International Journal of Advanced Research in Electronics and Communication Engineering (IJARECE)
Volume 3, Issue 6, June 2014

Fault Tolerance and Task Allocation in Distributed Mobile Computing System

Shilpa Gambhir[#1], Er. Sonia Goyal[*2]

[#1] Student (Mtech ECE), Electronics and Communication Department, Punjabi University, Punjab, India
[*2]Assistant professor, Electronics and Communication Department, Punjabi University Patiala, Punjab, India

Abstract - **Mobile Computing System is a distributed system where one of the processes is known as Mobile Node. As the mobile network is defined as a relatively dense collection of mobile entities connected by a wireless link, without any administration or fixed support. In the mobile network no central authority is present due to which the network disconnection is very frequent between the mobile nodes. A chance of errors in the mobile distributed network is very high. The load is equally divided among the mobile node to enhance the network efficiency and to reduce the task execution time. When the load is not equally divided among the mobile nodes, chance of error occurrences will be increased. The approach of fault tolerance is required to reduce the number of error rates in mobile distributed network. In this paper we present a load balancing algorithm which is used to effectively allocate the task among nodes in a distributed system.**

Keywords – **distributed computing system, fault tolerance, task allocation, check pointing, load balancing.**

I. INTRODUCTION

Computing System is a system which is consists of one or more computers and associated software with common storage. A distributed system consists of multiple autonomous computers that communicate through a computer network. Distributed computing utilizes a network of many computers, each accomplishing a portion of an overall task, to achieve a computational result much more quickly than with a single computer. A distributed system is one in which hardware or software components located at networked computers communicate and coordinate their actions only by message passing[1].In the term distributed computing, the word distributed means spread out across space. Thus, distributed computing is an activity performed on a spatially distributed system.These networked computers may be in the same room, same campus, same country, or in different continents. Distributed computing utilizes a network of many computers, each accomplishing a portion of an overall task, to achieve a computational result much more quickly than with a single computer. Distributed Computing System is heterogeneous in nature. So different type of hardware and software are required are required to build the distributed system.

Properties of Distributed Computing System

1. Each computational entity has local memory.
2. The entities communicate with each other with the help of message passing.
3. The system has to tolerate failures in individual computers.
4. The system structure and links may changes during the execution of the distributed programs.
5. Each system is only aware about the input of the system.

The rapid growth in Internet users and diverse services has highlighted the need for intelligent tools that can assist users and applications in delivering the required quality of services. As distributed systems gain complexity owing to increasing user needs, monitoring and adaptations are necessary to keep them fit and running. Mobile computing system has become a new paradigm for distributed real-time systems because of their inherent advantages[1].

The Distributed systems can reduce the load on the central authority. The central authority can distribute the task to various other mobile systems. This approach will enhance the network throughput, reduce execution time and reduce battery consumption. In the past, extensive work has been done to provide fault tolerance in wired networks and high speed data networks (e.g., public switched telephone networks and asynchronous transfer mode networks)[2]. These attempts have improved the reliability, availability, and survivability of the networks under study. However, there is inadequate study on the survivability of wireless access networks, even though they are more vulnerable than wired networks. Fault tolerance is the ability of a system to withstand failure and continue to provide service in the event of an internal or external error. Fault tolerant systems are designed to ensure that in the event of a failure, crash, or a major user error, data are not lost and the system can continue to provide its specified services, thereby increasing the reliability and dependability of a system usually by masking the software or hardware faults[3][4].

ISSN: 2278 – 909X

The task allocation among the mobile nodes is done with the use of task allocation model[5]. In task allocation model, on the basis of capacities of processors and communication links, we allocate the tasks among processors.

II. LITERATURE REVIEW

SajjadHaider, NaveedRiaz Ansari, Muhammad Akbar, Mohammad RazaPerwez and KhawajaMoyeezUllahGhori studied fault tolerance in distributed paradigms. They presented a comprehensive classification of errors, failures and faults that can be encountered in a Distributed environment [3].

Tome Dimovski, PeceMitrevski proposed a distributed transaction processing model in mobile environment which considers two communication scenarios, i.e. one when mobile hosts can connect to the fixed network and the second when they cannot. A decision algorithm is responsible for making a decision for a mobile host when it is disconnected from the fixed network for a certain period of time [6].

