10-Gigabit iWARP Ethernet: Comparative Performance Analysis with InfiniBand and Myrinet-10G

Mohammad J. Rashti Ahmad Afsahi
Department of Electrical and Computer Engineering
Queen's University, Kingston, ON, CANADA K7L 3N6
mohammad.rashti@ece.queensu.ca ahmad.afsahi@queensu.ca

Abstract

iWARP is a set of standards enabling Remote Direct Memory Access (RDMA) over Ethernet. iWARP supporting RDMA and OS bypass, coupled with TCP/IP Offload Engines, can fully eliminate the host CPU involvement in an Ethernet environment. With the iWARP standard and the introduction of 10-Gigabit Ethernet, there is now an alternative path to the proprietary interconnects for high-performance computing, while maintaining compatibility with existing Ethernet infrastructure and protocols.

Recently, NetEffect Inc. has introduced an iWARP–enabled 10-Gigabit Ethernet Channel Adapter. In this paper we assess the potential of such an interconnect for high-performance computing by comparing its performance with two leading cluster interconnects, InfiniBand and Myrinet-10G. The results show that the NetEffect iWARP implementation achieves an unprecedented latency for Ethernet, and saturates 87% of the available bandwidth. It also scales better with multiple connections. At the MPI level, iWARP performs better than InfiniBand in queue usage and buffer re-use.

1. Introduction

Interconnection networks and communication system software play a significant role in achieving high performance in clusters. In this regard, several contemporary interconnects such as *Myrinet* [20], *Quadrics* [3], and *InfiniBand* [12] have been introduced in the past decade. Such interconnects are the backbone for scalable high-performance clusters, offering extremely low latency and high bandwidth communication. Their main drawbacks are cost and their incompatibility with existing Ethernet infrastructure.

Currently, there are two trends in the high-performance networking community to bridge the performance, cost, and compatibility gap between cluster specific interconnects and Ethernet [2]. For instance, Myrinet has introduced the *Myri-10G* Network Interface Cards (NICs) supporting both 10-Gigabit Ethernet and 10-Gigabit Myrinet. Recently, they have ported their messaging software, MX-10G (designed for 10-Gigabit Myrinet), to support the Ethernet standard. This way, they are trying to address the short message latency problem in 10-Gigabit Ethernet networks. On the other hand, the introduction of 10-Gigabit Ethernet, and the aggressive push toward utilizing advanced techniques such as *TCP/IP Offload Engines (TOEs)*, *OS bypass,* and *Remote Direct Memory Access* (RDMA) over Ethernet, has made emerging high-performance Ethernet look very promising in bridging the performance gap.

RDMA is a one-sided operation, allowing direct data transfer from the source buffer to the remote destination buffer without the host CPU intervention or intermediary copies. This inherent *zero-copy* feature of RDMA, in concert with the OS bypass mechanism, effectively helps save CPU cycles and memory bandwidth. RDMA has been used to improve the performance of point-to-point communications over InfiniBand [26] and collective communications on Quadrics QsNet^{II} [22].

To enhance Ethernet performance, the NIC hardware and its messaging layer must support RDMA over TCP/IP, OS bypass and TOE. The RDMA consortium [23] has developed a set of standard extensions to TCP/IP and Ethernet to eliminate the host CPU overhead. These specifications are collectively known as *iWARP*. Such a standard provides an alternative path to non-Ethernet interconnects for high-performance computing, while maintaining compatibility with the existing Ethernet infrastructure and protocols.

Recently, NetEffect Inc. has introduced the first iWARP–enabled 10-Gigabit Ethernet Channel Adapter. It is the focus of this paper to assess the potential of such an interconnect for high-performance computing in comparison with two leading cluster interconnects. We have done an extensive performance analysis of the NetEffect iWARP, Mellanox InfiniBand, and Myricom

1-4244-0910-1/07/$20.00

Myri-10G at the user level and *Message Passing Interface* (MPI) level [17]. To the best of our knowledge, this is the first, detailed comparative study of these interconnects.

Our evaluation at the user-level includes basic latency and bandwidth results for both single and multiple active connections. At the MPI layer, we present and analyze the latency and bandwidth, MPI latency overhead over user-level, *LogP* parameters [14], buffer reuse, and MPI queue usage results. The results show a significant improvement in Ethernet performance, and remarkable multi-connection performance scalability. The NetEffect iWARP implementation achieves a high level of performance in MPI queue usage, and buffer re-use impact.

After briefly introducing the networks and iWARP in Section 2, we review related work in Section 3. The experimental framework is detailed in Section 4. Section 5 discusses the user-level performance results. In Section 6, we analyze the MPI implementation. Section 7 concludes the paper and discusses plans for future work.

