Optimizing TCP/IP Communication for Networked Machines as a Parallel System

OP Gupta[#], Karanjeet Singh Kahlon[##], Rakesh Jindal[#]

[#]Faculty of Computer Science, Punjab Agricultural University,Ludhiana, 141004 India
[##]Department of Computer Science, Guru Nanak Dev University, Amritsar, India

Summary

A Parallel system is collection of tightly coupled processors typically of the same type. In the present study, loosely coupled personal computers in a workgroup over the Intranet are going to be used. Though, networked machines are having different types of processors with varying clock speed yet computing on the networked machines are becoming very popular to solve both data intensive and compute intensive scientific problems due to the demand for higher performance and lower cost. Usually computational intensive areas have been referred to as scientific processing viz. linear algebra, information retrieval etc. One of the driving forces behind this shift is the availability of portable robust software to utilize and manage a network of PC's. As more and more organizations have high-speed local area networks interconnecting many general-purpose desktop PC's, the combined computational resources may exceed the power of a single high performance parallel computer. Such a design of distributed memory architecture using message passing interface between cooperative tasks within a parallel application is also called parallel system with loosely coupled processors. Since different tools and techniques are available to utilize and manage a network of PCs. It is believed that an efficient and affordable model of distributed parallel system based on the networked machines can be created with the use of these portable robust softwares and standardization of distributed parallel computing on network of PCs can be done. In this paper, we studied the behavior of parametric TCP/IP and optimize the communication among the networked machines keeping in view latency and communication load on bandwidth.

Key words:
Distributed Computing, Parallel Computing, Grid Computing, Local Area Network

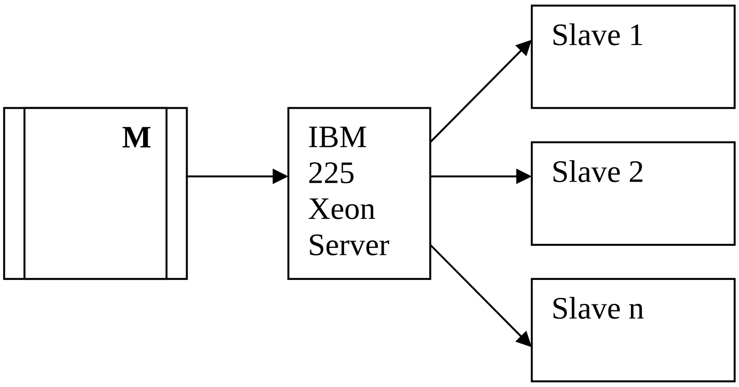
1. Introduction

Today, Distributed and high performance application requires high computation power and high communication performance. Of the top ten super computers, seven are COTS clusters: and the price of the third place COTS cluster is only 1.3 % of the cost of the first place parallel machine. So COTS has provided a cost-effective solution for high performance computing. Unfortunately, most software applications are not yet ready to harness the power of cluster computing. The majority of the application are based on shared memory architecture and parallelized for multithreaded SMP systems, while only a few applications have been created to exploit distributed memory using message passing architecture. Keeping in view of high cost of parallel machine (Rs. in Crores and Crores), Computing on Networked machines can be boon for the developing countries like India. This project can prove to be a stepping stone for the affordable supercomputing in universities and research organization. By using existing hardware, the cost of this computing will be very low. The latest processors can be easily upgraded to compete for future performance as compared to discarding the whole parallel machine. The virtual computer resources can

grow in stages and take advantage of the latest computational and network technologies. All these factors translate into reduced development and debugging time, reduced contention for resources, reduced costs, and possibly more effective implementations of an application. The purposed model will be suitable for those organizations that don't have any parallel machine but have a large no of computers on the network and are idle after office hours. Even when they are being used, most of them are doing word-processing, Web browsing, etc. and less than 5 percent of the CPU power is used.
It is believed that purposed work will not only enable small and medium size Research organizations and Universities to use networked computers as Virtual parallel computer to solve computational intensive tasks but also likely to contribute towards the standardization of Parallel computing on networked PCs. This will further refine the ways of development of parallel algorithm on networked PCs.

