ISSN (Online) 2278-1021
ISSN(Print) 2319 5940

International Journal of Advanced Research in Computer and Communication Engineering
Vol. 4, Issue 7, July 2015

To Propose a Novel Technique to Overcome Fault Tolerance in DCS during Node Mobility

Prabhjot Kaur[1], Shashi Bhushan[2], Karan Mahajan[3]

M. Tech Student, Information Technology, Chandigarh Engineering College, Landran, Mohali[1]

Professor& Head of Department, Information Technology, Chandigarh Engineering College, Landran, Mohali[2]

Assistant Professor, Information Technology, Chandigarh Engineering College, Landran, Mohali[3]

Abstract: Distributed systems play an important role on achieving good performance and high system utilization. The goal of a job scheduling system is to efficiently manage the distributed computing power of workstations, servers, and supercomputers in order to maximize job throughput and system utilization. There are many issues of distributed computing system which are discussed in this paper in brief. The main aim of the paper is to focus on fault tolerance and recover fault with less processing time. The proposed algorithm is assign tasks to other nodes only when master node moves from its original position using weights.

Keywords: Computing, master node time, e-cost, profit.

I. INTRODUCTION

Computing is an activity which is used for creating and designing computers. Computing includes all the software and hardware scheming, structuring and emergent processes and manages them properly. Parallel and distributed are the types of computing. In parallel computing, memory is shared between all the processors to exchange information. Bit-level, instruction level and data level are its type [1].

A distributed computing is software systems in which components are located on different attached computers communicate and organize their actions by transferring messages [2]. In distributed system computers are located at faraway places. Computers are connected with number of servers. If there is chance of any server failures, it gets data from other servers also. Another point is scalability. Due to scalability nature, a number of computers can be added at any time to increase its strength. Another point is redundancy. As discussed above, it is a combination of various smaller machines it is not that much expensive [3]. It can easily affordable and extendable. Mobile Computing is one of the types of distributed computing. Mobile computing is a branch of computing that comes under sharing of data instead of local servers. This policy of distributed computing is completed through pooling of all computer resources together. It is under the power of solve larger scale computational demands. It is also permissible users and its clients to access their personal files at any computer with the help of internet without installation. Software not under any external users. Services [4].

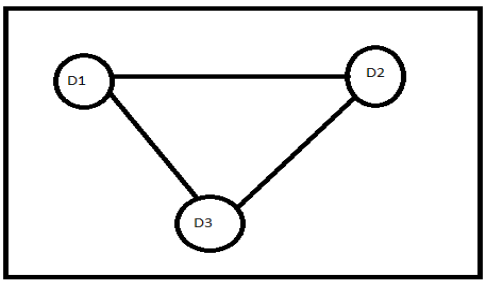

Fig.1.1 Distributed computing system

Grid computing are another types or category which comes under distributed computing. Grid computing is a computing environment with high performance to.

There are number of grids are present in grid computing which combine together to form cluster which helps to react at the common objective [5].

A. Challenges of Distributing Computing System:

There are many challenges in distributed computing systems. These are:

a) Resource Discovery and Selection:

The dynamic environment of the wireless mobile makes necessary the use of sophisticated mechanisms for resource discovery and selection. The number of authenticated machines and resources that are available in the mobile infrastructure is continuously updated. Some of the required parameters are resource accessibility, system workload, network performance, etc. A financial criterion of the resources used should be under consideration for the proper resource selection [5].

b) Information Security:

Due to their wireless nature, devices in the mobile communicate and pass information over standard radio frequencies that can be easily tapped. A number of encryption standards such as WEP have been devised to ensure data security and integrity over these otherwise insecure transmission frequencies [4].

c) Flexibility:

The distributed system should be flexible so that modifications and enhancement can be done easily by the users. Distributed systems are located at different places and having different servers so it is necessary it can be enhanced and modify easily. Sometimes inconsistency problem may occur due to its flexible nature.

d) Scalability:

System should be designed in this manner that it is easily coping up with it increase. It should avoid central algorithms

Copyright to IJARCCE

ISSN (Online) 2278-1021
ISSN(Print) 2319 5940

International Journal of Advanced Research in Computer and Communication Engineering
Vol. 4, Issue 7, July 2015

and central entities. It should be perform most of the operation at the client work station.

e) Fault Tolerance:

The system must be resistance to faults. In future if any fault may occur it doesn't degrade its performance. The best is considering that system which is free from all the faults and has immaculate and impeccable performance. So there are various techniques which premeditated to make the system fault tolerant.