2. Background

2.1. InfiniBand Architecture Overview

InfiniBand (IB) offers switched point-point communication and message based semantics. Each IB sub-network consists of end nodes and switches managed by a central subnet-manager [12]. End nodes use Host Channel Adapters (HCA) to connect to the network. The InfiniBand standard uses IB verbs as the lowest software layer to access the HCA. The verbs layer has queue pair (QP) based semantics, in which processes post send or receive work requests (WR) to a QP. A QP consists of a send queue and a receive queue. InfiniBand verbs support four communication models: Reliable Connection (RC), Unreliable Connection (UC), Reliable Datagram (RD) and Unreliable Datagram (UD) [12] (we use the RC mode in our verbs layer tests). IB verbs require memory registration prior to using them for communication.

2.2. Overview of Myrinet-10G Networks

Recently, Myricom [20] has introduced its 10-Gigabit Myri-10G products. The physical links of Myri-10G are 10-Gigabit Ethernet, but the NICs, switches, and MX-10G software support both Ethernet and Myrinet protocols at the Data Link level. Users can run MPI applications over the Myrinet switch using MX-10G over Myrinet (MXoM) or through the 10-Gigabit Ethernet switches using MX-10G over Ethernet (MXoE). The MX-10G library has semantics close to MPI. Basic MX-10G communication primitives are non-blocking send and receive operations that are directly used in the implementation of MPI communication primitives. Although MX-10G does not require explicit memory registration, it uses such a mechanism internally.

2.3. iWARP and NetEffect RNIC

The iWARP specification proposes a set of descriptive interfaces at the top layer, called iWARP verbs [11, 9]. Verbs provide a user level abstract interface for direct access to the *RDMA enabled NIC* (RNIC), offering direct data transfer ability and OS bypass using RDMA. iWARP verbs have QP based semantics, similar to InfiniBand. A lower layer (TCP layer) connection is also established as the data stream between two endpoints. This means that iWARP offers connection oriented semantics. The verbs also require user buffers to be locked down before the data transfer takes place.

Below the verbs layer, there is the *RDMA Protocol* (RDMAP) layer that is responsible for performing RDMA operations and supplying communication primitives for remote memory access calls in verbs. Below that is the *Direct Data Placement* (DDP) layer [25], which is used for the direct transfer of data between the user buffer and the iWARP RNIC. DDP supports two types of data placements: *tagged* and *untagged*. For the tagged model, the operation source provides a reference to the data buffer address while in the untagged model, the operation target specifies the data buffer address. The next layer, the *Marker PDU Aligned* (MPA) layer [6] is used to assign boundaries to DDP messages that are transferred over a stream oriented TCP protocol.

2.3.1. NetEffect RNIC. NetEffect has recently introduced a 10-Gigabit iWARP RNIC [21] that works on the PCI-Express x8 I/O interconnect. The NIC core hardware is connected to a local 64/133MHz PCI-X bus that is bridged to the PCI-Express bus. The NetEffect RNIC consists of a Protocol Engine integrating iWARP, IPv4 TOE and NIC acceleration logic in hardware, a RAM based Transaction Switch operating on in-flight data, a local Memory Controller for buffering non-RDMA connections, and a 256MB on-board DDR2 memory bank. Memory registration, generating completions, and managing errors and exceptions are part of Protocol Engine's overall responsibility. The NetEffect RNIC can be accessed using user-level and kernel-level libraries such as NetEffect verbs, OpenFabrics verbs, standard sockets, SDP, uDAPL [10], and MPI.

3. Related Work

A large body of work exists concerning the performance analysis of modern interconnects including [4, 15, 5, 28]. In [4], the authors evaluate Cray 3E, IBM SP, Quadrics QsNet, Myrinet 2000 and Gigabit Ethernet. The work in [15] compares the performance of InfiniBand with Myrinet 2000 and Quadrics QsNet at the user-level. The researchers in [5] compare InfiniBand with Quadrics

$QsNet^{II}$. A comprehensive performance analysis of Myrinet two-port networks is presented in [28].

Some initial work has been published on iWARP using the Ammasso RNIC, an early iWARP enabled Gigabit Ethernet adapter released in 2004 [7, 24, 13, 9, 1, 8]. In [7], the paper compares the MPI latency and bandwidth with those of TCP and iWARP. Other work [24] shows that RDMA Ethernet significantly outperforms conventional Ethernet. In [13], the authors compare the iWARP verbs and TCP sockets latencies for wide area network applications.

In [9], iWARP has been implemented in kernel space for the client side. Although this does not provide any performance benefit and may even impose extra overhead on the clients, it helps to reduce server CPU utilization. In [1], an extended socket interface is implemented that helps legacy socket-based programs run without modification on the iWARP cards.

