Queue Requests (WQR), or descriptors, and submitted to the work queue. The completion of WQRs is reported through Completion Queues (CQs). InfiniBand Architecture supports both channel semantics and memory semantics. In channel semantics, send/receive operations are used for communication. In memory semantics, InfiniBand provides Remote Direct Memory Access (RDMA) operations, including RDMA Write and RDMA Read. InfiniBand also supports different classes of transport services. In current products, Reliable Connection (RC) service and Unreliable Datagram (UD) services are supported.

2.2 PCI-Express Architecture

PCI Express is the third generation high performance I/O bus used to interconnect peripheral devices in applications such as computing and communication platforms [12].

PCI has been the standard local I/O bus technology for the last ten years. It uses a parallel bus at the physical layer and a load/store based software usage model. Since its introduction, both PCI bus frequency and bus width have been increased to satisfy the ever-increasing I/O demand of applications. Later, PCI-X was introduced as an extension to PCI. PCI-X is backward compatible with PCI in terms of both hardware and software interfaces. It delivers higher peak I/O performance and efficiency compared with PCI.

Recently, PCI Express [12] technology was introduced as the next generation I/O bus. Unlike traditional I/O buses such as PCI, PCI Express uses a high performance, point-to-point, and serial interface. In PCI and PCI-X architectures, bus frequency and width are limited due to signal skews in the underlying parallel physical interface. Further, a bus is shared among all devices connected to it. Therefore, PCI and PCI-X have limited bandwidth scalability. To achieve better scalability, PCI Express links can have multiple lanes, with each lane delivering 250 MB/s bandwidth in each direction. For example, an 8x (8 lanes in each link) PCI Express channel can achieve 2 GB/s bandwidth in each direction, resulting in an aggregate bandwidth of 4 GB/s. In comparison, a 64 bit/133 MHz PCI-X bus can only achieve around 1 GB/s bandwidth at most. In PCI or PCI-X based systems, I/O devices are typically connected to the memory controller through an additional I/O bridge. In PCI Express based systems, I/O devices can be connected directly to the memory controller through PCI Express links. This results in improved I/O performance.

2.3 InfiniHost III HCA Architecture

The InfiniHost MT25218 is the Mellanox third generation channel adapter [1]. It is a single-chip, dual port InfiniBand 4X HCA. The HCA architecture is shown in Figure 1. The HCA keeps all the InfiniBand related data-structures in a virtual memory area called

InfiniHost Context Memory (ICM). The ICM can be physically placed either in:

(a) DDR external memory (these are external DIMMs attached on the HCA). In this paper, we refer to these external DIMMs as "NIC memory".

(b) In the physical main memory of the system. This memory is accessed over PCI-Express. In this paper we refer to this memory as "Host memory".

In addition to the NIC memory, the HCA has a limited amount of memory on chip, which it uses to keep recently used data structures. We refer to this as the ICM Cache. If required data is missing in the ICM cache, the hardware brings the required data from the ICM physical location.

The HCA supports operation both with and without the attached NIC memory. These two operating modes will be referred to as "Mem" mode and "MemFree" mode respectively. In the MemFree mode the firmware on the HCA can use a part of the Host memory to allocate ICM. The ICM is usually required by the HCA to store Queue Pair (QP) Context/Completion Queue (CQ) and Address Translation Table entries.

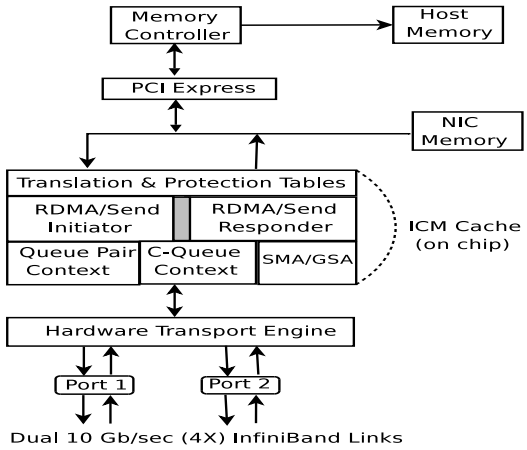


Figure 1. MT25218 HCA Architecture, courtesy Mellanox [1]

3 MemFree on PCI-Express: Is it a good idea?

The local NIC memory can potentially provide fast access to InfiniBand data structures during the critical data movement path. However, providing additional memory increases the overall cost of the HCA. It is not very clear in the case of PCI-Express systems (which have low-latency access to host memory) that additional NIC memory can still be beneficial.

