

Distributed Reliable Computing Systems based on COTS

D. Avresky
Dept. of El. and Comp. Eng.
Network Computing Research Lab
309 Dana research Building
Northeastern University
Boston, MA 02215
E-mail: {avresky}@ece.neu.edu

Abstract

The goal of this research is to identify and develop new key building blocks for reliable distributed systems built with inexpensive off-the shelf components. The platform consists of computing nodes connected via multiple interfaces to networks configured in fault-tolerant topologies. The software components of the distributed systems run in conjunction with operating systems services and standard network protocols, such as OS NT 4.0/2000, OS Linux, TCP/IP, Software Drivers for ServerNet1/2 and Myrinet, Message-Passing Interface (MPI/2) and Virtual-Interface Architecture (VIA). Features of the distributed system based on this components include high availability, dynamic reconfiguration and scalability. It can be used for improving availability, scalability and the performance of Internet data centers, distributed applications running concurrently across wide-area networks.

The distributed network simulation test bed (Distant), will be developed and used to validate the efficiency of SIFT approach and different routing and reconfiguration algorithms in arbitrary network topologies. Basis for Distant is a cluster of Windows NT workstations connected through gigabit Compaq ServerNet routers. All processes are organized in a MPI virtual topology, which corresponds to the physical topology of the simulated network. Distant can operate in the following modes: all processes are running on one node, partitioning the processes among the nodes and communicating via ServerNet or Ethernet (using TCP/IP). The proposed SIFT techniques will enable applications to be executed efficiently in distributed systems and network environments, and reliability and robustness proportional to the redundancy and spatial diversity available.

1 Distributed Reliable Computing Systems based on COTS

The popularity of the Internet is now making it possible to deliver computing services to millions, if the networks and software can meet the challenge. One of the major challenges in the area of Network Computing is creating reliable distributed systems that are based on commercial computing and communication technologies. The architecture in Fig. 1 represents a reliable heterogeneous distributed computing system based on inexpensive Components off-the-Shelf (COTS). Each of the nodes is a standard personal computer or workstation, using a standard operating system, such as Windows NT/2000 and Linux.

The nodes can be connected by commercial switches and adaptor cards to build different point-to-point networks. Specially ServerNet1/2 and Myrinet offer a simple, comprehensive solution to building such System Area Networks (SAN) that meet the demanding requirements of high availability and scalability. As well, the nodes can be connected by a high-speed Ethernet using the standard protocol TCP/IP.

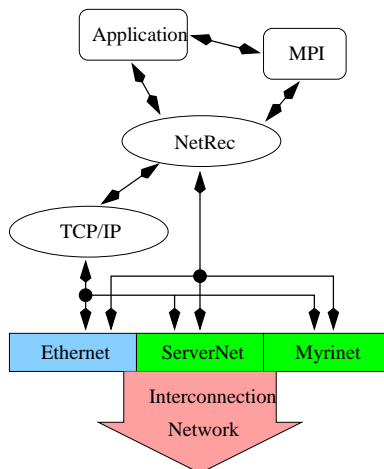


Figure 1: Distributed Reliable Computing System based on COTS

The main purpose of such distributed reliable computing system is to provide high availability and data integrity of distributed applications. Usually, the distributed application is represented as a set of user processes that communicate between themselves using a standard Application Programming Interface (API), like MPI, or by using directly the low level primitives provided by the reliable computation software. By using MPI and its functionality to create virtual topologies, any user application can be mapped on the distributed reliable computation system. Each application could be executed in several different modes, depending on the available resources and execution time requirements :

- single node - in this mode, all the processes are running on one node;
- Ethernet with TCP/IP - the application could be distributed over several nodes connected through standard Ethernet network using TCP/IP protocol;
- SAN (ServerNet1/2 and Myrinet) - SANs are high-speed low latency gigabit networks which can provide a platform for high-performance distributed computing.

The process of switching between these modes is transparent for the user and is controlled by simple configuration files.

The Virtual Interface Architecture (VIA), developed by Compaq for ServerNet will be used to improve the performance of distributed applications by reducing the latency associated with critical message-passing operations. The applications access hardware VIA mechanisms directly from user mode thus avoiding the costly kernel mode transitions that can reduce the efficiency of the program.