In [8], preliminary performance results are reported comparing the NetEffect NE010 RNIC on a 133MHz PCI-X bus with a Mellanox 4X IB card using an x8 PCI-Express interface. The results show that the NetEffect performs better in memory registration cost and CPU utilization for large messages, while lagging behind in latency.

4. Experimental Framework

We have conducted our experiments using four Dell PowerEdge 2850 servers. Each machine is a dual-processor Intel Xeon 2.8GHz SMP with 1MB L2-cache per processor and 2GB total physical memory. Each node has an x8 as well as an x4 PCI-Express slot.

For the iWARP tests, we used four single-port NetEffect NE010e 10-Gigabit Ethernet channel adapters [21], each with a PCI-Express x8 interface and CX-4 board connectivity. A Fujitsu XG700-CX4 12-port 10-Gigabit Ethernet switch was used. For the Myrinet network, we used four single-port Myri-10G NICs (10G-PCIE-8A-C) with 10GBase-CX4 ports [20], each with a PCI-Express x8 interface. Myrinet cards were forced to work in the PCI express x4 mode for effective performance on the nodes' Intel E7520 chipset. We connected the cards to a Myricom Myri-10G 16-port switch for the MXoM, and to a Fujitsu XG700-CX4 12-port 10-Gigabit Ethernet switch for the MXoE. Our InfiniBand network consists of four dual-port 10Gb/s Mellanox MHEA28-XT (MemFree) HCA cards [16], each with a PCI-Express x8 interface, connected to a Mellanox 12-port 4X MTS2400-12T4 InfiniBand switch. Note that in each experiment, only one of the network cards is installed to the x8 PCI-Express bus.

The machines run Linux Fedora Core 4 SMP for IA32 architectures with kernel version 2.6.11, unless otherwise noted. The NetEffect iWARP MPI is based on MPICH2 version 1.0.3 [19]. For Myrinet, we used MPICH-MX based on MPICH 1.2.7..1. For all of the Myrinet tests, we have enabled the MX registration cache. The MVAPICH2 over VAPI, version 0.9.5 [18], based on MPICH2 1.0.3, is used for the InfiniBand network.

5. User-level Performance

In this section, we compare the latency and bandwidth of four user-level communication libraries: NetEffect iWARP verbs 1.4.3, Mellanox VAPI 4.1.1, MX-10G over Ethernet (MXoE) preliminary version 1.2.1 and MX-10G over Myrinet (MXoM) version 1.2.0.

Unlike the MX-10G library, where communication is based on non-blocking Send/Receive operations, RDMA operations form the main communication model in the IB and iWARP protocol stacks (the Send/Receive model is also supported). For RDMA operations, the remote memory address tag needs to be exchanged prior to any data transfer. The communication model is one-sided, therefore an explicit synchronization is required at the end of communication to make sure the data is transferred completely. This kind of synchronization can be done using a Send/Receive operation. However, to measure optimistic results, we check completion of the RDMA write operations by polling the target buffer.

Figure 1 shows the inter-node bidirectional latency and bandwidth of the four user-level libraries. For MX-10G, we use MX *isend* and *irecv* primitives and test the completion using the MX *test* operation. For iWARP and IB, each side sends a message using an RDMA Write and waits for a reply from the other side. These tests are repeated a sufficient number of times to eliminate the transient conditions of the networks. The average round-trip time divided by two is reported as the ping-pong latency. Bandwidth is computed using the latency results.

The RDMA Write short message latencies for iWARP and InfiniBand are 9.78µs and 4.53µs, respectively. The latencies for MXoM and MXoE send/receive are 3.15µs and 3.45µs, respectively. This represents the best latency of all of the interconnects. The short message latency of iWARP is larger than cluster-specific interconnects. Nevertheless, these results show a dramatic improvement in Ethernet latency. The one-way bandwidth for iWARP is 880MB/s, which represents 83% of the maximum available bandwidth. On the other hand, the IB verbs saturate 97% of the one-way IB bandwidth, roughly 970MB/s, while the bandwidth of Myrinet does not exceed 75% of the available bandwidth. The dip in the Myrinet MX bandwidth curves is due to the *Eager/Rendezvous* protocol switching point in the MX library implementation.

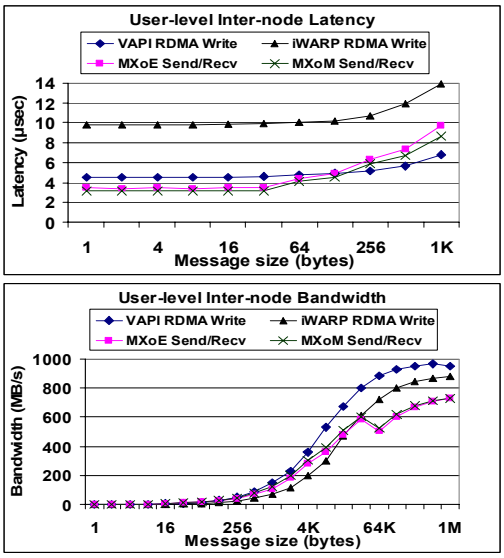

Figure 1. User-level ping-pong performance.