Jorge E. Pezoa, SagarDhakal, and Majeed M. Hayat studied the performance of DCS by redundancy approach. They present a framework to analytically characterize the service reliability of DCS in case of communication uncertainties and topological changes due to node deletions. The presented analysis is based upon the regeneration theory that exploited to derive a system of difference-differential equations characterizing the service reliability [5].

I. Maatouk, E. Chatelet, and N. Chebbo show the reliability of multi-state system with load sharing approach. They have presented an approach for evaluating the dynamic performance distribution of multi-states distributed computing system with dependent components. The dependency introduced the common cause failure and the load sharing between system components [6].

Vinod Kumar Yadav, MahendraPratapYadavand Dharmendra Kumar Yadav in [7], tried to solve the problem of maximizing reliability of heterogeneous distributed computing system where random node can fail permanently. They determined the candidate nodes for tasks that can satisfy to its requirements. Then they utilize the load sharing policies for handling the nodes failure as well as maximizing the service reliability of DCS.

III. PROBLEM STATEMENT AND SOLUTION

Consider a DCS here with 'N' heterogeneous computing nodes. We also suppose that workload is divided into 'M' tasks. 'M' tasks are divided among 'N' nodes depending on various parameters. We have to effectively distribute the tasks among the nodes. This is because if load is not equally divided among the nodes, chance of errors occurrences will be increased.

We use load balancing algorithm. Load balancing is based on the redistribution of processes among the processors during execution time. This redistribution is performed by transferring tasks from the heavily loaded processors to the lightly loaded processors with the aim of improving the performance of the application.Load balancing algorithm is defined by three inherent policies:

• Information policy, which specifies the amount of load information made available host node or base station

• Transfer policy, which determines the conditions under which a task should be transferred, that is, the current load of the nodes and the size of the task under consideration

• Placement policy, which identifies the processing node to which a task should be transferred.

On the basis of capacities of processors and communication links, we allocate the tasks among processors. Failure problem can be solved by task redundancy. Task redundancy is provided by backup system that is attached with each node of the DCS. Here, it is noted that backup system does not provide service to any tasks.

IV. PERFORMANCE EVALUATION

The proposed idea has been simulated in MATLAB. The computing nodes are heterogeneous in nature. It means they have different computational capabilities and failure rates and execution time.
The algorithm contains three major parts. The first part reads the 'M' tasks, 'N' Nodes and various parameters as failure rates and execution time of head node and sub nodes. Thus determines the candidate nodes on the basis of requirements of tasks and resources available on the processors. In the second part it assigns the tasks among most appropriate candidate nodes. In this assignment it considered the parameters: failure rate, and execution time. In part three it handle the case of node failure. When nodes failed before executing all the tasks assigned onto it. It transfers the remaining tasks among the next appropriate candidate nodes.

Figure 1 shows the placement of nodes. Here tree topology is considered. Head node assigns tasks to sub nodes. Sub nodes further assign the task to nodes. Here we have to 2 sub nodes, which act as head nodes to their respective nodes. Like sub node 1 acts as head node to node 3, 4 and 5. And sub node 2 acts as head node to node 6 and 7
The no. of tasks and the maximum failure rate and execution time of head node and others node are entered as input. Now the second part of program works. It chooses the candidate

ISSN: 2278 – 909X

nodes on the basis of considered parameters, failure rate and execution time.