In this section we evaluate the memory access times of both the Host memory and the NIC memory by the HCA DMA engines in the context of next-generation PCI-Express systems. Our experiment involves one process which has two QPs connected over RC. It performs data-movement in two ways. In the first method, the process registers two buffers in the Host memory separately. Then, it instructs the HCA (RDMA Write) to perform a DMA from one buffer to the another. It is to be noted that the adapter does not send the message out to the network, since both the QPs are bound to the same port of the HCA. In the second method the process allocates and registers two buffers in the NIC memory. Then, it again performs a RDMA Write to and from these buffers. The time for these two DMAs is noted. This time divided by two gives the one-way memory access latency for both Host and NIC memory. The entire experiment is performed on both PCI-X and PCI-Express based systems. The exact platform configuration is mentioned in Section 4.

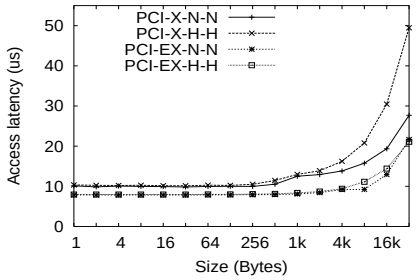


Figure 2. Host and NIC Memory Access Latency

The results are shown in Figure 2. We observe from the figure that for PCI-X based systems, the NIC memory access is faster than Host memory access (comparing lines PCI-X-N-N and PCI-X-H-H). Therefore, for PCI-X based systems attaching memory to the HCAs can prove to be beneficial. On the other hand, we observe from the PCI-Express results that the Host memory access is as fast as the NIC memory access (comparing lines PCI-EX-N-N and PCI-EX-H-H). Hence, regardless of where the InfiniBand data structures are kept, either Host or NIC memory, potentially they can be retrieved as quickly.

4 Microbenchmark Level Evaluation

In this section, we perform microbenchmark level evaluation of both Mem and MemFree modes of operation. Our experiments focus on the important elements of the ICM cached memory, namely the QP context and the virtual to physical address translation entries. Our experiments realize various communication scenarios and we analyze the performance of Mem and MemFree modes. Our experimental platform consists of 4 dual Intel Xeon 3.2 GHz, EM64T systems. These nodes are equipped with 786 MB of DDR2 memory. The nodes have MT25128 Mellanox HCAs, and IB Golden CD version 1.7.0 [1]. The operating system on these nodes is Red Hat Enterprise Linux AS release 3 (Taroon Update 2). The kernel version used is 2.4.21-15.EL.

4.1 ICM Cache Miss Penalty Comparison

As mentioned in Section 2, the ICM is used by the HCA to store QP context information. This QP context information is required when a QP is accessed for a work request operation. In this experiment we compare the cache miss penalty for Mem and MemFree modes of operation. Our experiment is designed as follows:

There are two processes which have n RC QPs between them. The processes are executing a simple ping-pong latency test on each QP. The QPs are used in a cyclic manner. The latency for the first QP is recorded. Since the QPs are being used in a cyclic manner we can be sure that when the first QP is used after the nth QP, the HCA will incur an ICM cache miss (provided n is a large enough number). We conduct the experiment for two values of n: 1 (100% cache hit) and 128 (0% cache hit). The results are shown in Figure 3(a).

We observe from Figure 3(a) that the latency for 128 QP for both Mem and MemFree modes are the almost same. Additionally, the 128 QP latency is much higher than the 1 QP latency. This proves that there has been an ICM cache miss for the 128 QP case. However, the cache miss penalty for both Mem and MemFree modes is the same. On the occurrence of a cache miss, the HCA must retrieve the appropriate QP context, which might be in NIC memory (in Mem mode) or Host memory (in MemFree mode). As we have seen in Section 3, the access times of both are same on PCI-Express systems. Hence, this experiment proves that even if the HCA operates in a MemFree mode, there is no impact on latency of ICM cache misses.

