Journal of Theoretical and Applied Information Technology
15st August 2011. Vol. 30 No.1

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

RELIABILITY ANALYSIS OF RING, AGENT AND CLUSTER BASED DISTRIBUTED SYSTEMS

[1] R.SEETHALAKSHMI, [2]K.S.RAVICHANDRAN

[1]Professor,School of Computing, Sastra University, Tanjore, India – 613 401

[2]Associate Dean,School of Computing,Sastra University, Tanjore,India – 613 401

Email: rseetha123@cse.sastra.edu, raviks@it.sastra.edu

ABSTRACT

The introduction of pervasive devices and mobile devices has led to immense growth of real time distributed processing. In such context reliability of the computing environment is very important. Reliability is the probability that the devices, links, processes, programs and files work efficiently for the specified period of time and in the specified condition. Distributed systems are available as conventional ring networks, clusters and agent based systems. Reliability of such systems is focused. These networks are heterogeneous and scalable in nature. There are several factors, which are to be considered for reliability estimation. These include the application related factors like algorithms, data-set sizes, memory usage pattern, input-output, communication patterns, task granularity and load-balancing. It also includes the hardware related factors like processor architecture, memory hierarchy, input-output configuration and network. The software related factors concerning reliability are operating systems, compiler, communication protocols, libraries and preprocessor performance. In estimating the reliability of a system, the performance estimation is an important aspect. Reliability analysis is approached using probability.

Keywords: *Reliability, Distributed System, Clusters, Agents, Ring Networks Cluster Information Gathering.*

1. INTRODUCTION

The recent era has witnessed the growth of high-speed computers and communication technology, which has immensely contributed to the commercial availability of real time distributed systems. Distributed systems provide cost-effective ways for improving resource sharing, performance, throughput, fault tolerance and reliability. The redundant resources and cooperation among processing elements of distributed systems significantly affect the reliability performance and fault tolerance. Often a distributed system is formulated using a ring network. The characteristic of a ring network is simplicity and elegant topology. The ring network is also a broadcast network. In a ring network data is quickly transferred without a bottleneck. The ring network transmits the data at a very fast rate. Reliability analysis of such network is encountered. Another important distributed computing paradigm is the agent based system. Here multiple agents interact cooperatively and achieve a specified task. Reliability analysis of agent based system is dealt with. Increasing demand for computing power in scientific and engineering applications has spurred deployment of high performance computing systems that deliver tera-scale performance. Current and future high performance computing systems that are capable of running large-scale parallel applications may span hundreds of thousands of nodes. For parallel programs, the failure probability increases significantly with the increase in number of nodes. Ignoring failures or system reliability can have severe effect on the performance of the high performance computing cluster and quality of service. A reliability monitoring and analysis is performed in such a system. [2]

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

2. RELATED WORK

There exist several reliability analysis techniques which have flourished and are discussed. The first method for distributed system reliability approach is the reliability estimation using the survivability index method. The survivability index is a quantitative measure of the distributed communication system. It is defined as the expected number of programs for execution after the failure of some node or link has occurred. Survivability index method is too complex and hence it is not considered these days. The next approach is source-to-multiple-terminal reliability. The Source-to-Multiple-Terminal reliability is defined as the probability that a specified processing node can reach every other processing element in the distributed computing system. It does not consider the redundant programs and replication and hence it is not considered. The next approach is the Minimal Task Spanning Trees. A minimal task spanning tree is minimal when there is no other minimal task spanning tree of smaller size. The probability of executing a given distributed task successfully is evaluated as the probability that, there exists at least one minimal task spanning tree in the operational state in spite of channel failure, node failure and process failure. For larger networks this approach is very expensive and tedious. There are several other existing techniques, which are used for reliability analysis. It includes file Spanning Trees approach (Huang et al., 1990), Component Dependency Graph based approach(Yacoub et al., 1996) ,Graph Partitioning approach, Symbolic Method and Factoring Method. These methods also pose their own problems like high complexity and tediousness and hence they are not used these days. Hence in this paper reliability analysis is carried out in a variety of distributed systems using a probabilistic approach.