The execution time as a function of the compression ratio for a quad-tree data compression program executed in the topology shown in Fig. 2 is presented in Fig. 3 for the following operational modes : single node, Ethernet using TCP/IP and ServeNet1. As expected, execution times over ServerNet1 are the smallest based on the advanced features of the ServerNet.

A major component of the distributed reliable computing system is *NetRec*, as shown in Fig. 1. It implements all steps of the algorithm for a dynamic reconfiguration described in previous sections. It will be further developed during the research project for investigating problems related to the data integrity and high availability of the distributed application. We will study how the *NetRec* algorithm will reconfigure the network from multiple faults. Specially, when there is a node which has more then one restoration leader, i.e., several *faulty stars* have a common node.

We will analyze and develop different approaches in order to maximize the network's resistance to partitioning. There are several methods proposed in the literature [6, 16, 15] for solving this problem. For many distributed algorithms a large set of nodes have become partitioned

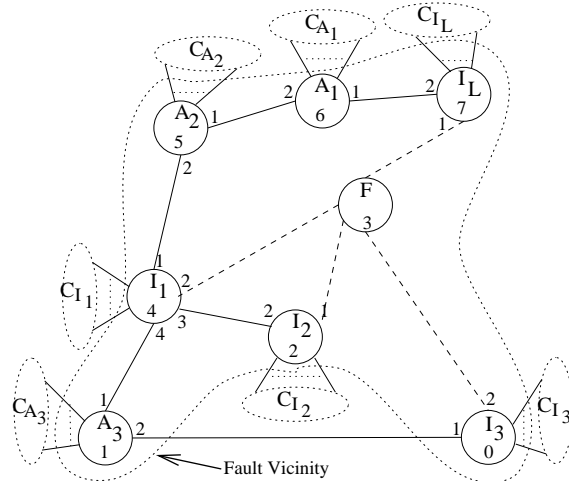


Figure 2: Reconfiguration Example

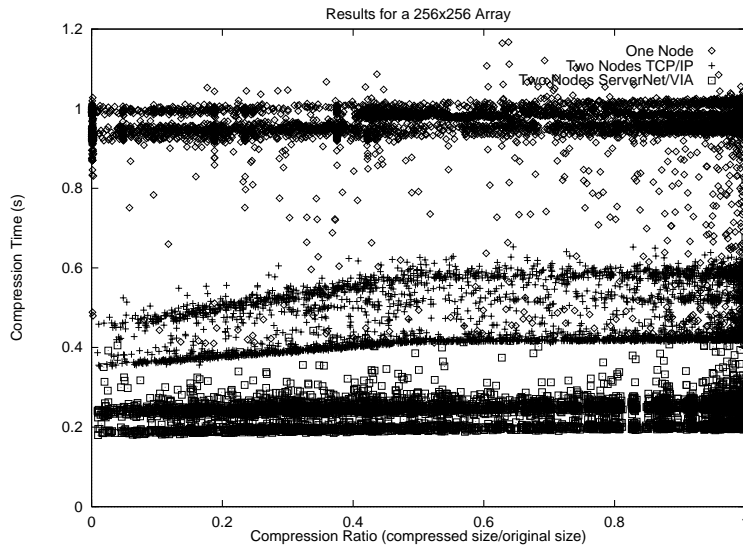


Figure 3: Quad-tree compression results

from others in the presence of faults. A network that is resistant to partitioning should lose only some constant numbers of nodes given that the number of failures do not exceed a certain value. Some fraction of the total number of compute nodes may be lost if additional failures occur. The problem will be how to create graphs of n computing nodes of a given degree connected to a clique network of n switches of a given degree that can tolerate k faults of any kind (switch, link, or node) without partitioning the network. Thus, only a constant number of nodes with respect to n will be lost.

Another challenging task is the implementation of a reliable group membership service in asynchronous distributed systems that ensures that the processes in a group maintain a consistent view of the global membership. There are a number of well-defined problems in these systems, such as consensus, group membership, commit, and atomic broadcast that have been studied in the literature [8, 14, 9, 17]. The problem is quite difficult, especially when a change in the membership occurs, either due to failures or to voluntary joins and withdrawals. Specially, the *NetRec* will be further developed to reconfigure dynamically the routing tables when the process (after the failure) will rejoin the topology and the traffic will be redirected through the

restored links. This will be done without suspending user traffic. In general, it is not possible to distinguish a slow process from a faulty process in an asynchronous environment [10, 7]. In this case, a live node can be excluded, if it is slow, from the topology. The *NetRec* algorithm will allow such a node to rejoin the topology. Special cases, such as "voluntary joins," will be studied. An active replication communication scheme and an adaptive majority voting scheme at the level of processes will be investigated in this environment, which will insure data integrity and high availability of the distributed applications.

