AN EFFICIENT LOAD BALANCING APPROACH IN CLOUD SERVER USING ANT COLONY OPTIMIZATION

Shanmuga Priya.J¹, Sridevi.A²

¹PG Scholar, Department of Information Technology, J.J College of Engineering and Technology
²Assistant professor, Department of Information Technology, J.J college of Engineering and Technology

ABSTRACT: Today cloud computing is a growing area in research and it also comprises virtualization, distributed computing, networking, software and web services. A cloud entails of numerous components such as clients, data centers and distributed servers, internet and it consist of fault tolerance, high availability, effectiveness, scalability, flexibility, reduced overhead for users, reduced cost of ownership, on demand services. One of the impact of cloud computing is load balancing and it also degrades its performance. In this paper to introduced an effective load balancing algorithm using Ant colony optimization technique and compare the system availability and efficiency with game theory algorithm.

KEYWORDS: cloud computing, load balancing, ant colony optimization, game theory

I. INTRODUCTION

Cloud computing is an interesting technology in the research field. In Gartner’s report [2], it utters that every changes in the IT industry are caused by cloud. It provides new types of services to users. Cloud computing is a term generally defined as with the help of internet we can access cloud at anytime, anywhere and anyplace. Different types of cloud are present like private, public and hybrid [3]. A private cloud can give access to single organization. A public cloud can give access to all; through internet anyone can access public cloud. A hybrid cloud is shared by multiple organizations.

The cloud computing is a model for allowing convenient, flexible, reliable on-demand network access to a shared pool of configurable computing resources, that can be quickly provisioned and unrestricted with minimal management effort, which helps availability and is collected of important characteristics like on-demand self-service, broad network access, resource pooling, rapid elasticity, measured services and efficiency and scalability; multi layered service abstraction like software as a service (SaaS), Platform as a service (PaaS) and infrastructure as a service (IaaS) [3].

The rest of the paper is started with related work in section II, followed by load balancing in section III. Proposed work discussed in section IV and results are observed in section V. Finally conclusion is discussed in section VI.

II. RELATED WORK

In [4] the authors propose to find the best efficient load balancing by using fuzzy logic. The authors designed the algorithm using load balancer fuzzy based on round robin approach. It minimized processing time and improves overall response time.
In [5] the authors perform the comparative study of three load balancing approaches like honeybee foraging behavior achieve good load balancing through local server but throughput is not increased. Biased random sampling achieves load balancing across all system nodes using random sampling but degrades population diversity. Active clustering optimizes job assignment by local re-wiring but degrades system diversity.

In [6] the authors proposed the Scheduling strategy on virtual machine load balancing based on genetic algorithm. It resolves the problem of load imbalancing and high migration cost but this method increases the computational cost and also dynamic changes in virtual machine.

In [7] the authors proposed the algorithm which combines the techniques of ant colony system and max-min ant system. This algorithm focuses on local pheromone trail update and the trail limit value and also minimizes the completion time of each job.

III. LOAD BALANCING

Load balancing is a process of distributing the loads between the collective system to improve the utilization of resources and response time of the job with increase throughput in the system. Load balancing are used to performance of system, comparison of load, interaction between loads, stability of different system, work to be transferred, estimation of load, selecting of nodes. The load may be memory capacity, network load, cpu load or delay.

There are several load balancing algorithm are available, which can be classified mainly as static and dynamic. Static algorithm needs prior knowledge of the system. In this algorithm the servers divide the traffic equivalently as round robin. The traffic on the servers will be disdained easily and consequently it will make the situation more imperfectly. Another version of load in load balancing is weighted round robin which is used to improve the critical challenges associated with round robin. In this all weights are equal, the servers will become balanced [8].

Dynamic algorithm depends only on current state of the system and it does not consider the pervious states. It takes decision based on current state of the system without need of prior knowledge. Dynamic algorithm selected a lightest server from whole network to balance the traffic. However, selecting a particular sever needed real time communication with the networks, which will leads to extra traffic added on system[8].

IV. PROPOSED WORK

ANT COLONY OPTIMIZATION:

Marco Dorigo in 1992 proposed the new algorithm that is called as Ant system(AS) in his Ph.D. thesis. Ants are small insert and it is capable of finding the shortest path. Ants work together to search their own food and go back to their nest. Ants have blind eyes how its able to follow the fellow ants means, while moving from one place to another place ants leave a chemical called pheromone. By following the pheromone the other ants came to the food sources. After some amount of time the pheromone evaporate quickly. If more number of ants travel in the same path, the pheromone intensity increases. The intensity of pheromone can vary on two factors.

1. Quality of food
2. Distance of food.
The ants select the next node based on the pheromone. If any obstacles in their path ants use the modify path to reach their food source. Figure 1, gives a brief idea about this scenario. The movement of these ants is updating a solution set. The traversal of ants is generally two types.

1. Forward movement

2. Backward movement

![Figure 1 modify path upon encountering obstacles](image)

In cloud computing the food implies different server. The path with highest pheromone is mostly chooses by other ants because of its shortest path. The shorter path will reach the food quickly and back to nest by the same path. More amount of pheromone are present in shorter path only so all ants are chooses that path only.

**SYSTEM MODEL:**

**A. USER MODULE:**

In cloud computing all work are done with the help of internet. So that unauthorized person can easily access the file without the knowledge of the own. For this reason users are having authentication and security to access their detail which is present in the ontology system. Before accessing the file in the system users are having the accounts in that cloud environment. If the user is new to access the file in the cloud system, the user first register in that environment after that access the file.