II. REVIEW OF LITERATURE

In paper [1] they represented a novel fault-tolerance mechanism to have the following advantageous features appropriate for large scale and dynamic hierarchical mobile agent-based monitoring organizations. It is in favor of failure detection which has fast functionality transmitting heart-beat messages to its immediate higher level manager with low failure-free overhead by each domain manager. Moreover, it allows consistent failure detection actions to be performed continuously in case of agent formation, relocation and annihilation, and is able to execute consistent takeover actions even in concurrent failures of domain managers. In this paper [2], they proposed a Connection Fault- Tolerant Model for mobile environment which considers two communication scenarios first is when MHs can connect to the fixed network through MSS, and the second when MHs cannot connect to the fixed network. They presented a Decision Algorithm which is responsible for making a decision for a MH when corresponding MH-Ag cannot communicate with its MH for a defined period of time. The CFT model reduces the blocking time of resources at the fixed devices provides fast recovery from connection failures owing to mobility of mobile devices and increases the number of committed mobile transactions. In this paper [3], they presented the mobile Ad hoc networks are distributed environments characterized by a high mobility and limited battery resources. Mobiles nodes are under discussion to many errors. Their work contributes to the resolution of two points. First, they recommend an algorithm for modeling in wireless networks by grouping them. After that, they study the fault tolerance by prediction of disconnection and partition in network. In this paper [4] they addressed the distributed tracking control problem for multi-agent systems with heterogeneous uncertainties and a leader whose control input might be nonzero and not available to any adherent. Both distributed continuous static and adaptive controllers based upon neighboring states have been designed to guarantee the uniform ultimate boundedness of the tracking error for each follower. A sufficient condition for the existence of these distributed controllers is that each agent is stabilizable. In paper [5] presented mobile agent based fault prevention and detection technique where the team of mobile agents monitor each host in mobile agent based system. They proposed an approach to introduce fault tolerance in multi agent system through check pointing based on updating of weights from time to time while calculating the dependence of hosts. From experimental results it can be safely inferred that the proposed monitoring technique for multi agent distributed application may effectively increase system's fault tolerance beside effective recognition of vulnerabilities in system.

In this paper [6] they represented Mobile Agent technology to improve the flexibility and doing by promise as a powerful agent and its mechanism. Mobile agent systems must also provide a customizability of applications with its ability to additional feature for the security for the agent from dynamically deploy application components across the malicious host and the security of the host from a network. The architecture proposed in this paper prototype systems satisfy all the requirements to address the above issues can be used to extend to provide a secure and reliable architecture, suitable for features of the existing systems.

III. CHECK POINTING IN DISTRIBUTED COMPUTING SYSTEM

The real time distributed systems like grid, robotics, nuclear air traffic control systems etc. are highly responsible on deadline. Any mistake in real time distributed system can cause a system into collapse if not properly detected and recovered at time. Fault-tolerance is the important method which is often used to continue reliability in these systems. Fault Tolerance is important method in distributed computing because nodes are distributed geographically in this system under different geographically domains throughout the web wide. The most difficult task in distributed computing is design of fault tolerant is to verify that all its reliability requirements are meet [8] [13].

A. Check Pointing:

Basically this technique is used to restore the process to certain point after failure occurs. Fault Tolerance can be achieved through various types of redundancy. Check-point start is the common method. In this method an application starts from the earlier checkpoint after a fault. Application may not be able to meet strict timing targets. Checkpoints are of various types. These are discussed as follow:

a) Co-ordinate checks pointing:

It is also known as synchronous checkpoint process which takes place checkpoints in such manners that results become consistent. It follows 2-phase commit state. In first phase check points are temporary and in second phase check points are permanent. The positive point of this process is that one permanent and one temporary check point is essential to be stored [10].

b) Un-coordinate Check-pointing:

In this technique, process co-ordinate with their check pointing activities and each process records it local checkpoint independently. It requires cascade rollbacks which lead to the initial state due to domino effect. It is responsible to create consistent global state recovery. Consistency can be track using dependencies. This technique may use useless checkpoints which incurred overhead problem.

c) Message logging check-pointing:

In this technique, all the interaction takes place through messages only. Each message received by a process that is saved in messages log on stable storage [10].

d) User Triggered Check-pointing:

It requires user interaction. User should have knowledge of all the processes related to it. The main problem is identification of location by the users.

