Enabling High Performance Data Transfer on Cluster Architecture.

Paul A. Farrell and Hong Ong
Department of Mathematics and Computer Science
Kent State University, Kent, OH 44224, USA
{farrell, hong}@mcs.kent.edu

Stephen Scott
Oak Ridge National Laboratory
Oak Ridge, TN 37831-6367, USA
scottsl@ornl.gov

Abstract

In this article, we present some results on the communication performance of TCP/IP in a cluster of Pentium based workstations connected by a Gigabit Ethernet network. The experiments were performed in order to identify the bottlenecks in the communication protocol stack. We outline our performance measurements and suggest means to improve the overall performance of TCP/IP. Finally, we present the performance of VIA as an alternative for high-speed communication.

1. Introduction

Due to the overhead incurred in processing the legacy protocol stacks at the operating system layer, only a fraction of the raw performance of high speed networks is attained by applications. Recently, the Virtual Interface Architecture (VIA) [1] has been developed to standardize the efforts to address this problem and to make these ideas available to commercial systems. VIA defines mechanisms that will bypass layers of protocol stacks and avoid immediate copies of data during sending and receiving of messages.

2. Testing Environment

The testing environment for collecting the performance results consists of dual processor Pentium III PCs running at 450MHz with 256MB 100MHz (PC100) SD-RAM. The PCs are connected together through a SVEC 5 port 10/100 Mbps auto-sensing/auto-switching hub via DEC *Tulip* NICs and through a Foundry FastIron IIGC 1000Mbps switch via SysKonnect *SK-9821* and *SK9843* NICs. In addition, two pairs of these PCs have Alteon *ACEnic* and Packet Engine *GNIC-II* NICs installed. All the NICs are installed in the 33MHz PCI slot. The PCs run the Red-Hat 6.1 Linux distribution with SMP kernel 2.2.14. The device drivers used to perform the experiments are Tulip v0.91 for DEC *Tulip* NICs, Sk98lin v3.04 for SysKonnect *SK-9821* and SK-*9843* NICs, Acenic v0.41 for Alteon *ACEnic* NICs, and Hamachi v0.11 for Packet Engine *GNIC-II* NICs. In accordance with [2], the PCs have been configured to support the TCP/IP extension for high performance (RFC1323), Path MTU discovery (RFC1191), and TCP Selective Acknowledge (RFC2018). In addition, the TCP window size has been increased from the default 64KB to 1MB. M-VIA [3], a Linux based VIA implementation, has been installed to investigate the VIA communication performance.

3. Performance Results

In this section, we summarize the performance characteristics of TCP/IP and VIA. For a detailed description of the performance results and graphs, please refer to http://dune.mcs.kent.edu/~farrell/distcomp/perf/.

3.1 TCP/IP Performance Characteristics

TCP/IP latency and bandwidth figures for the different NICs were obtained using the NetPIPE benchmark [4]. Here, we will present the TCP/IP performance results obtained using socket buffer size equal to 128KB and MTU equal to 1500 and 9000 bytes.

With MTU of 1500 bytes, the maximum TCP/IP throughput obtained from the *Tulip* is approximately 86Mbps and its latency is approximately 105μsec. The *SK-9821* and *SK-9843* achieved maximum TCP/IP throughput of 279 Mbps and 275 Mbps, respectively. The latency for both adapters is approximately 96μsec. As compared to the predecessor of Sklin98 running on Linux kernel 2.2.12, the *SK-9843*'s throughput has decreased by 17% and its latency has increased by a factor of 2.15. Moreover, there are many severe dropouts for message sizes greater than 1MB, which we did not see in the previously collected data. For *ACEnic*, the maximum TCP/IP throughput is approximately 239Mbps with latency of approximately 498μsec. By setting the socket buffer size to 256KB, we were able to increase the throughput by 13% without affecting the latency. The maximum throughput for the *GNIC-II* is approximately 267Mbps with latency of approximately 154μsec. There is a drop of throughput for message size ranges from 3KB to 4KB. This anomaly has been noted in kernel 2.x and

observed by other research groups as well. We discovered that this anomaly is related to the transmission and receipt interrupts in the *GNIC-II* device driver. By tuning these parameters, we were able to smooth the curve and substantially increase the TCP/IP throughput by approximately 28% for message sizes under 1MB and decrease the latency to approximately 47μsec.