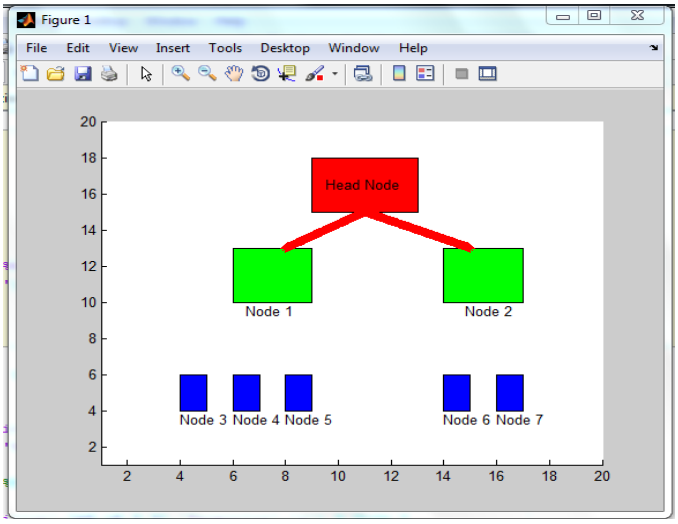

Figure 1: Placement of head nodes and sub node

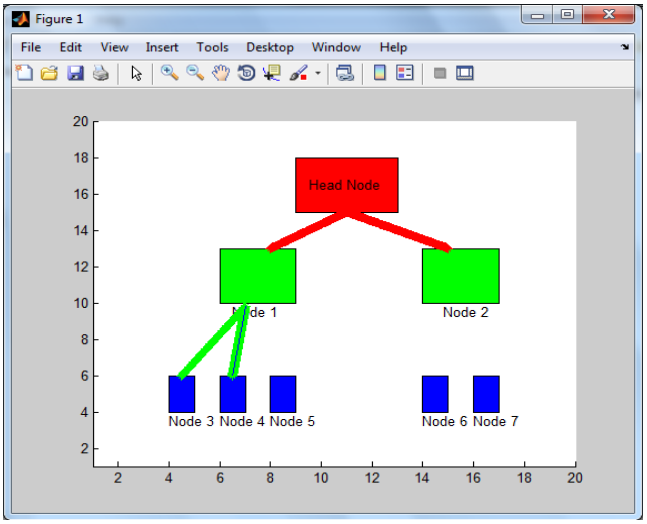

Figure 2: Sub node assigning tasks to candidate nodes.

The no. of tasks and the maximum failure rate and execution time of head node and others node are entered as input. Now the second part of program works. It chooses the candidate nodes on the basis of considered parameters, failure rate and execution time.

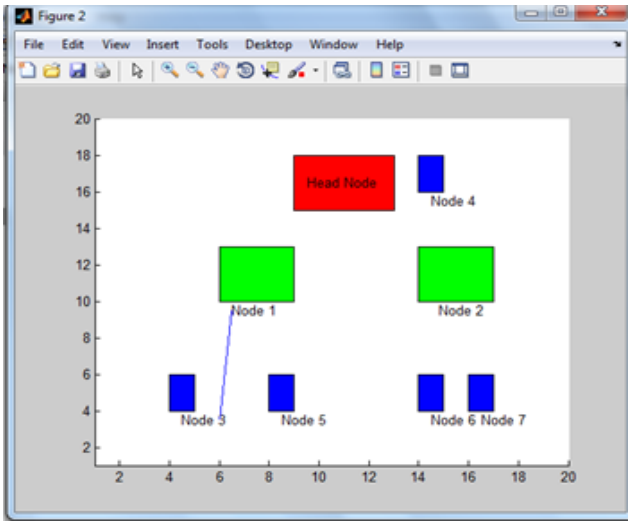

Figure 3: Sub node checking availability.

As shown in the figure 3, the sub node when pings the node to which the task is allocated. When the node will not respond in the threshold time, it will be detected as faulty node. As in the figure 4, node no 4 is detected as faulty node

