International Journal of Science and Research (IJSR)
ISSN (Online): 2319-7064
Impact Factor (2012): 3.358

To Improve Fault Tolerance in Mobile Distributed System

Anupama Padha[1], Meenakshi Sharma[2]

[1]Student, Department of CSE, Sri Sai college of Engineering and Technology, Badhani, Punjab, India

[2]HOD of Department of CSE, Sri Sai college of Engineering and Technology, Badhani, Punjab, India

Abstract: ***Distributed system is a system which is heterogeneous in nature and better than centralized systems. There are some issues in distributed systems like fault tolerance, scalability, openness etc. In this paper we focused to improve the fault tolerance of the mobile distributed system so that it will take less time for execution when one node moves from its position as compared to other systems.***

Keywords: Distributed computing, Parallel Computing, Fault Tolerance

1. Introduction

Distributed computing utilizes a network of numerous computers, each accomplishing a portion of a taken as a whole task, to achieve a computational result much more speedily 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. Distributed system is better than centralized system in the following manners.

Scalability: By adding more machines as needed the system can easily be expanded
Redundancy: Several machines can provide the equivalent services, so if one is out of stock, work does not stop. As well, because many smaller machines can be used, this Redundancy does not need to be prohibitively expensive.

1.1 Parallel Computing

Parallel computing is a type of computation in which calculations are carried out simultaneously. It is a process of simultaneous use of multiple computer resources to solve computational problems [1]. SIMD, MIMD, MIMD, SISD are its taxonomy. In this process larger task is decomposed into smaller tasks. Parallel computing is fast. There are many different approaches and models of parallel computing. Parallel computing is the future of computing. It is time saving process. It is more difficult than sequential ones to write because of concurrency and bugs.

1.2 Distributed Computing System

It is heterogeneous in nature. Different types of hardware and software are required to build distributed system. The nodes which participated in distributed system are also having different types of characteristics. A computer program that runs on distributed system is known distributed program. The process of writing such types of languages is called distributed programming [14]. Grid computing and Cluster computing are types of distributed computing systems. A distributed system consists of a group of independent computers associated through a network and sharing middleware which enables computers to organize their behavior and to share the property of the system so that users identify the system as a single, incorporated computing facility [15].

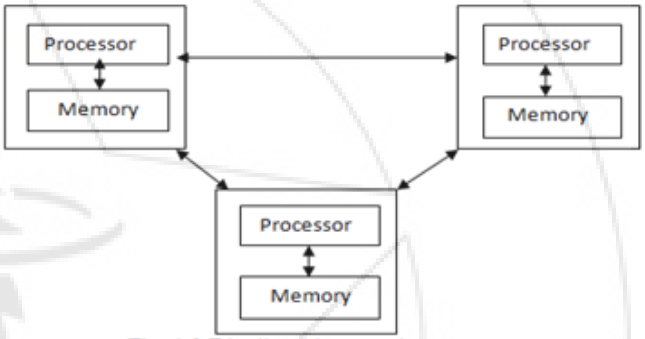


Fig. 1.2 Distributed computing system

In section 2 we will do literature survey and in section 3 we will introduce fault tolerance.