3. RELIABILITY ANALYSIS OF RING NETWORKS

Unidirectional ring networks were proposed as an alternative to the bus structure in the local area networks. Ring network is highly sophisticated and works efficiently when processing is distributed across various sites. There are several types of ring networks namely unidirectional ring network, bidirectional ring network, dual counter rotating ring network and chordal ring network. The unidirectional ring network is the conventional network and data movement is unidirectional. The processes in the ring network interact cooperatively. If a node needs a file, it places the request on the ring, which is received by other nodes. Each node checks if it has the file. If so it will be transmitted to the node in need of it. Else, the request is sent in a broadcast manner to all the nodes. If the file is unavailable an error message is generated and sent to the node which generated the request. A typical unidirectional ring network is shown in the figure 1 below with the nodes and edges. Unidirectional ring network is less reliable, as with a link failure the system gets down. The next type of ring network that is encountered is the bidirectional ring network. In the case of bidirectional ring network the transmission is faster and reliable since the ring evaluates the shortest path and communicates using this shortest path.

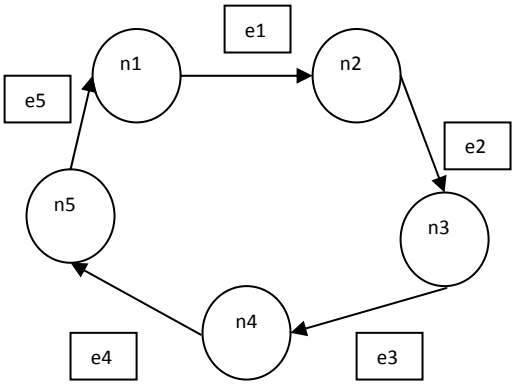

figure 1: Unidirectional ring network

If the link fails in one direction, communication is still possible by the links in the other direction. If a file is the most needed one during the operation, that file can be replicated in some of the nodes in the bidirectional ring and this enhances the efficiency of the ring, by reducing the data traversal. The next type of ring network is the double counter-rotating ring network. The architecture consists of two ring networks. One ring will be the back-up ring. The other ring is used for data and message exchanges. When a primary link fails, the stations adjacent to the failed link will attempt to isolate the failure through healing, in such a way as to form a single ring. If additional links fail, then the recovery mechanism removes the nodes or links forming isolated simple rings. Thus, the network always works with a high reliability. A monitoring program monitors the status of each ring and appropriately switches to the working ring. Chordal ring network is a familiar ring network. Here additional bidirectional links

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

called chords are connected to every node. If the conventional links or nodes fail, it is easily detected by means of message exchange and the other nodes are informed about this. Then the working nodes bypass this link and use chords for communication, enhancing reliability.[3]

4. RELIABILITY ANALYSIS OF AGENT BASED DISTRIBUTED SYSTEMS

Mobile agents are autonomous objects that are able to migrate from node to node in a computer network and perform specified tasks simultaneously. The concept of a mobile agent allows for asynchronous operation of arbitrarily complex tasks. When an agent decides to migrate to another node, the agent's code, data and execution state are captured and transferred to the next node, where it is re-instantiated after arrival. The actions, an agent has to perform on a single node are called a step. The ability to roam the network is provided by a middleware platform which is a mobile agent execution environment. Agents are broadly categorized as Proactive agents and Reactive agents. The reactive agents respond to events whereas the proactive agents achieve goals. Both are very widely used. The first step in the agent computing is the scenario development. This is followed by developing plans, events generation and configuration. Plans are stored in the plans library. Scenarios are complementary to goals in that they show the sequences of step that take place within the system. In developing goals, the first step is to build scenarios of how these goals will be part of various processes within the system. The core of a scenario consists of a sequence of steps. Possible steps are achieving a goal (GOAL), performing an action (ACTION), receiving a percept (PERCEPT), or referring to another use case scenario (SCENARIO). A goal is something that the agent is working on or towards achieving it. Often the goals are defined as states of the world that the agent wants to bring about. Goals give the agent its autonomy and proactiveness. A plan is a way of realizing a goal. Configuration is used to set the agent environment. To start with the agents are invoked. Next the execution cycle consists of the steps like, events are processed to update beliefs and generate immediate actions, goals are updated, and plans are selected from the plan library for achieving goals or handling events. A plan step is executed in the next plan, yielding new events, goals, belief, changes or actions. In the present scenario five types of agents namely creator agent, updater agent, file search agent, remote desktop capture agent and the shortest path agent are created and configured. Searching for data and files is an important aspect in the programming environment. The agent whenever it wants to search for a file, it creates multiple agents, which searches the file parallel and returns the search result. The shortest path agent computes the shortest path from specified node to other nodes. The agents migrate over the shortest path and perform the specified operation. Another type of agent is the remote desktop capturer agent, which captures the specified desktop and user can manipulate the captured desktop from his own system. This is sometimes needed for controlling. Thus the various agents collaborate and achieve the desired goals.[8]