**B. ADD SERVER AND LOCATION:**

In cloud environment a range of technologies can be applied to location determination. The actual method applied is similar to the wire map method, if a radio transmitter can be identified, then the location of the device can be given as the position of the transmitter with a region of uncertainty. This module is designed to set the server for one particular region. Then details of the particular location are stored into the main server. So everyone can retrieve the information from main server. This increase searching speed, from this user can view and download even rural details.

**C. MAIN CONTROLLER AND BALANCER:**
The cloud is divided into number of partition. After creating the partition one part is assigned as controller and other part are assigned as balancer. When jobs are arrives the controller deciding which partition should receive the job first. After receiving jobs, the balancer decides how to assign jobs to the nodes.

**D. ASSIGNING JOB TO THE PARTITION:**

When jobs arrives first to select the correct balancer. The balancer has three type of status

1) idle- The status to become idle. When the amount of idle nodes exceeds X.

2) normal- The status to become normal. When the amount of normal nodes exceeds Y.

3) overload- The status to become overload. When the amount of overload nodes exceeds z.

The parameter X, Y, and Z are assigned by the balancer.

**Best partition searching**

Begin

While job do

SearchBestPartition(job);

If partitionstate==idle!!partitionState== normal

then

Send job to partition;

Else

Search for another partition;

End if

End while

End

**WORKING OF IDLE STATUS:**

When the balancer status is idle many jobs are arriving. In this condition, the balancer are able to process the job quickly. Many load balancing algorithm are available. one of them is round robin algorithm. It is one of the simplest algorithms, which passes new request to next server in the queue. This algorithm gives equal opportunity to all nodes.

Before performing round robin steps, the balancing tables are ordered based on load degree. The system construct a circular queue and travel again and again so all jobs are assigned. First the lowest load degree nodes only have the jobs.

**WORKING OF NORMAL STATUS:**

When the balancer status is normal, more numbers of jobs are arrives much faster than in the idle status. Load balancing use different strategy to reduce the complex. All users want to complete their jobs in short duration. To reduce the response time here used game theory. Game theory has two types of games:
1) Non-cooperative games

2) Cooperative games

In non-cooperative games the decision are taken only for its own benefits which is taken by decision makers. In cooperative games the decision are taken for all benefits. The decision makers compare the nodes to each other after that only they take decision. Normal status are working based on the non-cooperative games.

E. ASSIGNING JOB TO NODE IN CLOUD:

Every node in the partition should send the load information to the balancer. Balancer to evaluate the status and define the load degree of each node.

The load degree is calculated from these parameters as mention below:

Step 1: Define a load parameter set: \( X = \{X_1, X_2, \ldots, X_3\} \).

Step 2: calculate the load degree as:

\[
\text{Load\_degree}(T) = \sum_{n=1}^{m} aX
\]

Step 3: To calculate average load degree

\[
\text{Load\_degree(avg)} = \sum_{n=1}^{i} \frac{\text{Load\_degree}(T)}{i}
\]

Step 4: Define node status level:

Idle

\( \text{Load\_degree}(T) = 0 \)

No job is processed.

Normal

\( 0 < \text{Load\_degree}(T) \leq \text{Load\_degree(high)} \)

Jobs are processed when it in normal.

Overload

\( \text{Load\_degree(high)} \leq \text{Load\_degree}(T) \)

When it in overload status no jobs are received.
F. ANT COLONY ALGORITHM:

In this module, the ants imply as jobs and food implies as servers. The ants first select the server, if the server is available the ants perform in that server and evaluate the best result as shown in figure 2. If the server is not available to do the jobs the ants choose the next server by using the transition rule. The server already have different amounts of load. The load increases or decreases according to the jobs. By using the number of ants the response time will slightly increase.

![Figure 2 Ant Colony Optimization Flow Chart](image)

The proposed algorithm is as follows:

1. Select the server.
2. Schedule the job to the server.
3. While the job is not scheduled to the server, repeat steps 3 and 4.
4. If the server is available, then acquire the job. Else go to step 3.
5. Finish the job.
6. Return.
7. End.

G. COMPARISON:

To evaluate the result of ant colony optimization and game theory were simulated in Windows 7 (32-bit), Intel i3 processor, 330 m processor, 2.40 GHz of speed with memory of 4 GB and Java are used as language. More number of jobs is scheduled with different servers. The result shows that the response time of ant colony optimization is minimized as shown in figure 3 compared to the game theory.
V. CONCLUSION:
In cloud computing load balancing is one of the impact. It distribute the load equally to all nodes in the system to achieve user satisfaction and utilization of the resources. In this paper to proposed an efficient load balancing algorithm like ant colony optimization have been discussed to reduce the response time. Similarly ant colony optimization can achieve better load balancing and provide good response time than game theory.

VII. FUTURE WORK:
Except ant colony optimization, there are many other load balancing algorithm are available. Other algorithm may provide the better result so test and compare the efficiency and availability.

REFERENCES
[1] GaochaoXu, Junjie Pang, and Xiaodong Fu, A Load Balancing Model Based on Cloud Partitioning for the Public Cloud, tsinghua science and technology, ISSNl1007-0214I 04/12l lpp34-39 Volume 18, Number 1, February 2013.

