VIT®

Vellore Institute of Technology

(Deemed to be University under section 3 of UGC Act, 1956)

CSE 4001

Research Paper

Improving Fault Tolerance in Distributed Computing System

UNDER THE GUIDANCE OF

PROFESSOR KANNADASAN R

Slot: B2+TB2

PRESENTED BY

Shreyash Kawalkar	**16BCE0221**
Utkarsh Sharma	**16BCE0226**
Kusumakar Kashyap	**16BCE0351**

Improving Fault Tolerance in Distributed Computing System

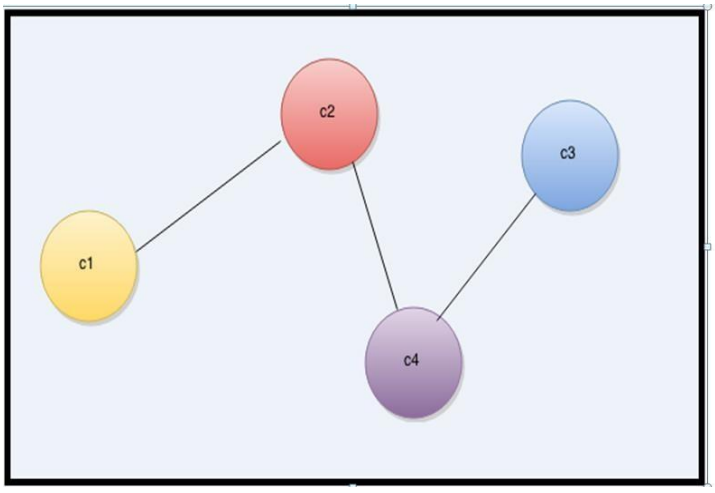
Abstract

A distributed computing is a software system in which components are located in different connected computers to communicate and organize their actions by transferring messages. There are some challenges in the distributed computer system. In this paper, we focus on error tolerance, which is responsible for system degradation. A new reliability-based technique is proposed to overcome the error tolerance problem and reassign the task.

Keywords: Reliability , DCS, execution time and Master Node Time

1. Introduction

The computing system is a system that is a combination of the number of computers and associated software that shares common memory. It can also be equal and multiple. A distributed system is a project in which several components are in different computers but function as one coherent computer system. A distributed system connected by local networks and physically interconnected [1]. The distributed calculation uses an network of many computers, each of which performs a part of a general task, to obtain a computational result much faster than with a single computer. The distributed computing system is of a heterogeneous nature. Because of its varied nature, different hardware and software of different kinds are required. The distributed system is better than the centralized system in several ways. The first point is that it does not have a central controller. Therefore, there is no possibility of system error if the central system is compressed. In the distributed system, computers are in distant places. They are connected with the number of servers. If there is a possibility of server errors, it also gets data from other servers. Another point is scalability. Due to the nature of scalability, multiple computers can be added at any time to increase endurance. Another point is redundancy. As we have already said, the combination of several smaller machines is not so expensive. It can be easily accessible and extensible. Distributed computing uses a network of many computers that perform a portion of the entire task. A distributed program is a computer program that runs on a distributed system. Distributed programming is the process of writing these kinds of languages [4]. Grid computing and cluster computing are types of distributed computer systems.



A distributed system has several independent computers connected to a shared network and a middleware that allows computers to organize their behavior and share ownership of the system so that users identify the system as a single integrated computer installation

1.1 Mobile Computing System: The mobile computing system is also one of the types of distributed computing system. In this network the nodes are called mobile nodes. It is an example of the DCS. In this type of system, mobile nodes have their own wireless interface [5]. They are connected to the interface via a wireless connection, even when they are mobile. The mobile computing system has different models in the field of networks. It is used in wireless LAN, MANET, WAN, MAN. In all fields, the wireless device must be able to interconnect the complete interface with each other. In the mobile computing system we need a node and a channel for communication.

1.2 Issues in Distributed Computing System: Issues in distributed system which are responsible to lowering down its rate are:

1. ***Flexibility:***
   The distributed system must be flexible so that users can easily make changes and improvements.
2. ***Scalability:*** The system must be designed in such a way that it is easy to cope in order to increase system growth. Central and central entity algorithms should be avoided. Most of the operations on the client workstation need to be performed.
3. ***Fault Tolerance:*** The system must be resistant to faults. In the future, if an error occurs, its performance is not reduced. Failure can occur due to mobility, overload, load unbalance and many other factors.
3. ***Security:*** In order for users to trust the system, the various resources of an information system must be protected from destruction and unauthorized access. The application of security in a distributed system is more difficult than a centralized system due to the lack of a single control point and the use of non-secure networks for data communication.

2. Literature Review

Various papers to improve fault tolerance of the system have carefully been studied. In this section we will discuss various techniques and algorithms to improve system performance.