Copyright to IJARCCE
DOI 10.17148/IJARCCE.2015.4759

ISSN (Online) 2278-1021
ISSN(Print) 2319 5940

International Journal of Advanced Research in Computer and Communication Engineering
Vol. 4, Issue 7, July 2015

IV. PROBLEM FORMULATION

Distributed execution has been proposed as a means to reduce energy consumption in mobile applications. These solutions partition a mobile application into local and remote parts, so that energy intensive functionality is executed remotely at a server. Because modern mobile devices are network-enabled, offloading functionality to a remote server presents a promising avenue for reducing energy consumption. Unfortunately, this optimization also makes the application vulnerable to network outages, as mobile networks are characterized by high volatility. A key difference of our approach is that it can execute energy-intensive functionality on the server without having to partition the application. Instead, it efficiently replicates state to switch between local and remote executions, both to reduce client energy consumption and to tolerate network outages. When the network is operational, energy-intensive functionality is offloaded to the server by transferring only the program's state needed for the remote execution. Efficient check pointing synchronizes local and remote executions. If the network becomes disjointed during the offloading, the remote execution is redirected back to the mobile device. Thus, network outages only inhibit energy optimization rather than rendering the application unusable. The distributed systems follows are the rule of divide and conquer scheme in which task is divided into multiple other systems. The fault may occur in the network due to system failure due to overloading in the system. In this work, the proposed methodology Solve the problem of fault tolerance and work on efficient task scheduling.

V. PROPOSED METHODOLOGY

In the present techniques there is one drawback that is node failure. A node failure problem occurs due to mobility of the node. In present algorithm there are number of nodes available. From these nodes candidate node will be chose on the basis of failure rate and minimum execution time. Here Master node set threshold value which includes two parameters one is failure rate and other is maximum execution time. The nodes which have equal to and less failure rate and minimum execution time are elected as candidate nodes by the master node.

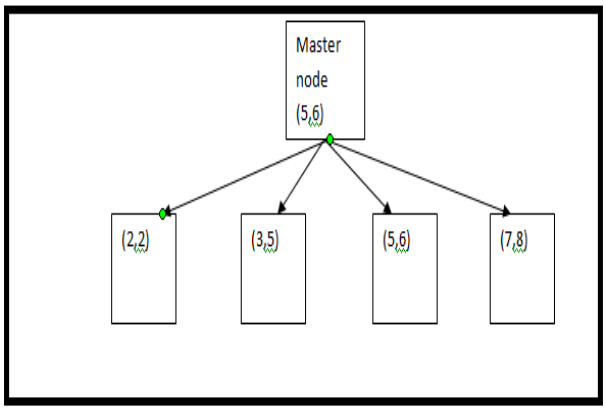

Fig. 3.1 Election of candidate node

In fig 3.1 master node has set threshold value of failure rate and maximum execution time is (5,6). The node which has value equal to and less this threshold value is selected as a candidate

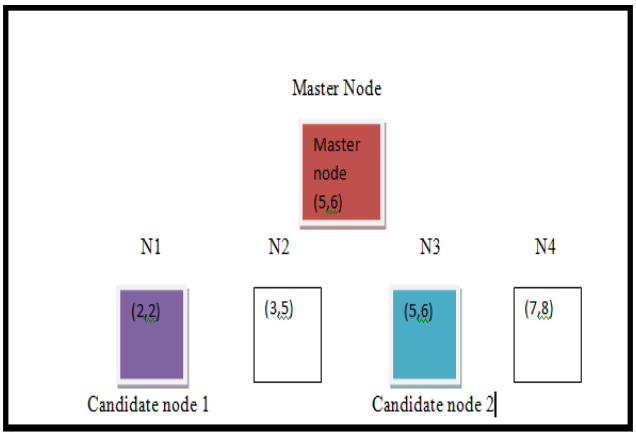

Fig.3.2: Candidate nodes are selected according to the m. failure and execution time

node. N1 has smaller value than threshold value so it will be a candidate node.N2 has one parameter less and other is high so it will not be chosen as a candidate node. N3 has value equal to the threshold than it will be selected as a candidate node 2. Again N4 has a greater value than threshold value than will not be selected as candidate node. After the selection candidate node will start perform their tasks? We will also enter number of task in this scenario. Suppose during execution of task one move moves from its location than failure occurs at that point. To overcome this problem a novel technique has been proposed which overcome the problem of failure due to mobility of the node.