5. RELIABILITY ANALYSIS OF CLUSTERS

Very often applications need more computing power than a sequential computer can provide. The solution to this problem is the cluster system which consists of multiple processors. At a basic level a cluster is a collection of workstations that are interconnected via some network technology. A cluster works as an integrated collection of resources and can have a single system image spanning all its nodes. The reliability analysis of clusters depends on the system reliability analysis, system availability analysis and program reliability analysis. The server is the most important part of the cluster system, but forms a single point of failure. To eliminate this, a redundant server is added to the system. The periodical transmission of heartbeat messages traverses across a dedicated serial link, as well as an IP channel, and works as healthy detection of the working server. When a failure of the primary server occurs, the system can automatically transfer the control of services to the failover server, allowing services to remain available with minimal interruption. The failed server can then be repaired while the application continues on the failover server. The program reliability analysis of the cluster is performed by constructing a task graph. Task graph is a directed acyclic graph in which each vertex represents a task and link represents control flow. This graph is constructed based on the problem complexity and the number of sub tasks been splitted from primary task. Initially task 0 acts as a root node task and other successive tasks that follow the root node are the descendants of root node and this leads to the construction of a tree. Task 0 after collecting the input data, distributes the data to the other processes by using a simple message passing

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

scheme. The slave node works and returns the result to the master. If at any point of time any node is malicious, a null value is returned to the master. When the master node receives the null value, it invokes the cluster information gathering algorithm to detect the malicious node. This node is temporarily suspended, repaired and added later. Thus, it is found that the cluster system is a highly reliable and an available system executing programs successfully.

6. EXPERIMENTAL OUTCOME AND ANALYSIS

A typical distributed computing system is formed using n processing elements in the form of a ring network. In the ring network every node has two neighbors a left neighbor and a right neighbor. If unidirectional link is established between the distributed processes, the communication is possible in one way. The reliability analysis in the ring network is approached using probability. The distribution of probability for process not available is 0. This can occur in many cases in distributed systems. There may be several servers, which may run in the environment. Sometimes server would not have been launched in the system or it may be faulty. In this case, it leads to a probability of 0. The next category of reliability is, process is available and is loaded onto memory. No other activity takes place. It is just the loading of the processes onto the main memory. The probability of this category is 0.2. The next category of probability is 0.3 – 0.8. Here processes have various types of errors like programming errors, data errors, database errors, socket errors and so on. The final category is the reliability of correct state in which the reliability analysis of the processes with various status like the files are available, the link is perfect and transition is high. This is in the range 0.9-1. In order to perform reliability analysis the statuses specified are simulated and tested. The process reliability analyzer analyzes these processes. As part of the reliability analysis, a unidirectional ring network is constructed. The processes with various faults are simulated. Table 1 shows the processes with the corresponding probability as assigned by the reliability analyzer. A graph is plotted for the simulation and is shown in figure 2**.** Similarly the reliability analysis of other ring networks is carried out.