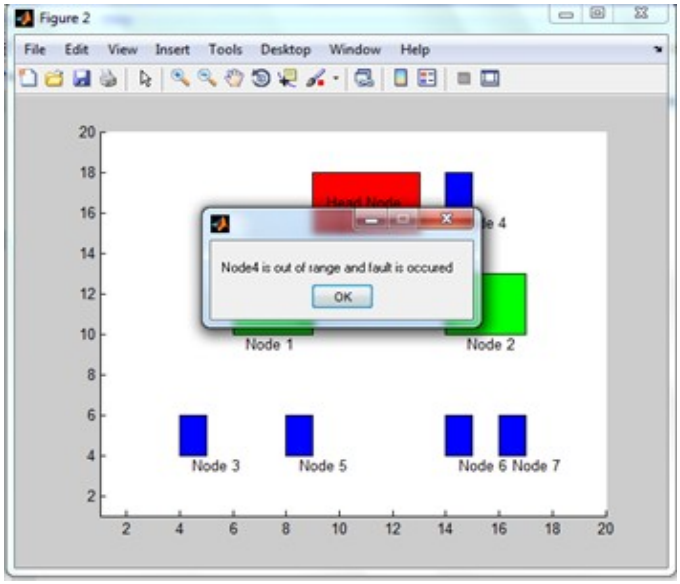

Figure 4: Faulty node detected

ISSN: 2278 – 909X

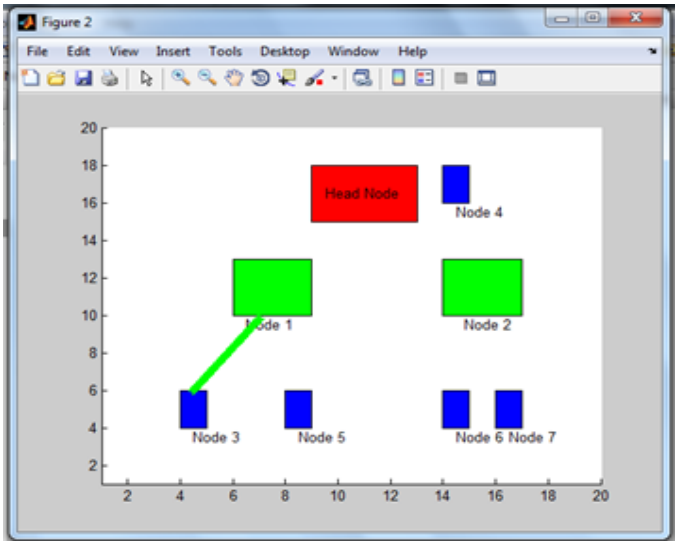

Figure 5: Task reassignment

As illustrated in the figure 5, sub node will send message to node which is not faulty to complete the task of the particular node on which fault is occurred. In this way task is effectively distributed among the nodes in the system.

V. CONCLUSION AND FUTURE WORK

We studied the concepts of fault tolerance and task scheduling algorithms. For task scheduling, Load balancing algorithm is used to handle the execution of tasks. It effectively assigns the tasks to the nodes. The limitation of load balancing is that when number of processes on a node is fluctuating, it can't assign the tasks effectively. The major disadvantage of dynamic load balancing schemes is the run-time overhead due to: the load information transfer among processors, the decision-making process for the selection of processes and processors for job transfers, and the communication delays due to task relocation itself. Future work can be done on reducing time overhead.

REFERENCES

[1] Prof. Elizabeth White, "Introduction to Distributed system" , Distributed Software Systems.

[2] PetruEles, "DISTRIBUTED SYSTEMS" http://www.ida.liu.se/~petel

[3] SajjadHaider, NaveedRiaz Ansari, Muhammad Akbar, Mohammad RazaPerwez, KhawajaMoyeezUllah Ghori1, "Fault Tolerance in Distributed Paradigms", International Conference on Computer Communication and Management, 2011.

[4] Tome Dimovski, PeceMitrevski, "Connection Fault-Tolerant Model for Distributed Transaction Processing in Mobile Computing Environment"ITI 2011 33rd Int. Conf. on Information Technology Interfaces,June 27-30, 2011, Cavtat, Croatia

[5] Jorge E. Pezoa, SagarDhakal, and Majeed M. Hayat, Maximizing Service Reliability in Distributed Computing Systems with Random Node Failures: Theory and Implementation, IEEE trans. on parallel and distributed system, 2010.

[6] I.Maatouk, E.Chateet, N.Chebbo, Reliability of multi-states system with load sharing and propagation failure dependence, 978-1-4577-1232- 6/11/2011 IEEE.

[7] Vinod Kumar Yadav, MahendraPratapYadavand Dharmendra Kumar Yadav , "Reliable Task Allocation in Heterogeneous Distributed System with Random Node Failure Load Sharing Approach", International Conference of Computing Science, 2012

[8] Rajwinder Singh and Mayank Dave, Senior Member, "Antecedence Graph Approach to Checkpointing for Fault Tolerance in Mobile Agent Systems", IEEE transactions on computing, 2013

[9] Shan Zhang and Jian-ping Wu, "Construction of Distributed and Heterogeneous Data Sharing Platform" ,International Conference on Web Information Systems and Mining, 2009

[10] RajwinderSingh, Mayank Dave, "Using Host Criticalities for Fault Tolerance in Mobile Agent Systems", 2nd IEEE International Conference on Parallel, Distributed and Grid Computing, 2012

[11] Zhongkui Li and ZhishengDuan, "Distributed Tracking Control of Multi-Agent Systems with Heterogeneous Uncertainties", 10th IEEE International Conference on Control and Automation (ICCA) Hangzhou, China, June 12-14, 2011.

ISSN: 2278 – 909X