2. Network Design and Methodology

In this paper, study of the TCP/IP protocol is done and optimal value of parameters, which are going to affect the latency and bandwidth, will be purposed. A 3-tier network design is used for the study, which is shown in the figure 1.



The network design consists of three components.
Master PC (Job Submitter)
Server (Job Synchronizer and Partitioner)
Cluster of nodes (Task Executors)

Master PC is responsible for defining the job and submitting the job to Server. It can be anywhere in the Intranet and connected to server in the peer-to-peer network fashion at the time of submission of the job and getting back the results from the server.
Server breaks the job and allocates tasks to client nodes and sends back the final result to Master PC to display the results. Server can be centrally located so-that average distance of the all the nodes to server may be equally poised. The main advantage of using IBM server is to partition the job according to the computing power of the available nodes so that load balance may be maintained and server should not come under the situation of starvation.
Cluster of nodes or Task Executors, actually execute the code of the task and submit the results back to server. If the cluster consists of a set of identical workstations / PCs, the system is homogeneous. Further, a cluster can be divided into two classes: A dedicated and Non dedicated system. In the Non-dedicated system, each PC executes its normal work such as word processing or Internet browsing and only idle CPU cycles are used to execute parallel tasks (90% cycles are unused for 90% of the time as per study of MicroSoft SIGMetrics, 1998-2000).

In network evaluation, latency and asymptotic bandwidth are the two parameters, which effect the performance of both data intensive and compute intensive applications. In this study, certain parametric values of TCP/IP are suggested which improves the latency and bandwidth. "Ping" and "Inetperf" software written in "C" language is used to check one way communication and two-way communication in a network of PCs.
The TCP window size, a TCP / IP parameter, is the amount of buffering allowed by the receiver before an acknowledgement is required. Data is sent by TCP in segments that are typically 1460 bytes in length. In order to improve throughput, the sender must transmit

multiple segments without waiting for the acknowledgement. The formula that governs the optimum window size is

TCP Windows Size > = Bandwidth (in Bytes) X Latency (RTT)

Effective bandwidth always varies and can be affected by high latency. Too much latency in too short a period of time can create a bottleneck and prevents data from “filling the pipe”, thus decreasing effective bandwidth. In parallel system based on the networked PCs, the time of sending and receiving the message should be analyzed based on the TCP/IP parameters defined above and suitability of Ethernet / Fast Ethernet for coarse grain applications will be tested. The parameters of TCP/IP affecting the throttling of the bandwidth were studied and design of the network based on the above configuration has been tested.

3. Results and Analysis

Test - I

Results obtained using default TCP window sizes are summarized in the Table 1 and Table 2. Table 1 represents the send time and pack time between Server and Client node through a switch.

Table 1 Server-Switch-Slave

Message Size (B)	Send Time (ms)	Pack Time (ms)
128	0.041	0.065
256	0.062	0.093
512	0.098	0.110
1024	0.152	0.159
4096	0.598	0.601

Table 2 Server-Master

Message Size (B)	Send Time (ms)	Pack Time (ms)
128	0.031	0.038
256	0.048	0.052
512	0.078	0.086
1024	0.140	0.141
4096	0.331	0.330

Table 2 contains the timing between Server and Master PC, where Master PC is directly connected to Server through the multi-port Ethernet card installed in the Server.

Analysis of Test I

Pack time in case of Table 1 is on higher side as compared to Table 2. This is due to the slower CPU used as Client node. The lower send time in Table 2 as compared to Table 1 is attributed due to the direct connection of the Master PC in the multi-port Ethernet card of IBM Server.

On analyzing the above graph, it reveals that latency varies a lot when message size is relative small and it became stable for large values of message size. So the message size that can be standardized to tune of 4kB.

Test II

In the 2nd test, the size of the message was kept constant i.e. 4kB for purposed design. We evaluated network performance with different TCP window sizes. The network load during the experiment was kept at zero level so that results with more accuracy and stability can be collected. The other TCP/IP (DWORD values) parameter TCP1323Opts was kept to be 3.