On paper [1] they proposed a model in mobile computing that has two scenarios. They considered two cases, one case is when mobile hosts connect to the landline. The second case is when the mobile host does not connect to the host. It has a decision algorithm that decides when the node must connect to the fixed network and when the node must be disconnected.

On paper [2] they mentioned a brief categorization of errors, failures and errors found in a distributed environment.

On paper [3] they presented a technique to eliminate the problem that occurred due to the failure of the permanent node. Basically they tried to eliminate the overcoming of the complexity that occurs due to the mobility of the nodes. They proposed a shared load technique to maintain system performance.

On paper [4], they proposed an algorithm based on the control point technique. It is used to make the system error-free and improve performance based on antecedent graphics. They also proposed a future work in

which they mentioned the integration of a graphic and non-graphic scheme to achieve a high error tolerance system.

On paper [5] they proposed a solution for the dynamic assignment technique. In this they proposed a technique in which each process can perform the activity by process. Every time you move on to the next stage, the reassignment cost is added to the process. At this point you can be in the same processor or you can switch to another processor. By doing so, we will ultimately achieve an optimal cost at the optimal cost.

In document [6], they proposed a technique based on some conditions, for example that their entry should not be zero and is not available to all users. Based on the relative states of adjacent agents, both continuous and adaptive static distributed controllers were designed to ensure the uniform final limit of tracking error for each follower. A sufficient condition for the existence of these distributed controllers is that each agent is stable.

In this paper [7], they represented the Mobile Agent technology to improve flexibility and fulfill the promise of being a powerful agent and its mechanism. Mobile agent systems must also provide application customization with their additional agent security capabilities through dynamic distribution of application components on the malicious host and host security from a network. The architecture proposed in this document The prototype systems meet all the requirements to solve the above problems and can be used to provide a safe and reliable architecture, adapted to the characteristics of existing systems.

On paper [8] they presented a model for error-tolerant groups based on MAS, which provides a problem and provides decisions about critical nodes. Your work contributes to the resolution of two points. First, they propose an algorithm for group modeling in the ad hoc wireless network. Second, they study fault tolerance by providing disconnection and partition in the network; therefore, we provide an approach that efficiently distributes information across the network by selecting certain objects in the network as duplicates of information.

In document [9] the detection and fault prevention technique based on the mobile agent was presented in which the team of mobile agents monitors each host in the mobile agent based system. They proposed an approach to introduce fault tolerance in the multi-agent system, checking each time the points based on the update of the weights during the calculation of host dependency. From the experimental results, it can safely be inferred that the monitoring technique proposed for the distributed application of several agents can effectively increase the tolerance to system failures in addition to the effective recognition of vulnerabilities in the system.

3. Fault Tolerance in Distributed Computing Systems

Distributed systems in real time such as networks, robotics, nuclear air traffic control systems, etc. They are highly responsible in the term. Any errors in the real-time distributed system can cause a system to collapse if it is not detected and restored correctly at the moment. Fault tolerance is the important method that is often used to continue reliability in these systems. Distribution calculation is a calculation system in which the software and hardware infrastructure provides consistency, reliability and economy to access high-level calculations. An imperfect system due to some reasons can cause some damage. A task that works in a real-time distributed system must be possible, reliable and scalable [11]. By applying additional hardware such as processors, resources, communication links,

hardware fault tolerance can be achieved. In software fault tolerance activities, messages are added to the system to handle errors. Distributed computing is different from the traditionally distributed system. Fault tolerance is an important method in grid computing because the grids are geographically distributed in this system in different geographic domains throughout the web. The most difficult task in calculating the network is that the design of fault tolerance is to verify that all the reliability requirements are met [12].

3.1 Techniques for Fault Tolerance: It is better to consider that system that is devoid of all the defects and has impeccable performances. So there are several techniques that premeditated to make the system tolerant to failures. These techniques are:

3.1.1 Hardware Resilience: Hardware resistance is the technique related to hardware transparency. In this technique, the transparency of the unit's hardware is used to manage network reliability. It is the technique of correction of memory errors [6].

3.1.2 Application based Resilience: It treats faults using information about the application that uses and makes the developer smarter.

3.1.3 Intercrosses Communication: In the distributed system to share the information process must communicate with each other. This is why we need synchronization between all the processes. But it must be controlled so that the process can not judge the speed of another process. All scenario communications are done through the message transmission scheme. No need for shared memory for passing messages [8].

a) Synchronization Message Passing: This process is without buffer. The send command is waiting until the receive command has been canceled or executed. It has a synchronization point where both the sender and the receiver have synchronized them with that point. The statements are made in this process.

b) A synchronization Message Passing: Communication and synchronization both processes are taken separately by this process [9].