2. Review of Literature

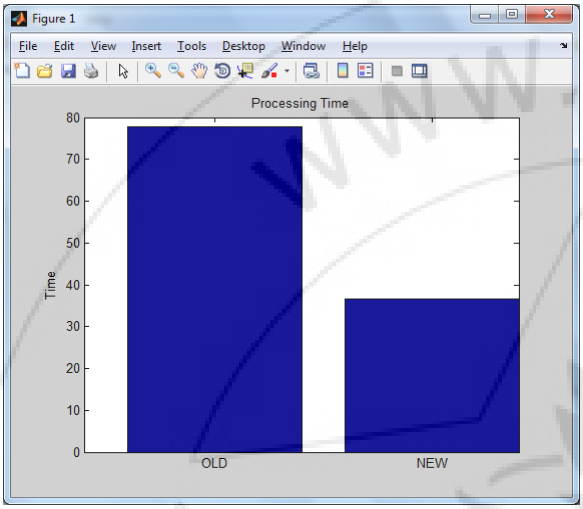
In paper [6] Shan Zhang and Jian-ping Wu , they mentioned criteria based on the combination of the description function of metadata, the distributed computation function of web service, the centralized storage ability of database, From their implementation experiences, logical centralization and physical dispersion of shared dataset were implemented. Data demander can conveniently access what he wants from this platform. Shared dataset's owner can easily publish, manage and maintain shared dataset by using this platform. The security of the shared datasets is also effectively ensured. Other issues for future work involve: (1) developing more efficient standardization methods for different types' data. Because of the complex structure of shared dataset, this platform only supports three types' data sharing. And the standardization approach for heterogeneous data is also relatively simple. (2) Methods to improve the security of shared dataset. Not only the user's download right should be checked before downloading shared dataset, but also we can develop sophisticated algorithms for encrypting and decrypting the shared dataset stream in order to raise the shared datasets security level. In paper [7] Vinod Kumar Yadav et.al they tried to solve the problem of maximizing reliability of heterogeneous distributed computing system where random node can fail permanently. Because the DCS is heterogeneous, so its various nodes

have different hardware and software characteristics. They have undertaken a two-phase hybrid approach to analyze the service reliability of heterogeneous DCS in the presence of communication and node uncertainty. On the basis of requirements of tasks, capabilities of processors and communication links, we determined the candidate processors. In second-phase we used the concepts of load sharing to handle the execution of tasks. They assigned each tasks to those processor that is most reliable and cost effective for this tasks. Instead of that, if any processor failed before executing the tasks assigned on it, it will transfer reaming tasks to next more reliable and cost effective processor. Repeat this phase until all the tasks got executed and compared with those derived by load sharing approach. The simulation result shows that, in most of the cases this hybrid solution gives the more cost effective results than load sharing approach. For a small test case of eight tasks, it improved the performance up to 20% from load sharing solutions. Performance graph shows as the number of tasks increases the performance of hybrid approach will also increases. In paper [8] Zhongkui Li and Zhisheng Duan they considers the distributed control problem for heterogeneous multi-agent systems with matching uncertainties and a leader whose control input might be nonzero and not available to any follower. Based on the relative states of neighboring agents, two distributed continuous controllers with, respectively, static and adaptive coupling gains, are designed, under which the tracking error of each follower is uniformly ultimately bounded, if the communication graph among the followers is undirected, the leader has directed paths to all followers, and the leader's control input is bounded. A sufficient condition for this existence of the distributed controllers is that each agent is stabilizable. Based on the relative states of neighboring agents, both distributed continuous static and adaptive controllers 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 [9] Jihno, presented a novel fault-tolerance mechanism to have the following advantageous features appropriate for large scale and dynamic hierarchical mobile agent-based monitoring organizations. In this paper [10] Tom Divoksi 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 [11] Rajwinder Singh presented mobile agent based fault prevention and detection technique where the team of mobile agents monitor each host in mobile agent based system. This research focuses on building an automatic, adaptive and predictive determining policy where critical host agents are identified in advance by monitoring agents, to avoid their failures. This paper presented a new approach to make mobile agent systems reliable. 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 the future, They intend to work out a more formal model of the quantity of dependence and incorporate other parameters to gauge the efficiency of the model in accurately measuring host vulnerability. In this paper [12], Asma Insaf Djebbar, Ghalem Belalem presented the mobile Ad hoc networks are distributed environments characterized by a high mobility and limited battery resources. In these networks, mobiles nodes are subject to many errors. In this paper, we present our approach of modeling by groups for faults tolerance based in MAS, which predicts a problem and provide decisions in relation to critical nodes. Their work contributes to the resolution of two points. First, they propose an algorithm for modeling by groups in wireless network Ad hoc. Secondly, they study the fault tolerance by prediction of disconnection and partition in network; therefore we provide an approach which distributes efficiently the information in the network by selecting some objects of the network to be duplicates of information.In this paper [15] Dr. Kapil Govil, presented an efficient solution to the dynamic allocation problem. Starting with the definition of thephase of a modular program, a model based on dynamic programming approach is suggested. Earlier the researchers advised that the dynamic allocation strategy is the best allocation technique as it facilitates the user to take decision for allocating the during run time. The suggested algorithm is implemented on the several sets of input data and it is recorded that algorithm is workable in all the case. Here they have considered the phases and each phase has the tasks are to be processed by the processors. In each phase only one task shall be executing on these processors. During the next phase an executing task may remain on the same processor for execution or may shift to another processor, in case of shifting the task to another processor, it added the reallocation cost. The impact of inter task communication cost is to be considered. Thus phase wise optimal costs are obtained. In this model, there are five phases and each phase has the equal numbers of tasks. Optimal allocation has been obtained along with phase wise optimal costs.