This parameter is going help in the cycle of shrinking and slowing expanding windows size and subsequently in the data intensive and compute intensive parallel applications. The amount of data that can be in transfer in the network, termed as “Bandwidth-Delay-Product” or BDP for short, is simply the product of the bottleneck link bandwidth and round trip time (RTT). Table 3 summarizes the BDP for different size of TCP window and congestion is determined after comparing the BDP and actual TCP window size.

Table 3. BDP Measurement

	TCP Window Size			
No. of Hops	16kB (Default)	32kB	64kB	Congestion if BDP >=TCPWinSize
	BDP (kB)	BDP (kB)	BDP (kB)	*means Congestion
0	8.4	8.2	8.2	No
1	16.3*	15.28	15.10	No – Size to be =32kB
2	20.8*	24.0	24.4	No –Size to be =32kB
3	24.7*	32.2*	36.6	No – Size to be = 64kB

Analysis of Test II

From the table 3, it is clear that TCP window size should be 32kB or 64kB for good throughput in the network of level of two and three. This can set statically by the network administrator or can be set dynamically in the parallel application.

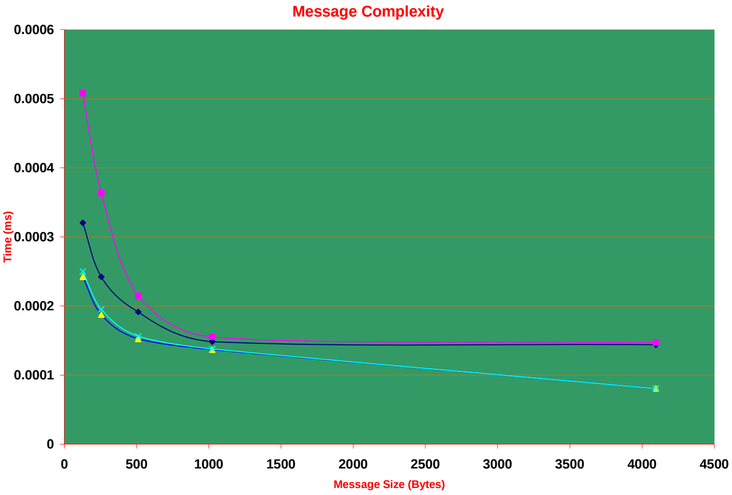


Figure 1 Message Complexity

4. Conclusions

In this paper, the design of message-passing communication subsystem is low-latency and scalable in nature. Our performance evaluation shows that it effectively delivers low latency for small messages and high bandwidth for large messages. This communication subsystem has been integrated into Deskgrid API based on Windows for parallel processing in local area network.

5. References

1. Anurag Acharya, Guy Edjlali, Joel Saltz (1997), “*The utility of exploiting idle workstations for parallel computation*” *ACM SIGMETRICS Performance Evaluation Review , Proceedings of the 1997 ACM SIGMETRICS international conference on Measurement and modeling of computer systems SIGMETRICS '97*, Volume 25 Issue 1
2. ACS - University of Kansas. Introduction to the Message Passing Interface (MPI) using C. *http://www.cc.ukans.edu/~acs/docs/other/intro-MPI-C.shtml*.
3. Becker, D., Sterling, T., Savarese, D., Dorband, J., Ranawak, U., Packer, C., "Beowulf: A Parallel Workstation for Scientific Computation, '~ Proceedings, International Conference on Parallel Processing, 1995. http://www.beowulf.org/.
4. Brent Wilson (2005) “ *Introduction to parallel programming using message-passing*” *Journal of Computing Sciences in Colleges*, Volume 21 Issue 1
5. Dave MacDonald and Warren Barkley, “Microsoft Windows 2000 TCP/IP Implementation Details”

OP Gupta received his M.E. from TIET, India in 1999. He is working as Asst. Professor in PAU, Ludhiana-India. His area of interest is parallel processing in Grid Environment. He is member of International Association of Engineers, Computer Society of India.