Dynamic Scheduling Of Task with Dynamic Load Balancing For Effective Resource Utilization in Distributed System

Kishorkumar Madhad¹ Arjun Sojitra²
¹,²Atmiya Institute of Technology and Science, Rajkot, Gujarat
¹,²Gujarat Technological University, India

Abstract—The dynamic load balancing was studied and also distributed system. Our work based on process scheduling and works load balancing that we proposed. Here ELBS (ERP Load Balancing Scheme) studied and create some new step that can able to balance resource and maintenance time. It is suitable for large application with number of network nodes, the ERP system relative to the system without ant load balancing strategy.

Keywords: Load Balancing, Distributed System, Process Scheduling, ERP System, Socket, RMI

I. INTRODUCTION

In current era, situation has become changed and all company and institutes are generally used Internet for anything. Because it is easy to contact anyone from any place through network and sharing the data and also it is inexpensive as compared to others. To connect with each other, some Distributed systems are available that is the network generated like WWW, Internet, Intranet. A distributed system is piece of software that ensures that a collection of independent computer appears to its user as single and easy to explain.

The word distributed in terms such as "distributed system", "distributed programming" and "distributed algorithm", originally referred to computer networks where individual computers were physically distributed within some geographical area. The terms are currently used in a much wider sense, even referring to autonomous processes that run on the same physical computer and interact with each other by message passing. While there is no single definition of a distributed system, the following defining properties are commonly used:

1) There are several autonomous computational entities, each of which has its own local memory.
2) The entities communicate with each other by message passing.

A distributed system may have a common goal, such as solving a large computational problem. Alternatively, each computer may have its own user with individual needs, and the purpose of the distributed system is to coordinate the use of shared resources or provide communication services to the users.

Other typical properties of distributed systems include the following:

- The system has to tolerate failures in individual computers.
- The structure of the system (network topology, network latency, number of computers) is not known in advance, the system may consist of different kinds of computers and network links, and the system may change during the execution of a distributed program.
- Load balancing is the allocation of the workload among a set of co-operating computational elements and also we can say that all processors working all the time and distribute the load (work) to meet the goal. There are two types of Static and Dynamic.

In Static, all information available before starting and In Dynamic, there no information provide before starting. So dynamic is very efficiency.

We can also use of Socket which is one types of communicator, a Socket is one of endpoint of way to communicate link between two programmer running on the network.

Here we also discuss about CORBA which full form is Common Object Request Broker Architecture and it provide effective and convenient tools for utilizing in Distributed System. It also allows application implement in differing language the ability of communication while RMI is java feature.

And in last we briefly discuss about RMI is Remote Method Invocation. It use in JAVA programming language and write in object oriented programming on different computers are communicates in distributed network. JAVA RMI is a higher form of communication where the messaging is handled by this invocation.

II. PROBLEM STATEMENT FOR DISSERTATION

Our work shall be focused on the distributed system in which we will focus on dynamic load balancing. Different person may have different system some have no need to load the balance but higher system and major working like Server, DBMS,IBM system and like that types of major company servers are need to distributed their load in other system with uses of effective resources with load balancing.

In Ref.[1] paper that worked on ERP load balancing and algorithm is ELBS(ERP Load Balancing Scheme) based on processing scheduling and load balancing. Their algorithm avoids the high complexity of central algorithms and single point failure problems. But there are some future work remaining, like to reduce a process migration cost and no time is waste and all process are done time to time.

III. OBJECTIVE OF DISSERTATION

This thesis is focused on the study of the distributed system that involved load balancing systems composed system executing time and make the faster. There are many different types of distributed computing systems and many challenges to overcome in successfully designing one. The main goal of a distributed computing system is to connect users and resources in a transparent, open, and
scalable way. Ideally this arrangement is drastically more fault tolerant and more powerful than many combinations of stand-alone computer systems.

The main objective of the thesis work has been to schedule a task in order to achieve the timing constraints and dividing the load in this distributed system and the goal of load balancing is improving the performance by balancing the loads among computers.