4.2 Effect of I/O Bus Transactions on ICM Cache Miss Penalty

In this section we design an experiment to see how the ICM cache miss penalty is affected by the location of the ICM (either NIC or Host memory) under heavy I/O bus usage. Our experiment is as follows:

There are two processes which have n RC QPs between them. One process is executing a buffer scatter across the QPs to the other process. This process sends a different registered buffer on all the QPs. As in the earlier experiment, the QPs are accessed cyclically. The difference is that the sending process posts descriptors for all n QPs back-to-back and then waits for the completion of all of them. This sort of a communication patterns makes sure that when the ICM cache misses occur (i.e., when the HCA wants to access a QP context after a doorbell ring by the descriptor post), there are other pending I/O bus transactions. The time measured is for the scatter operation across all QPs. The results are shown in Figure 3(b).

We observe from Figure 3(b) that the MemFree mode performs around 10% worse than the Mem mode for messages up to 8KB. This is because when the ICM cache misses (for QP context) occur, the I/O bus is already under use by (a) Descriptor Post (b) DMA for previous ICM cache miss (c) DMA for a message. We also notice that for larger messages, there is no difference between the Mem and MemFree mode. This may be due to various reasons. The I/O bus has flexibility in re-ordering some DMA transactions. As the message sizes increase, the I/O bus might choose to satisfy the ICM cache miss DMA first (since the QP context information might be much smaller than the message size). Additionally, another reason can be the scheduling of DMA requests by the HCA.

4.3 ICM Cache Misses for Address Translation

The ICM is used to keep the virtual to physical address translation entries. There is one entry per page. The entries contain the physical address corresponding to a virtual address. The HCA needs to look up this entry to find out the actual destination before initiating the DMA on the local I/O bus. In the previous experiment 4.2, we have seen the case where the cache miss can have both QP context misses as well as translation entry misses. In this section we design an experiment to find out the impact ICM cache misses when accessing only translation entries.

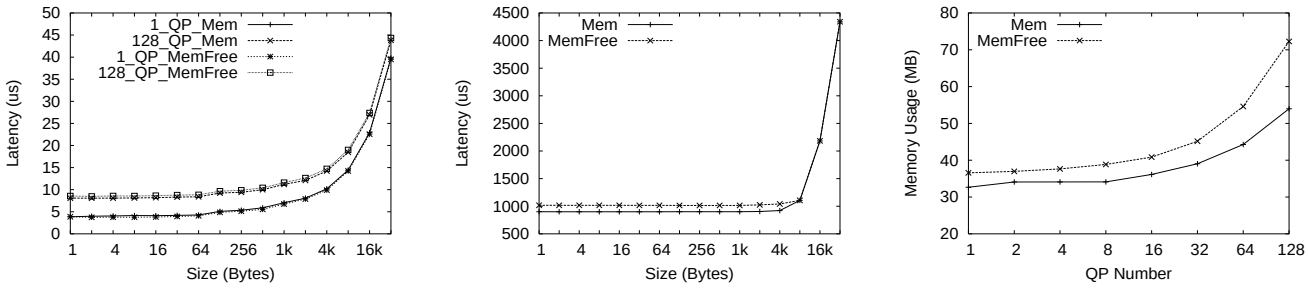
In this experiment, there are two processes which are connected over one RC QP. These processes are conducting a bandwidth test, but the buffers are reused only for a certain fraction [3]. The idea is that for low buffer reuse rate, we should induce cache misses for the translation entries. The results are presented in Table 1. The message size used is 64KB.

Table 1. Bandwidth with ICM Cache Misses for Translation Entries

Buffer Reuse (%)	MemFree	Mem
100	918 MB/s	926 MB/s
75	918 MB/s	925 MB/s
50	919 MB/s	927 MB/s
25	918 MB/s	926 MB/s
0	917 MB/s	922 MB/s

As we can see from Table 1 the performance of Mem and MemFree modes is similar (less than 1% difference). This is because PCI-Express is bi-directional in nature. On the receiver side, the DMA for the destination buffer can happen in parallel with the fetching of the translation table entry for the subsequent physical page. Hence, the MemFree mode can perform equal to the Mem mode.