Table 1: Reliability of distributed processes

Process (Category)	Reliability
Proc1 – does not exist	0
Proc2 – input data exception	0.3
Proc3 – process with problems in database connectivity	0.6
Proc4 – process with socket error	0.8
Proc5 - correct process	1

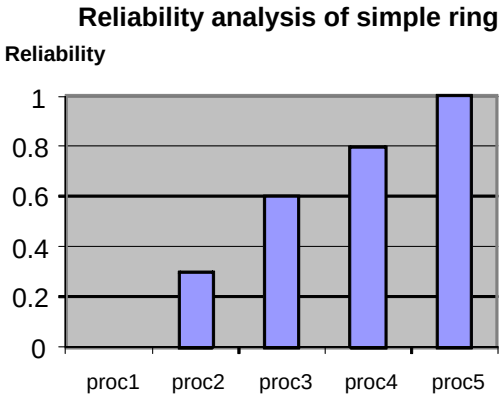

Figure 2: Plot of process reliability

The reliability analysis of multi agent system is simulated using JACK tool kit. A scenario is created where the agents like creator agent, updater agent, remote desktop agent, search agent and shortest path agents are deployed. The agents execute events using plans. The time taken for execution of the events is calculated. It is observed that the non agent based systems take seconds to complete a specified task whereas, agent based systems takes time in the range of milliseconds to complete the same specified task. It is also observed that the reliability of agent based system is high when compared with the non agent based systems. Table-2 shows the analysis of multi agent system where the round trip time is accounted for. Figure 3 shows the graph for this.

it.org E-ISSN: 1817-3195

Table 2: Roundtrip time of multi-agents

No_of_nodes inCIG	Correct messageTransfer time (ms)	Fault Detection (FD) time(ms)
4	0.001735	0.001693
5	0.003034	0.002985
6	0.002788	0.002985
7	0.003610	0.003571
8	0.005117	0.005386

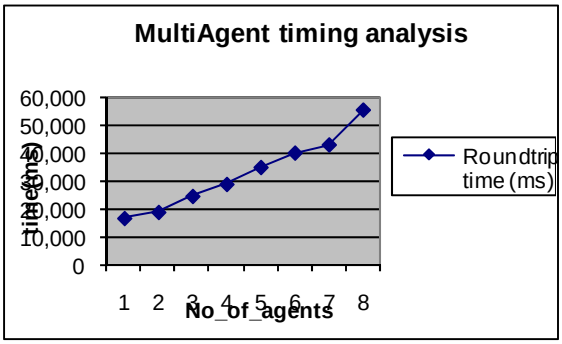

Figure 3: A plot of agent's round trip time

The next issue is the reliability analysis of cluster system. The reliability of cluster system depends upon the system reliability and program reliability. The system reliability of the cluster is enhanced by means of the dual server system. If the primary server fails, the back-up server immediately takes over, thus improving system reliability. Availability analysis is carried out in the cluster system and it is derived that the availability of the cluster system is enhanced by redundancy. It is found that the configuration file of the cluster plays a major role in the system analysis. Program reliability analysis in a loosely coupled system is performed by constructing a task graph. This graph is constructed based on the problem complexity and the number of sub tasks been splitted from primary task. A simple message passing scheme is used to analyze the nodes in the task graph. If any node receives null message, it is found to be malicious. This node is temporarily suspended from the network. It is repaired and added to the network. The reliability of the cluster system is high, provided the requirement of the nodes is lesser than the maximum node capacity of the cluster. Figure 4 depicts the cluster analysis. If at any point of time, the delay increases the node is suspected for its maliciousness. If the time is within the limit the node is perfect. Table 3 shows the cluster information gathering analysis.

Table 3: cluster information gathering analysis

No. of Agents	No of operations	Roundtrip time (ms)
3	4	16,647
4	5	18,782
6	6	28,691
7	7	34,679
8	8	39,789
9	9	42,658
10	10	54,979

Figure 4: CIG Analysis

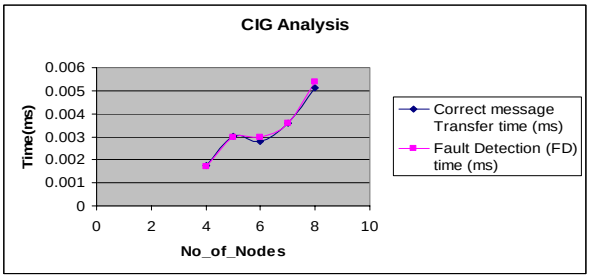

Figure 3: CIG analysis

7. CONCLUSION

Recently Distributed System is the most popular network that is available and it is advantageous. End users attached to such a network need an environment which is more sophisticated and user friendly. Hence it is important that reliability analysis is performed on such systems. This paper addresses the issue of reliability analysis which is performed in ring networks, agent based systems and clusters In the case of ring networks it is derived that the bidirectional ring, dual ring and chordal ring networks are highly reliable. Agent based distributed system is a highly reliable and faster network. Agent enhances the performance of

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

the distributed system. In the case of cluster analysis redundancy is found to increase the reliability. Clusters networks are proved to be highly available. The features offered by Clusters are high performance, expandability and scalability, high throughput and high availability.