IV. RELATED WORK

Here one algorithm used called as ELBS (ERP Load Balancing Scheme). Bottleneck nodes in different systems are compared between the mode without any load balancing strategy which is called normal and the mode of ELBS. First of all, a certain number of nodes are generated on the two-dimensional plane. \( R_p(v) \) which depicts the number of the parallel processes and \( Q_p(v) \) which depicts the number of the serial processes are also produced for each node. The value ranges of \( R_p(v) \) and \( Q_p(v) \) are designated between 10 and 20 and between 30 and 50 respectively. At the same time, \( R_{\text{max}}(v) \) and \( Q_{\text{max}}(v) \) are generated at the very beginning in each node. The value ranges of the two parameters are between 30 and 40 and between 50 and 100 respectively.

Secondly, \( R_d(v) \) which is between -4 and 4 and \( Q_d(v) \) which is between -20 and 20 are generated periodically. It can make sure that \( R_d(v) \) and \( Q_d(v) \) in each node will be updated in the experiments. In below algorithm, is generated 20 times in every scale of nodes number. And the different proportions of the bottleneck nodes are compared between normal mode and ELBS.

![Proportion of bottleneck nodes vs. Number of nodes](image)

Fig. 1: Proportion of bottleneck nodes vs. Number of nodes [1].

V. PROPOSED WORK

After reviewing whole status, we can provide that change and modified the above algorithm as per following equation. The algorithmic style can be reviewed as under

For process migration 2 cases

1) For long running jobs, when waiting queue goes up:-

\[ \text{FW(bl)} = \text{featured container of resources (the factor which will find the available resources for node)} \]

\[ \text{OR(bl)} = \text{Occupied resources} \]

\[ \text{Smallest (FW (bl)) will be assigned} \]

Where FW is work load, bl = balance

2) For overloaded node

Workload factor – best match first assign (BMFA) \[ \text{WF (bl)} = \text{In central node the best match , on basis of load calculation will be assigned first.} \]

A. For process migration and switching purpose

Maximum Idle Utilization Technique (MIUT) can be used, it can be a improvement over ELBA algorithm. Architecture of algorithm has been defined as below, we have considered 2 cases:

B. For long running jobs, when waiting queue goes up:

We have already defined variable to calculate maximum value of pending queue on basis of X threshold value.

\[ \text{WF(bl)} = \text{future container of resources (the factor which will find the available resources for node to execute a processes).This variable will be helpful to utilize the most efficient resources. Therefore other effective tasks can be switched over the most idle resources.} \]

\[ \text{OR(bl)} = \text{occupied resources. This variable will find out the occupied resources by the node.} \]

Smallest (FW(bl)) will be assigned on basis of BMFA (Best Match First Assign) technique.

C. For overloaded node

When maximum utilization of resources over a node occurs such a situation is called overload or bottleneck.

Workload factor –

\[ \text{WF(bl)} = \text{This factor will be helpful to determine workload of nodes. Best Match First Assign (BMFA) will be chosen.} \]

\[ \text{AF(bl)} = \text{This factor will check the pulse of each node to ensure availability.} \]

We are also planning to include bandwidth factor for migrating process which is needful for lengthy tasks.

Independent threads will maintain all information about nodes which will supposed to gather details at every n seconds without interfering the original tasks.

1) Step to be used:

There are some steps which should be modified to this system which is shown below:

Step 1: WHILE ( \( v \) is overloaded || \( v \) has long running job || \( k < L) \)

Step 2: \( \text{FW (bl)} = \text{future container of resources (the factor which will find the available resources for node)} \)

Step 3: \( \text{OR(bl)} = \text{occupied resources} \)

VI. CONCLUSIONS

From the above all analysis we conclude that proposed distributed dynamic scheme algorithm will be useful in many areas. It may be useful for colleges, major industries etc. the application system which needs more efficiency and time. That situation can be occurred through such a background mechanism which will distribute the load across multiple network connected nodes that may be physically separated by locations. Normally scientific organization, ERP application etc. use such a concept to
overcome performance issues with the application system. Such a concept called active – active cluster where load is distributed over multiple nodes as per internal algorithm. In this scenario we can utilize an idle in terms of the node with minimum capacity for maximum utilization. With this mechanism we can procure the most idle and efficient node for future use. Also we have tried to utilize this methodology for heavy loaded nodes or for long running jobs. The system will be extended for large purpose of system. It can be enhanced to a lot of greater extent by improving the algorithm which can include geographic locations, enhanced path system by load balancing of each node. Here also we can found any sequence of number in distributed system by our Maximum idle utilization technique by using RMI.

REFERENCES