It is to be noted that this parallelization is possible since there is only one QP between the processes. If there is a QP context miss, the QP context will have to be fetched across the PCI-Express bus, before the DMA for the buffer is possible. In this experiment, DMA of pages for which the physical translation is known can proceed, while address translation entries for subsequent pages are fetched over the bi-directional



(a) ICM Cache Miss Penalty

(b) Effect of I/O Bus Transactions on ICM Cache Miss Penalty

(c) Memory Usage

Figure 3. Microbenchmark Level Evaluation Results

PCI-Express bus. In that respect a QP context miss might be more critical than a miss for the translation entry.

4.4 Host Memory Usage

The HCA when operating in MemFree mode keeps the ICM in the Host memory. This can lead to increased usage of the Host memory. Another factor to note is that all the ICM memory should be kept in registered (pinned) pages. In case the system is running low on memory, it cannot swap out these pages. We conduct an experiment in which two processes establish n RC QPs. After establishing QPs both the processes go to sleep. Then we record the virtual memory allocated for these processes. The UNIX utility *pmap* is used for finding out this memory. The results are shown in Figure 3(c).

We observe from the Figure 3(c) that the Host memory usage for the MemFree mode of operation is higher than that of the Mem mode of operation. This indicates that as the QPs are being allocated more and more Host memory is being utilized. It is to be noted that the memory usage number for one QP includes all the memory occupied by the process including various libraries. This memory is not necessarily registered. This should not be interpreted as the minimum amount of memory required for RC connection.

Our experiment indicates that the MemFree mode consumes up to 18 MBytes more Host memory than the Mem mode for 128 QPs between two processes. Since the current Mem mode NICs typically have 128 MBytes of memory each, these experiments indicate that next generation InfiniBand systems can be designed with MemFree NICs with a very small increase in Host memory size and with an effective reduced memory size (considering both Host and NIC memory) compared to systems using Mem mode NICs. Further, the MemFree HCAs can boost overall memory efficiency. Instead of just dedicating 128 MB memory at the HCA, we can have other applications use that memory (if it is not needed by the HCA).

5 NAS Parallel Benchmark Performance

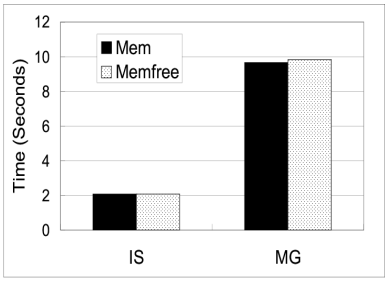
In this section we present some real world application performance numbers. We evaluate the performance of Mem and MemFree mode of operation on the NAS Parallel Benchmarks [2]. These benchmarks were run using MVAPICH [9]. MVAPICH is an open-source implementation of MPI [8] over InfiniBand. It is based on the ADI layer of MVICH [5]. It was derived from MVICH [7]. The results are shown in Figure 4.

We observe from the Figure 4, that the performance of the Mem and the MemFree modes is practically the same for IS, MG, LU and CG on 8 processes. The MemFree mode is able to offer the practically the same performance (less than 0.1% difference) as the Mem mode.

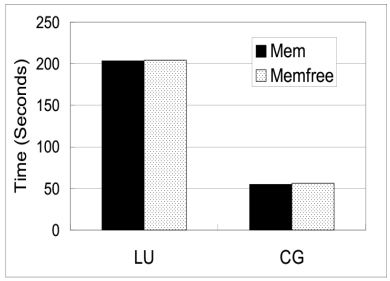
6 Related Work

There has been several research works on the performance of NIC memory and how to take advantage of the NIC memory for various interconnects. Petrini et al.[10, 11] have studied the impact of NIC memory on the Quadrics interconnect. Wu et al. [13] suggest various techniques to improve communication performance by utilizing NIC memory on the Mellanox HCAs. Geoffray et al. [4] have proposed an off-processor I/O architecture to move data between the disk and the Myrinet NIC memory directly over the PCI bus.