5.1. Multiple Connection Scalability

High-performance cluster computing nodes are being replaced with emerging multi-core multiprocessor servers. In such clusters, each core will run at least one process with connections to several other processes. Therefore, it is extremely important for the NIC hardware and its communication software to provide scalable performance with the increasing number of connections.

Both the iWARP and InfiniBand standards have a QP based communication model with similar semantics and they both support connection-oriented RDMA operations. Therefore, it is worth comparing the hardware implementation of these two standards in terms of multi-connection scalability. To have a head-to-head and fair architectural comparison between the NetEffect iWARP and Mellanox InfiniBand NICs in handling multiple connections, we used the OpenFabrics/Gen2 verbs as a common user-level interface, running over Fedora Core 5 SMP for x86-64 architecture with kernel version 2.6.17.7.

For each network, we pre-establish a number of connections (up to 256 connections) between two processes running on two nodes (each node has only one NIC) and then start communicating over those connections. To determine latencies, we performed a ping-pong test using all of the connections in parallel. A fixed number of messages are sent and received over the connections in a round-robin fashion. We vary the number of connections and message sizes and report the cumulative half round trip time divided by the number of connections and messages as the *normalized multiple-connection latency*. This shows how well communications over multiple connections can be performed in parallel.

As shown in Figure 2, by increasing the number of connections, the normalized multiple-connection latency for the NetEffect's iWARP implementation decreases even for up to 128 connections. This reflects the ability of the card to keep up with a large number of parallel connections. A relatively fixed latency for larger number of connections shows a serialization of communications over multiple connections for messages smaller than 1KB.

For the IB card, the normalized multiple-connection latency of small messages (up to 4KB) decreases with the increasing number of connections, but only up to eight connections. After that, the latency increases but stays relatively constant, showing the serialization impact on the communication. For the tested numbers of connections, normalized multiple-connection latency increases at some point for the IB card (this is not the case for the iWARP). We speculate that the processor-based communication in IB NIC core hardware is the main reason behind the serialization (all other components are the same for both networks). On the other hand, the iWARP RNIC has a pipelined architecture, which parallelizes multi-connection communication. The behavior of both networks is very similar for messages larger than 4KB.

For the throughput test, a both-way communication is performed, where each process sends messages to its peer in a round-robin fashion over the established connections. The test lasts for a certain amount of time and at the end, the throughput is reported as the ratio of the amount of data transferred over the communication time. Figure 2 shows that for both networks, the results are in harmony with the normalized multiple-connection latency results. In the case of the IB card, the throughput drops at eight connections (for messages smaller than 4KB). On the other hand, iWARP sustains the throughput with any number of connections in our measurement range. The throughput behavior of networks is the same beyond 4KB messages.

6. MPI Performance

We evaluate the potential of the MPI implementation using several micro-benchmarks. We begin with the ping-pong latency and bandwidth tests, followed by a number of in-depth experiments including parameters of the LogP model, and the effect of MPI buffer re-use and queue usage on communication latency (space does not allow including the hotspot, overlap and independent progress results).

We bind the affinity of processes to processors in order to avoid the overhead of process migration on cache performance. We use the *MPI_Wtime* and consider its timing overhead in each measurement. All results are reported as the average over 1000 iterations.

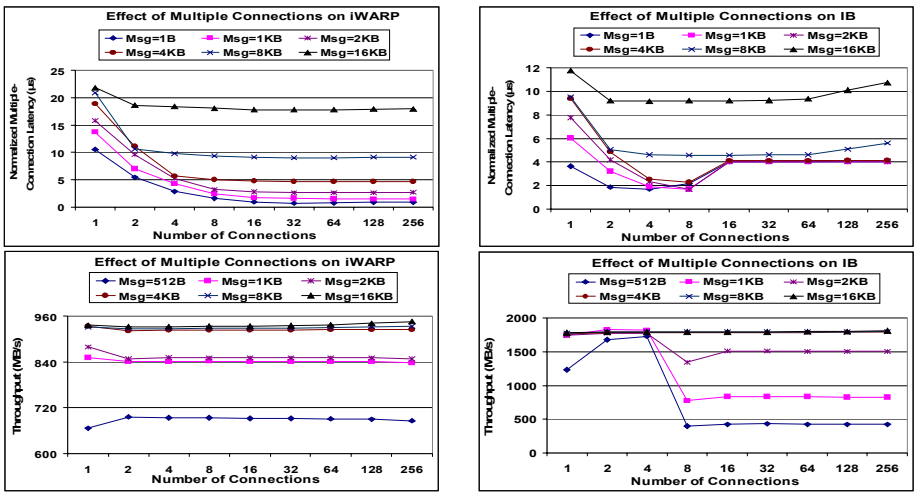

Figure 2. Throughput and normalized multiple-connection latency of NetEffect iWARP and Mellanox IB.