SOFIT will be ported in the Distributed Reliable Computing System. It will be used to inject link, router, and node faults. Thus, the recovery overhead and the coverage factor c of the error-detection algorithm could be obtained in order to estimate the efficiency of the proposed SIFT for providing reliable computations in distributed and network environments.

We will investigate different approaches for mapping a given virtual topology onto a physical topology, based on ServerNet (Myrinet) COTS, or Ethernet environment by using TCP/IP. The goal will be to achieve a high reconfiguration rate δ and a small recomputation overhead in the presence of faults in the distributed systems in a way transparent to the user, which will provide high rMTTF. Thus, the research community and a large number of users will have the opportunity to build reliable distributed systems based on COTS in a user transparent way. This research project will have a positive impact on a large community implementing applications in distributed and network environments such as : e-commerce, telepresence, telemedicine, Internet data centers, distributed applications running concurrently across wide-area networks.

2 Distributed network simulation testbed - Distant

Distant is a distributed network simulation testbed, that is currently being developed to validate routing and reconfiguration algorithms in arbitrary network topologies. Different strategies for a dynamic process allocation in the presence of failures in order to ensure data integrity and accomplishing the task will be studied. *Distant* is implemented on a cluster of Windows NT 4.0 workstations connected through 1.28 gigabit/s Compaq ServerNet I routers. ServerNet [12, 11, 1, 2, 3, 4, 5] is a wormhole-based high performance System Area Network (SAN). It has been extensively studied and results can be found in [1, 2, 3, 4, 5, 13, 18, 19]. A cluster of Compaq workstations running MPI-1.2 is operational in the Network Computing Lab. In addition, the cluster nodes are linked through 10Mb Ethernet network running TCP/IP. In the cluster is installed MPI 1.2 software, which is used as a communication interface for the simulations.

Based on MPI, *Distant* provides the following standard features: simulation of network nodes, topology creation, shortest path routing, simulation of node failures, fault detection based on "I'm alive messages," and exchange of data messages between the nodes. Each network node is simulated by a separate MPI process which handles all the communication and routing activities. It is responsible for generating the "I'm alive" messages to its neighbors and detecting missing messages from them. It is also handling all message routing, as well as network reconfiguration in presence of failures. All processes are organized in a MPI virtual topology, which corresponds to the physical topology of the simulated network. Each process uses the MPI virtual topology functions to get the list of its neighbors and assign them to the communication ports of the node. In *Distant*, the MPI process rank is used as a node address.

2.1 Preliminary Results

The validation of *NetRec* has been performed by simulation, using the sample topology from Fig. 2 in three different modes:

- all processes running on a single node;
- processes equally split between two nodes with communication over ServerNet/VIA;
- processes equally split between two nodes with communication over Ethernet/TCP/IP.

Additionally, the validation of the algorithm was performed with different faults and different network addresses. In all cases the *NetRec* algorithm correctly reconfigured the routing tables of all FANs. In order to validate the *NetRec* complete traces were made of the message routing before and after network reconfiguration, based on generation of messages from random sources to random destinations. The routing tables of the FINs (0, 2) during the reconfiguration of the network are given in Table 1.

Table 1: Routing tables of the FINs before and after reconfiguration

Node 0				Node 2			
Before		After		Before		After	
Dest	Port	Dest	Port	Dest	Port	Dest	Port
1	1	1	1	0	1	0	2
2	2	2	1	1	2	1	2
3	2	3	-	3	1	3	-
4	1	4	1	4	2	4	2
5	1	5	1	5	2	5	2
6	2	6	1	6	1	6	2
7	2	7	1	7	1	7	2