3.1.4 Replications:

Replication means making multiple copies of similar data on the servers. For any action to succeed, replication is required. Suppose if a node fails and there is no replication of that data, this will affect system performance. But there are replicas of data from which the user can get data from other servers in case there is also an error in the node. It is a proxy-based monitoring technique that is applied in distributed systems. It has two strategies, active or passive. Helps to eliminate general problems and complexities. It is a well-known technique that is used to improve availability. But repetitions can produce some serious problems such as inconsistencies [3].

Replicas help improve system performance. Using numerous protocols, users receive updated data. It also has some limitations as it is expensive. It also increases system availability due to multiple copies. But it requires updating data every time. It also helps to make the system tolerant to failures [5].

3.1.5 : Load Balancing Algorithm:

The load balancing algorithm is based on reassigning processes during execution time between processors. The main goal of the system is to improve system performance. This can be done by assigning the asset to the light weighted activity of the weighted heavy activity. Runtime overload is the drawback of the dynamic load balancing schemes due to the transfer of load information between the processors, to the process of making decisions for the selection of processes and processors for job transfers and to communication delays due to transfer of the task itself [9].

***3.1.6 : Check pointing*:** Basically, this technique is used to restore the process up to a certain point after the error. Fault tolerance can be achieved through various types of redundancy. The beginning of the checkpoint is the common method. In this method, an application starts from the previous control point after an error. The application may not be able to meet strict time objectives.

4. Proposed Methodology

The number of users of distributed networks and systems increases considerably with the increasing complexity of their services and policies. System administrators try to guarantee a high quality of the services that each user needs by maximizing the utilization of system resources. To achieve this objective, the correct and efficient real-time management and monitoring mechanisms are essential for the systems. But, as systems infrastructures expand rapidly, a large amount of monitoring information is produced by a greater number of managed nodes and resources, so the complexity of the network monitoring function becomes extremely high. Therefore, monitoring mechanisms based on mobile agents have been actively developed to monitor these distributed and dynamic large-scale network systems in an adaptable and efficient manner. The proposed algorithm is to assign tasks to other nodes only when the master node moves from its original position. The main problem in this architecture is task scheduling, if a slave node fails, the task assigned by the master node will not complete and an error occurred. In this work, we will work on a technique that helps reduce system fault tolerance and increase system performance.

In this work, a new reliability formula is added that will calculate the reliability of each node responsible for executing the task. All available nodes have a reliability value of 1 in the initial phase of the project. The formula that is based on the maximum failure rate and the maximum execution time is applied. Each node in the network has its own failure rate and execution time. Based on these two and the amount of tasks that will be executed, the reliability value is calculated. The node that has the maximum reliability value will execute the task on the node that changed its position. The formula for calculating reliability is given below.

1. AB=Maximum execution time+ Maximum Failure rate
2. BC=execution time of each node + failure rate of each node
3. DE=BC*number of task for execution
4. Reliability= AB-DE

5. Conclusion

The distributed calculation system is of a heterogeneous nature. Because of its varied nature, different hardware and software of different kinds are required. Mobile computing is one of the types of distributed

computing. There are several problems of distributed computing, such as scalability, availability and fault tolerance. In this paper various fault tolerance techniques were studied in distributed computer systems. Each technique has its pros and cons. In this article a new technique has been proposed to improve the accuracy of the system. In case of node mobility and node failure, it can be used to improve performance and make the system tolerant to failures.

References

[1] Take Dimovski, Pece Mitrevski, "fault model tolerance in connection for distributed processing mobile computing environment transactions" ITI 2011 33rd Int. Conf. On information technology interfaces, June 27 and 30, 2011, Cavtat , Croatia

[2] Sajjad Haider, Riaz Naveed Ansari, Muhammad Akbar, Mohammad Raza Perwez, Khawaja MoyeezUllah Ghori1, "fault tolerance in distributed paradigms", International Conference on Computers, Communication and Management 2011

[3] Vinod Kumar Yadav, Mahendra Pratap Yadav and Dharmendra Kumar Yadav, "Reliable Assignment Activities in Heterogeneous Distributed Systems with Random Node Failure: Load Sharing Approach, International Conference on Computer Science 2012

[4] Rajwinder Singh, Mayank Dave, "Using critical host characteristics for fault tolerance in mobile agent systems, 2nd IEEE international conference on parallel, distributed and grid computing, 2012

[5] Dr. Kapil Govil, "An Intelligent Algorithm for Dynamic Assignment of Tasks to a Distributed Processing Environment" International Journal of Computer Applications (0975 - 8887) Volume 28 - No.2, August 2011

[6] Zhongkui Zhisheng and Li Duan, "Monitoring monitoring distributed heterogeneous multi-agent systems with uncertainties", 10th IEEE International Conference on Control and Automation (ICCA) Hangzhou, China, 12-14 June 2013

[7] Sreedevi RN, UN Geeta, UPKulkarni, ARYardi, "Improving mobile applications and security agents fault tolerant capacity, International Conference on Advanced Computing IEEE 2009 (IACC 2009) Patiala, India, March 6-7 2009