3. Fault Tolerance in Distributed System

Distributing computing is a computational system in which software and hardware infrastructure provides consistence, dependable and inexpensive to accesses high end computations. An imperfect system due to some reasons can cause some damages. A task which is working on real time distributed system should be achievable, dependable and scalable [6]. 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. By applying extra hardware like processors, resource, communication links hardware fault tolerance can be achieved. In software fault tolerance tasks,

to deal with faults messages are added into the system. Distributed computing is different from traditionally distributed system [7]. Fault Tolerance is important method in grid computing because grids are distributed geographically in this system under different geographically domains throughout the web wide. The most difficult task in grid computing is design of fault tolerant is to verify that all its reliability requirements are meet. A distributed computing system must be fault tolerant. It should be able to continue in its functioning in the presence of faults. Most of the faults are related to dependability.



Time Graph

Above graph shows old and new scenario times in which the nodes will complete the tasks in case of faults. The proposed methodology takes less time as compared to existing one which shows that proposed method is less time consuming.

4. Conclusion

In this paper, we conclude that distributed computing is a very important technique of computing system. There are different types of computing techniques like parallel, peer-to-peer and distributed computing. The main objective of the paper is to improve the fault tolerance of the distributed system so that it takes lesser time for recovery. Experimental results show that proposed method is far better than existing method as it takes lesser time to complete the task after node failure.

References

[1] George Coulouris, Jean Dollimore and Tim Kindberg , Distributed Systems : Concepts and Design, Addison-Wesley,Pearson Education 3rd Edition 2001.
[2] Asma Insaf Djebbar, Ghalem Belalem , “Modeling by groups for faults tolerance based on multi agent systems”, IEEE,2010
[3] Rajwinder Singh and Mayank Dave, Senior Member, “Antecedence Graph Approach to Checkpointing for Fault Tolerance in Mobile Agent Systems”, IEEE TRANSACTIONS ON COMPUTERS, VOL. 62, NO. 2, FEBRUARY 2013
[4] ANDREW S. TANENBAUM and ROBBERT VAN RENESSE, “Distributed Operating Systems”, 2006
[5] 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
[6] Shan Zhang and Jian-ping Wu, “Construction of Distributed and Heterogeneous Data Sharing Platform” , 2009 International Conference on Web Information Systems and Mining
[7] Vinod Kumar Yadav, Mahendra Pratap Yadav and Dharmendra Kumar Yadav , “Reliable Task Allocation in Heterogeneous Distributed System with Random Node Failure: Load Sharing Approach, International Conference of Computing Science, 2012
[8] 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
[9] Jinho Ahn, “Lightweight Fault-tolerance Mechanism for Distributed Mobile Agent-based Monitoring” IEEE, 2008
[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] 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
[12] Dr. Kapil Govil, “A Smart Algorithm for Dynamic Task Allocation for Distributed Processing Environment” International Journal of Computer Applications (0975 – 8887) Volume 28– No.2, August 2011
[13] E. N. Mootaz Elnozaby, L. Alvisi, Y. Wang and D. B. Johnson," A survey of rollback-recovery protocols in message-passing systems", ACM Computing Surveys, Vol. 34, NO. 3, pp. 375-408, 2002.