References

- [1] D. Avresky, V. Shurbanov, and R. Horst. The effect of the router arbitration policy on the scalability of ServerNet™ topologies. *J. of Microprocessors and Microsystems*, 21:545–561, 1997. Elsevier Science, The Netherlands.
- [2] D. Avresky, V. Shurbanov, and R. Horst. Optimizing router arbitration in ServerNet™ topologies. In *Proc. of the 31st Annual Simulation Symposium*, pages 50–57, Boston, MA, April 1998.
- [3] D. Avresky, V. Shurbanov, R. Horst, W. Watson, L. Young, and D. Jewett. Maximum delivery time and hot spots in ServerNet™ topologies. In *Proc. of the IEEE Int. Par. Proc. Symp.*, pages 756–761, Geneva, Switzerland, April 1997.
- [4] D. Avresky, V. Shurbanov, R. Horst, W. Watson, L. Young, and D. Jewett. Performance modeling of ServerNet™ topologies. *The J. of Supercomputing*, 14(1), August 1999. Kluwer Acad. Pub.
- [5] D. Avresky, V. Shurbanov, R. Wilkinson, R. Horst, W. Watson, and L. Young. Maximum delivery time and hot spots in ServerNet™ topologies. *J. of Computer Networks and ISDN Systems*, 31(10), 1999. Elsevier Science, The Netherlands.
- [6] V. Bohossian, C. Fan, P. LeMahieu, M. Riedel, L. Xu and J. Bruck. Computing in the Rain: A Reliable Array of Independent Nodes. In *IEEE Trans. on Parallel and Distributed Systems, Special issue on Dependable Network computing*, eds. D. Avresky, D. Culler And J. Bruck, to appear, January, 2001.
- [7] T. Chandra, V. Hadzillacos, S. Toueg and B. Charron-Bost. On the Impossibility of Group Membership. *Proc. of the Fifteen ACM Symposium on Principals of Distributed Computing*, ACM Press, pp. 322-330, 1996.
- [8] M. Franceschetti and J. Bruck. On the Possibility Of Group Membership Protocol. *Dependable Network Computing*, editor D. Avresky, Kluwer Academic Publishers, pp. 77-92, 2000.
- [9] U. Fritzke, P. Ingels, A. Mostefaoui, M. Raynal. Consensus-Based Fault-Tolerant Total Order Multicast. In *IEEE Trans. on Parallel and Distributed Systems, Special issue on Dependable Network Computing*, eds. D. Avresky, D. Culler and J. Bruck, to appear, January, 2001.

- [10] M. Fisher, N. Lynch and M. Paterson. Impossibility of Distributed Consensus with One Faulty Process. *Journal of the ACM*, Vol.32, No. 2, pp.374-382, 1985.
- [11] D. Garcia and W. Watson. ServerNet-II. In *Lecture Notes in Computer Science*. In *Proc. of the Workshop on Parallel Computer Routing and Communications*, pp 119-136, Springer, June 1997.
- [12] R. Horst. TNet: A reliable system area network. *IEEE Micro*, pages 37–45, February 1995.
- [13] R. Horst, D. Avresky, R. Wilkinson, D. Jewett, W. Watson, L. Young, and C. Cunningham. Performance modeling of ServerNet™ topologies. In *Proc. of the IEEE Int. Par. Proc. Symp.*, pages 518–523, Honolulu, HW, USA., April 1996.
- [14] I. Keidar and D. Dolev. Totally ordered broadcast in the face of network partitions. *Dependable Network Computing*, editor D. Avresky, Kluwer Academic Publishers, pp. 51-75, 2000.
- [15] H. Ku, J.P.Hayes. Connective Fault Tolerance in Multiple-Bus Systems. *IEEE Trans. on Parallel and Distributed Systems*, Vol. 8, No. 6, pp.574-586, 1997.
- [16] P. LeMahieu, V. Bohossian and J. Bruck. Fault-Tolerant Switched Local Area Networks. In *Proceedings of the International Parallel Processing Symposium*, pp. 747-751, 1998.
- [17] D. Powell. Special issue on Group Communication. *Communications of the ACM*, Guest editor,39(4), April, 1996.
- [18] V. Shurbanov, D. Avresky, and R. Horst. The effect of the router arbitration policy on the scalability of ServerNet™ topologies. In *Proc. of IEEE 12th Int. Par. Proc. Symp.*, pages 604–609, Orlando, Florida, April 1998.
- [19] V. Shurbanov, D. Avresky, R. Horst, W. Watson, L. Young, and D. Jewett. A scalability study of ServerNet™ topologies. In *Proc. of ISCA 10th Int. Conf. on Par. & Dist. Comp. Sys.*, pages 558–563, New Orleans, Louisiana, USA, October 1997.