REFERENCES

[1]. Deng-Jyi Chen and M. S. Lin, "On Distributed Computing Systems Reliability Analysis under Program Execution Constraints". *IEEE Transactions on Computer*, Vol. 43,No.1,1994 pp. 87-97.

[2]. Deng-Jyi Chen and T. H. Huang, "Reliability Analysis of Distributed Systems based on a Fast Reliability Algorithm", *IEEE Transactions on Parallel and Distributed Systems*,Vol.3No.2,1992 pp. 139-154.

[3]. Ming-Sheng Lin et al, "The Distributed Program Reliability Analysis on Ring Type Topologies", *Computers and operations Research*,Vol.28,No.7,2001 pp. 625-635.

[4]. Jason L.Cook and Jose Emmanuel Ramirez-Marquez, "Two-terminal reliability analyses for a mobile ad hoc wireless network", *Reliability Engineering and System Safety*,Vol.92,No.6,2007,pp. 821-829.

[5]. Yue Ma, James J.Han and Kishor S. Trivedi, "Composite Performance and Availability Analysis of Wireless Communication Networks", *IEEE Transactions on Vehicular Technology*,Vol.50,No.5,2001,pp.1216-1223.

[6]. Chung-Chi Hsieh and Yi-Che Hsieh, Reliability and Cost Optimization in Distributed Computing Systems, *Computers and Operations Research*, Vol.30,No.8,2003,pp. 1103 -1119.

[7]. Ming-Sang Chang et al, "The Distributed Program Reliability Analysis on Star Topologies", *Computers and Operations Research*, Vol.27No.2,2000,pp. 129-142.

[8]. Mosaab Daoud and Qusay.H.Mahmoud, "Monte Carlo Simulation-based Algorithms for Estimating the Reliability of Mobile Agent-based Systems", *Journal of Network and Computer Applications*, Vol.31,No.1,2008,pp. 19-31.

[9]. Hung M.Dao and Charles B.Silio, Jr., "Ring-network with a constrained number of consecutively-bypassed stations", *IEEE Transactions on reliability*, Vol.47,No.1,1998,pp.35-43.

[10]. Jinn-Shyong Yang et al, "Reducing the height of independent spanning trees in chordal rings", *IEEE Transactions on Parallel and Distributed Systems*, 18(5),2007, pp.644-657.

[11]. Fabien Houeto et al, "Reliability and cost evaluation of third generation wireless access network topologies: A case study", *IEEE Transactions on reliability*, Vol.51,No.2,2002,pp. 229-239.

[12]. Geir Egeland and Paal E.Englestad, " The availability and reliability of wireless multi-hop networks with stochastic link failures", *IEEE journal on selected areas in communication*, Vol.27,No.7,2009,pp. 1132-1146.

[13]. Teerawat Issariyakul et al, "End-to-End batch transmission in a multihop and multirate wireless network: latency,reliability and throughput analysis", IEEE Transactions on mobile computing, Vol.5,No.9,2006,pp. 1143-1155.

[14]. Masaab Daoud, Qusay H. Mahmoud, "A hybrid approach to reliability estimation of mobile agent –based systems", Canadian conference on Electrical and computer engineering CCECE/CCGEI, Saskatoon,2005,pp. 1344-1350.

[15]. Fred Stann, John Heidemann, Rajesh Shroff and Muhammad Zaki Murtaza, "RBP: robust broadcast propagation in wireless networks, Fourth international conference on embedded networked sensor systems Sensys ' 06, Boulder, Colarado, USA,2006,pp. 85-98.