6.1. Latency

Figure 3 illustrates the standard inter-node MPI ping-pong latency of the interconnects. The short message latency is around 11.7µs for iWARP, 4.8µs for IB, 3.3µs for MXoM, and 3.6µs for MXoE. Figure 3 also shows the latency overhead of the MPI implementations over their respective user-level libraries. Results show that MPICH-MX offers the lowest overhead among the interconnects. This could be because the MX-10G is the only library with communication semantics close to that of MPI.

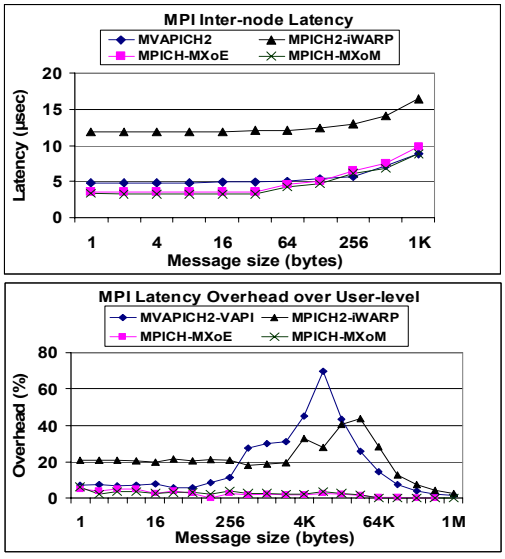

Figure 3. MPI ping-pong latency and its overhead over user-level.

6.2. Bandwidth

The inter-node bandwidth for unidirectional, bidirectional and both-way communication modes is presented in Figure 4. In the unidirectional test, the sender repeatedly transmits windows of non-blocking messages to the receiver, waits for each window to be completed and then for the last message to be acknowledged. The unidirectional bandwidth results show that all networks saturate the communication path for small messages. However, all networks incur a dip in bandwidth for some larger message sizes. This is the Eager/Rendezvous protocol switching point, which in our system happens between 4KB and 8KB for MVAPICH2, at 128KB for MPICH2-iWARP, and after 32KB for the Myrinet. InfiniBand has a steeper slope at the switching point.

In the bidirectional test, a blocking ping-pong operation is performed. Up to 856MB/s for iWARP, 962MB/s for IB, and 734MB/s for Myrinet can be achieved for the bidirectional case. However, the networks are not well utilized for small to medium size messages when compared to the unidirectional and both-way tests.

In the both-way test, both the sender and receiver post a window of non-blocking send operations, followed by a window of non-blocking receive calls. This method puts more pressure on the communication and I/O subsystems. While a maximum bandwidth of 1064MB/s is achievable with the NetEffect card on its internal PCI-X bus, the effective both-way bandwidth for 1MB messages is 925MB/s, which is a high level of network utilization with only two communicating processes. Both-way results for IB are also very promising and show around 89% utilization of the 2GB/s of available IB bandwidth. Myrinet achieves some 70% of the 2GB/s bandwidth on our system. InfiniBand is the clear winner in the bandwidth tests.

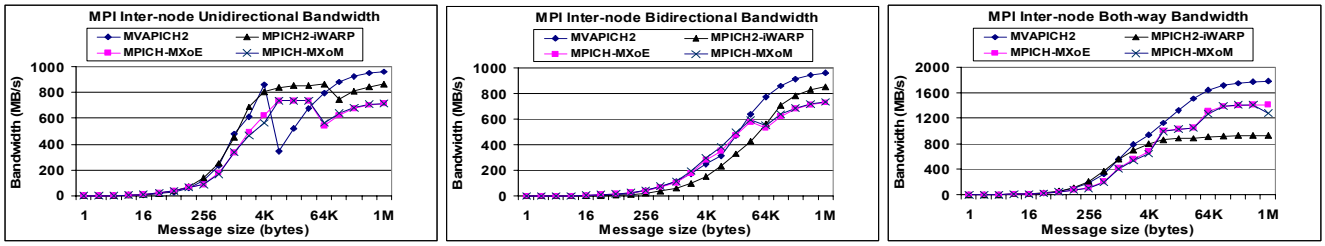

Figure 4. MPI bandwidth.

6.3. LogP Parameters

The *LogP* model has been proposed to gain insights into the different components of a communication step [14]. For a message of size *m*, $O_s(m)$ is the overhead in processing the message at the sender and $O_r(m)$ is processing overhead at receiver. The gap, *g(m)*, is the minimum time interval between two consecutive message transmissions. Figure 5 shows the results for *g(m)*, $O_s(m)$ and $O_r(m)$ for all four networks, in logarithmic scale.