With MTU of 9000 bytes, the maximum attainable TCP/IP throughput increases to approximately 621Mbps for *SK-9821* and *SK-9843*, and the latency is approximately 93μsec and 97μsec for the *SK-9821* and *SK-9843* respectively. For the *ACEnic*, increasing MTU size and socket buffer size does not improve TCP/IP performance. We obtained a maximum throughput of 253Mbps with latency of 499μsec. In fact, for Alteon, the peak performance of 306Mbps can be obtained with MTU equal to 6500 bytes. In addition, there were many severe dropouts for the *ACEnic* performance curves. This behavior has been noted in our previous results using kernel 2.2.12 as well. In general, the TCP/IP performance of *ACEnic* has not been improved since kernel 2.2.1 and Acenic v0.28 where we achieved maximum TCP/IP throughput of 470Mbps using MTU of 9000 bytes.

To understand and interpret these performance results, we attempt to identify the overhead incurred at each of the TCP/IP protocol layers. At the data link layer, the Gigabit NICs under test are capable of transferring up to approximately 1.25Gbps in each direction. This implies that in order to obtain this performance the I/O bus of the host should deliver data at the rate of 2.5Gbps. Since the Gigabit NICs are installed in the 33MHz PCI bus, the maximum transfer rate is roughly 1.056Gbps (32bit word at 33 MHz) which is 2.3 times slower that the maximum transfer rate of the Gigabit NICs. In addition, the frequency of interruption from the kernel and the method of allocating memory buffers of the device driver also affect the overhead cost incurred at this layer.

At the network layer, IP performs the route searching and fragmentation/defragmentation of packets according to the MTU. With our current system configuration, there is no need for routing. However, the fragmentation and defragmentation (MTU=1500) caused a significant overhead. We have shown that the performance increases when the MTU increases to 9000 bytes.

At the transport layer, the TCP provides, among other things, reliable data flow, multiplexing for the application layer, and synchronization. For our current setting, the cost of maintaining a single channel of communication is not significant. However, the cost of providing reliability is significant. This is because the operations involved require at least two data touching operations, a memory copy that saves the data for possible retransmission, and checksum calculation. Although we have taken care to enlarge the socket buffers to the "bandwidth * delay" product, increase the path MTU, and uses SACK support, the overall performance is still not close to realizing the underlying gigabit bandwidth.

3.2 MVIA Performance Characteristics

In the case of M-VIA on the *SK-9821* and *GNIC-II*, the test results are collected using the "vnettest" which comes with the distribution. "vnettest" is essentially a ping-pong like network benchmark program. For message sizes around 30KB, the throughput of the GNIC-II reaches approximately 455Mbps with latency of only 16μsec and the throughput of SK-9821 reaches approximately 450Mbps with latency of 25μsec. The maximum attainable throughput for MVIA remains yet to be determined. This is because the "vnettest" stops when message size reaches 32KB, which is the maximum data buffer size supported by the MVIA implementation. From this, it is obvious that streamlining and simplification of the higher-level communication layer provides higher throughput and latency.

4. Acknowledgements

Research sponsored by the Laboratory Directed Research and Development Program of Oak Ridge National Laboratory (ORNL), managed by UT-Battelle, LLC for the U. S. Department of Energy under Contract No. DE-AC05-00OR22725.

This work was supported in part by NSF CDA 9617541, NSF ASC 9720221, and through the Ohio Board of Regents Computer Science Enhancement Initiative.

References

[1] Compaq Computer Corp., Intel Corporation, Microsoft Corporation, Virtual Interface Architecture Specification version 1.0. http://www.viarch.org

[2] Paul Farrell and Hong Ong, Communication Performance over a Gigabit Ethernet Network, IEEE Proceedings of 19th IPCCC. 2000

[3] M-VIA: A High Performance Modular VIA for Linux. http://www.nersc.gov/research/FTG/via/

[4] Q. O. Snell, A. R. Mikler, and J. L Gustafson, NetPIPE: Network Protocol Independent Performance Evaluator, Ames Laboratory/Scalable Computing Lab, Iowa State. 1997