In the proposed algorithm, we have added a new parameter in the present algorithm that is master node time. Master node time is the result time to join the end users. It is for node collaboration. After that we will find out E-cost based on maximum execution time plus master node time. Using E-cost. We will find out profit based upon e-cost plus failure of each node. At the end we will calculate weights according to profits. During mobility of node, the node which has been moved from its location, the task of that node will be assigned to the node which has the highest weight. Weight will be calculated according to the above mention formulae.

The proposed idea will be implemented in NS2 which is widely used in all areas of TCL script, tool command, in education and research at universities.

VI. CONCLUSION

The approach of fault tolerance is required to reduce the number of error rates in mobile distributed network. The task allocation among the mobile nodes is done with the use of task allocation modal. In this paper, novel technique has been proposed which reduces the fault detection time in the network and reduces the resource consumption to execute the allocated tasks using weight based technique. The proposed algorithm is based on the failure rate, minimum execution and time taken by the master node scheme for fault recovery and concurrent execution of processes for the process execution. This technique leads to reduce in processing time and reduce in energy consumption.

Copyright to IJARCCE
DOI 10.17148/IJARCCE.2015.4759

ISSN (Online) 2278-1021
ISSN(Print) 2319 5940

International Journal of Advanced Research in Computer and Communication Engineering
Vol. 4, Issue 7, July 2015

REFERENCES

[1] Jinho Ahn, "Lightweight Fault-tolerance Mechanism for Distributed Mobile Agent-based Monitoring" IEEE, 2008
[2] Tome Dimovski, Pece Mitrevski, "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
[3] Asma Insaf Djebbar, Ghalem Belalem, "Modeling by groups for faults tolerance based on multi agent systems", IEEE,2010
[4] Sreedevi R.N, Geeta U.N, U.P.Kulkarni, A.R.Yardi, "Enhancing Mobile Agent Applications with Security and Fault Tolerant Capabilities, 2009 IEEE International Advance Computing Conference (IACC 2009) Patiala, India, 6-7 March 2009
[5] Rajwinder Singh, Mayank Dave, "Using Host Criticalities for Fault Tolerance in Mobile Agent Systems", 2nd IEEE International Conference on Parallel, Distributed and Grid Computing, 2012
[6] Sreedevi R.N, Geeta U.N, U.P.Kulkarni , A.R.Yardi, "Enhancing Mobile Agent Applications with Security and Fault Tolerant Capabilities, 2009 IEEE International Advance Computing Conference (IACC 2009) Patiala, India, 6-7 March 2009
[7] Zhongkui Li and Zhisheng Duan, "Distributed Tracking Control of Multi-Agent Systems with Heterogeneous Uncertainties" , 10th IEEE International Conference on Control and Automation (ICCA) Hangzhou, China, June 12-14, 2013
[8] Jinho Ahn, "Lightweight Fault-tolerance Mechanism for Distributed Mobile Agent-based Monitoring" IEEE, 2008
[9] G. Susilo, A. Bieszczad and B. Pagurek, Infrastructure for Advanced Network Management based on Mobile Code, In Proceedingsof the IEEE/IFIP Network Operations and Management Symposium(NOMS'98), 1998, 322-333.
[10] Tome Dimovski, Pece Mitrevski, "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
[11] P K Yadav, M P Singh and Kuldeep Sharma. Article, "An Optimal Task Allocation Model for System Cost Analysis in Heterogeneous Distributed Computing Systems", A Heuristic Approach. Interntional Journal of Computer Applications 28(4):30-37, August 2011
[12] Prabhjot Kaur, Karan Mahajan, "Various techniques for fault tolerance in distributed computing system-A Review", International Journal of Computer Science and Mobile Computing, Vol.4 Issue.5, May- 2015, pg. 754-759.