The gap value for a 1-byte message is 2µs for iWARP and Myrinet, and 3µs for the IB network. The gap value grows steadily with the message size. The sender and receiver overheads for all of the networks are 1µs for very short messages. Such a small overhead for O_s and O_r, especially for short messages, is mostly due to excellent offloaded protocol processing at the NICs. The sender overhead for all networks remains low at the protocol switching point and afterwards. In general, InfiniBand has lower sender overhead for medium size messages.

For medium size messages, the NetEffect card clearly is the best with respect to the receiver overhead. However, a dramatic jump in the receiver overhead is observed at the Eager/Rendezvous switching points, except for Myrinet. Large receiver overhead for the iWARP and IB networks shows that mostly the receiving process is involved in the data transfer for messages using the Rendezvous protocol. On the other hand, Myrinet has a progression thread that is awakened for starting large message transfers.

6.4. Effect of Message Buffer Re-use

Applications using different message buffers may need to register/deregister their user buffers. Pinning/unpinning is expensive and an application's buffer re-use pattern may have a significant impact on communication performance. In some MPI implementations, a pin-down cache is used, where the buffer re-use policy affects its performance. Memory address translation overhead and TLB performance are other important factors.

To evaluate the impact of MPI buffer management on communication performance, we examine different buffer re-use patterns for the MPI ping-pong test [28]. In this test, for each message size we statically allocate 1024 separate memory buffers. Depending on the buffer-reuse pattern, we select a new buffer from the available buffers or use the previously used buffer. Figure 6 shows the ratio of ping-pong latency when changing the buffer re-use pattern from no re-use (0% re-use) to full re-use (100% re-use or always use a constant buffer) for each network.

For small messages up to 256B we see less than 10% impact on the networks. For Eager size messages, this ratio is less than 1.8 for iWARP, 1.55 for IB and 1.53 for Myrinet. The ratio grows for Rendezvous size messages and reaches 4.3 for IB at 8KB, 2.1 for iWARP at 256KB, and around 2.4 for Myri-10G network at 1MB. This will have a large effect on the Rendezvous protocol's performance. Looking at the MPI source code shows that the MPI implementations for iWARP and IB networks require costly memory registrations [8] for the Rendezvous protocol at the sender and receiver sides when new message buffers are used. However, the impact on the IB is significant. For very large messages, iWARP performs the best. Note when we disable the Myrinet registration cache, the effect of buffer re-use decreases to a maximum of 1.8.

6.5. Effect of MPI Queue Usage

MPI implementations use several queues for processing communication calls. The *receive call queue* is used to keep early posted receive calls. It is traversed upon reception of a message from the network to find a matching receive call. The *unexpected message queue* is used to save unexpected messages temporarily (messages that no matching receive call has yet been posted for). When a receive call is invoked, the unexpected message queue is traversed for a matching message. In this section, we examine the effect of queue usage for these two queues.

6.5.1. Unexpected Message Queue. We use an algorithm similar to what is proposed in [27]. Each process sends a certain number of small, unexpected messages to the other side. Then the processes synchronize and start communicating in a ping-pong fashion. We changed the algorithm in [27] and used synchronous send calls instead of non-blocking calls to avoid any overlapping of queue processing with message communication time, in order to measure the worst-case latency effects.

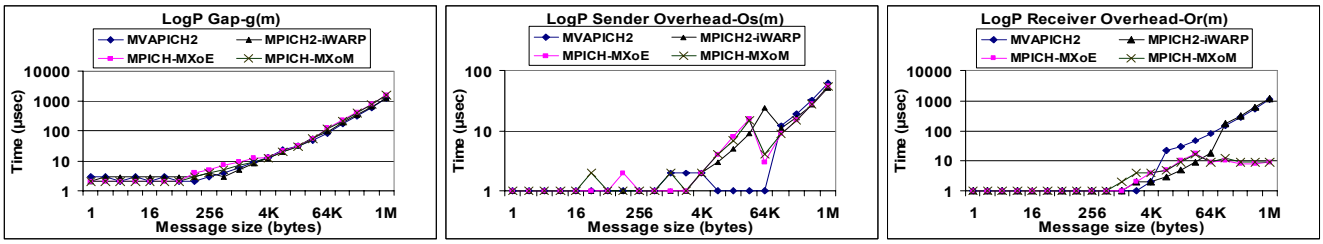

Figure 5. Parameterized LogP parameters.