However, our work is significantly different from the above. We are trying to evaluate the performance of NIC and Host memory for holding HCA internal information. But the above works focus on improving communication performance by using NIC memory buffers for application use. In addition, we focus on next generation PCI-Express systems.



(a) NAS: IS and MG Performance



(b) NAS: LU and CG Performance

Figure 4. NAS Parallel Benchmark Results

7 Conclusions and Future Work

In this paper we design several new microbenchmarks to evaluate the performance of Mem and MemFree modes of operation of the Mellanox MT25128 HCAs. Our investigation reveals that the memory access times of both local NIC memory and Host memory are similar on PCI-Express. In addition, the basic ICM cache miss penalty (QP context miss) is the same for both Mem and MemFree modes. Further, the translation entry cache misses do not cause any degradation in bandwidth performance. However, under heavy I/O bus usage, the performance of the MemFree operation can drop up to 10% but only for small messages (1B - 4KB). Finally, our performance evaluation of some of the NAS Parallel Benchmarks [2] suggests practically no performance degradation (less than 0.1%) for the MemFree mode. Thus, the memory-less network adapters are able to provide same performance as those with memory for various microbenchmarks and end applications.

Our experiments with memory usage reveal that the MemFree mode consumes up to 18 MBytes more Host memory than the Mem mode for 128 QPs between two processes. Since the current Mem mode HCAs typically have 128 MBytes of memory each, these experiments indicate that next generation InfiniBand systems can be designed with MemFree HCAs with a very small increase in Host memory size and with an effective reduced memory size (considering both host and nic memory) compared to systems using Mem mode NICs.

We plan to continue working in this direction. We plan to design much larger scale microbenchmarks to investigate the scalability of the MemFree HCA design for tera-scale clusters. We also plan to run more real world applications and application benchmarks on larger scale InfiniBand clusters.

References

[1] Mellanox Technologies. http://www.mellanox.com.

[2] D. H. Bailey, E. Barszcz, J. T. Barton, D. S. Browning, R. L. Carter, D. Dagum, R. A. Fatoohi, P. O. Frederickson, T. A. Lasinski, R. S. Schreiber, H. D. Simon, V. Venkatakrishnan, and S. K. Weeratunga. The NAS parallel benchmarks. volume 5, pages 63–73, Fall 1991.

[3] B. Chandrasekaran, P. Wyckoff, and D. K. Panda. MIBA: A Micro-benchmark Suite for Evaluating InfiniBand Architecture Implementation. In *Performance TOOLS*, 2003.

[4] P. Geoffray. OPIOM: Off-processor i/o with myrinet. *Future Generation Computing System*, 18(4):491 – 499, 2002.

[5] W. Gropp, E. Lusk, N. Doss, and A. Skjellum. A High-Performance, Portable Implementation of the MPI, Message Passing Interface Standard. Technical report, Argonne National Laboratory and Mississippi State University.

[6] InfiniBand Trade Association. InfiniBand Trade Association. http://www.infinibandta.com.

[7] Lawrence Berkeley National Laboratory. MVICH: MPI for Virtual Interface Architecture. http://www.nersc.gov/research/FTG/mvich/index.html, August 2001.

[8] Message Passing Interface Forum. *MPI: A Message-Passing Interface Standard*, Mar 1994.

[9] Network-Based Computing Laboratory. MPI over InfiniBand Project. http://nowlab.cis.ohio-state.edu/projects/mpi-iba/.

[10] F. Petrini, W. chun Feng, A. Hoise, S. Coll, and E. Frachtenberg. The quadrics network: High-performance clustering technology. *IEEE Micro*, 22(1):46 – 57, 2002.

[11] F. Petrini, S. Coll, E. Frachtenberg, and A. Hoisie. Performance evaluation of the quadrics interconnection network. *Journal of Cluster Computing*, 6(2):125 – 142, 2003.

[12] The PCI Special Interest Group. PCI, PCI-X and PCI Express Architecture. http://www.pcisig.com.

[13] J. Wu, A. Mamidala, and D. K. Panda. Can NIC Memory in InfiniBand Benefit Communication Performance? A Study with Mellanox Adapter. Technical Report OSU-CISRC-4/04-TR20, April 2004.