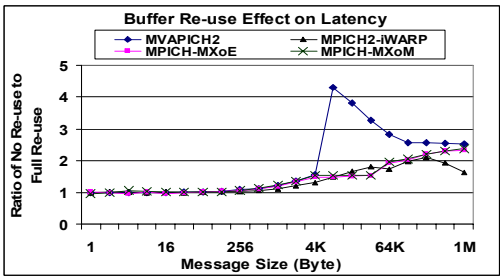

Figure 6. Effect of buffer re-use.

In Figure 7, we show the ratio of message latency when there are 1000 unexpected messages over the message latency when the queue is empty. As shown, small and medium size messages are considerably affected when 1000 unexpected messages exist in the queue. However, the impact on large messages is insignificant, especially for the iWARP case. MPICH-MX for both Myrinet and Ethernet offers the best performance. This is because Myrinet offloads unexpected message handling to the NIC.

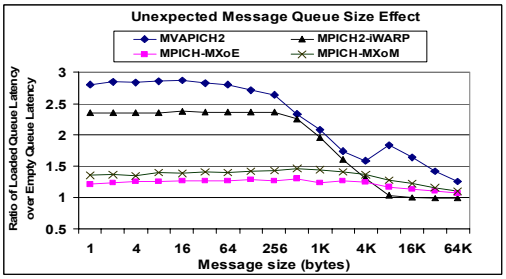

Figure 7. Effect of unexpected messages.

6.5.2. Receive Call Queue. For the receive call queue, we use an algorithm similar to the one in [27]. Initially, both sides of the communication pre-post a certain number of non-blocking receive calls with a certain tag (*tag1*), which are called "traversed calls". These calls sit at the beginning of the receive call queue. Then the actual latency-measured non-blocking receive call is posted with a different tag (*tag2*). This call sits in the queue after the traversed calls. At this stage, both sides synchronize and start communicating. One side sends a message with *tag2* and waits for a similar message in response. Upon reception of a message at either side, the queue is traversed to find the matching receive with *tag2*. Before finding the matching call, all pre-posted calls with *tag1* are traversed (but not processed). The latency is measured against different number of traversed messages.

In Figure 8, we show the ratio of latency when there are 1000 traversed messages to the basic message latency when the queue is empty. Obviously the receive queue impact on MVAPICH2 performance is more than twice that of MPICH2-iWARP for small messages. The best implementation in this case is the MPICH2-iWARP with a maximum ratio of 2.5. Myrinet is the worst network for this test. Myrinet uses the same technique as in the unexpected message processing for the early received calls and this approach apparently does not perform well.

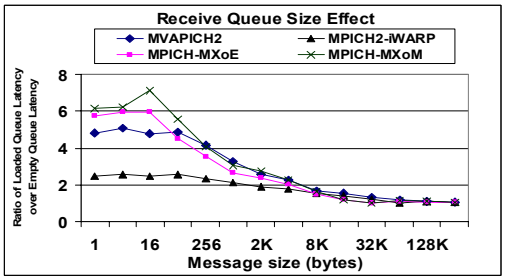

Figure 8. Effect of receive queue usage.

7. Conclusions and Future Work

Ethernet is the most common interconnect used in local area networks and clusters. TCP offload Engines and RDMA help to achieve low host CPU utilization in Ethernet networks. The iWARP protocol has been recently proposed to take advantage of RDMA over Ethernet.

We have compared the NetEffect iWARP with Mellanox InfiniBand and Myricom Myri-10G at the user-level and MPI layer. Although Myrinet is the winner in the latency tests, and InfiniBand is the best in the bandwidth tests, results show that the NetEffect RNIC achieves an unprecedented (TCP) latency for Ethernet, and is able to saturate 87% of the internal PCI-X available bandwidth. The hardware parallelism of the NetEffect device demonstrated better scalability for multi-connection communication when compared to the Mellanox IB card. The iWARP MPI implementation also performs better than MVAPICH2 in MPI queue usage and the buffer re-use.

Overall, the results show iWARP could be a key player in high-performance computing as technology matures.

We plan to put these networks to the test in a larger testbed to have a better evaluation of the extent to which the multiple-connection performance of the NetEffect device will affect real world applications. We intend to extend our study to include uDAPL, sockets, and applications. We would also like to enhance the NetEffect MPI implementation.

8. Acknowledgments

We would like to thank NetEffect for the resources to conduct the iWARP tests. Special thanks go to Vadim Makhervaks, Brian Hausauer and Terry Hulett of NetEffect for providing technical details about the NetEffect iWARP implementation. We are indebted to Myricom for providing the Myrinet switch and technical support. We are grateful to the anonymous referees for their insightful comments. This research is also supported by grants from the Natural Sciences and Engineering Research Council of Canada, Canada Foundation for Innovation, Ontario Innovation Trust, and Queen's University.

References

[1] P. Balaji, H. Jin, K. Vaidyanathan and D.K. Panda. Supporting iWARP compatibility and features for regular network adapters. In *2nd IEEE Workshop on Remote Direct Memory Access (RDMA): Applications, Implementations, and Technologies (RAIT 2005)*, 2005.

[2] P. Balaji, W.-C. Feng, and D.K. Panda. Bridging the Ethernet-Ethernot performance Gap. *IEEE Micro,* 26(3):24-40, 2006.

[3] J. Beecroft, D. Addison, D. Hewson, M. McLaren, D. Roweth, F. Petrini, and J. Nieplocha. QsNetII: Defining high-performance network design. *IEEE Micro*, 25(4):34-47, 2005.

[4] 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 *17th IEEE/ACM International Parallel and Distributed Processing Symposium (IPDPS'03),* 2003.

[5] R. Brightwell, D. Doerfler and K.D. Underwood. A comparison of 4X InfiniBand and quadrics elan-4 technologies. In *2004 IEEE International Conference on Cluster Computing (Cluster 2004),* pages 193-204, 2004.

[6] P. Culley, U. Elzur, R. Recio and S. Bailey. Marker PDU aligned framing for TCP specification (v1.0), 2002. http://www.rdmaconsortium.org/.

[7] D. Dalessandro and P. Wyckoff. A performance analysis of the Ammasso RDMA enabled Ethernet adapter and its iWARP API. In *2nd IEEE Workshop on Remote Direct Memory Access (RDMA): Applications, Implementations, and Technologies (RAIT 2005)*, 2005.

[8] D. Dalessandro, P. Wyckoff and G. Montry. Initial performance evaluation of the NetEffect 10 Gigabit iWARP adapter. In *3rd IEEE Workshop on Remote Direct Memory Access (RDMA): Applications, Implementations, and Technologies (RAIT 2006)*, 2006.

[9] D. Dalessandro, A. Devulapalli and P. Wyckoff. iWARP protocol kernel space software implementation. In *6th IEEE Workshop on Communication Architecture for Clusters,* 2006.

[10] DAT Collaborative: http://www.datcollaborative.org/.

[11] J. Hilland, P. Culley, J. Pinkerton and R. Recio. RDMA protocol verbs specification (v1.0), 2003. http://www.rdmaconsortium.org/.

[12] InfiniBand Architecture: http://www.infinibandta.org/.

[13] H. Jin, S. Narravula, G. Brown, K. Vaidyanathan, P. Balaji and D.K. Panda. Performance evaluation of RDMA over IP: A case study with the Ammasso Gigabit Ethernet NIC. In *Workshop on High Performance Interconnects for Distributed Computing (HPIDC'05),* 2005.

[14] T. Kielmann, H.E. Bal and K. Verstoep. Fast measurement of LogP parameters for message passing platforms. In *4th Workshop on Runtime Systems for Parallel Programming,* pages 1176-1183, 2000.

[15] 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. In *IEEE Micro,* 24(1):42-51, 2004.

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

[17] Message Passing Interface Forum: MPI, A Message Passing Interface standard, Version 1.2, 1997.

[18] MPI over InfiniBand Project, The Ohio State University: http://nowlab.cse.ohio-state.edu/projects/mpi-iba/.

[19] MPICH2: http://www-unix.mcs.anl.gov/mpi/mpich2/.

[20] Myricom: http://www.myricom.com/.

[21] NetEffect NE010e 10 Gb iWARP Ethernet Channel Adapter: http://www.neteffect.com/.

[22] Y. Qian and A. Afsahi. Efficient RDMA-based Multi-port Collectives on Multi-rail QsNet^{II} Clusters. In *6th IEEE Workshop on Communication Architecture for Clusters (CAC)*, 2006.

[23] RDMA Consortium. iWARP protocol specification. http://www.rdmaconsortium.org/.

[24] C.B. Reardon, A.D. George, and C.T. Cole. Comparative Performance Analysis of RDMA-Enhanced Ethernet. In *Workshop on High-Performance Interconnects for Distributed Computing, 2005.*

[25] H. Shah, J. Pinkerton, R. Recio and P. Culley. Direct data placement over reliable transports (v1.0), 2002. http://www.rdmaconsortium.org/.

[26] S. Sur, H. Jin, L. Chai and D. K. Panda. RDMA read based rendezvous protocol for MPI over InfiniBand: Design alternatives and benefits. In *11th ACM SIGPLAN Symposium on Principles and Practice of Parallel Programming (PPoPP '06),* pages 32-39, 2006.

[27] K.D. Underwood and R. Brightwell. The impact of MPI queue usage on message latency. In *2004 International Conference on Parallel Processing,* pages 152-160, 2004.

[28] R. Zamani, Y. Qian and A. Afsahi. An evaluation of the Myrinet/GM2 two-port networks. In *2004 IEEE Workshop on High Speed Local Area Networks (HSLN '04)*, pages 734-742, 2004.