International Journal of Emerging Trends & Technology in Computer Science (IJETTCS)
Web Site: www.ijettcs.org Email: editor@ijettcs.org
Volume 5, Issue 6, November - December 2016 **ISSN 2278-6856**

Distributed Load Balancing in Cloud using Honey Bee Optimization

S.Jyothsna

Asst.Professor,IT Department Department CVR College of Engineering

Abstract

Load Balancing is a method to distribute workload across one or more servers, network interfaces, hard drives, or other computing resources. Typical data center implementations rely on large, powerful (and expensive) computing hardware and network infrastructure, which are subject to the usual risks associated with any physical device, including hardware failure, power and/or network interruptions, and resource limitations in times of high demand. Load balancing in the cloud differs from classical thinking on load balancing architecture and implementation by using commodity servers to perform the load balancing. This provides for new opportunities and economies of scale, as well as presenting its own unique set of challenges. As the scheduling problem is a NP hard problem, using heuristic optimization algorithms is the feasible solution for efficient load balancing in cloud.
This paper presents the application of honey bee optimization for load balancing in cloud computing and compares with genetic and ACO algorithms. comparatively this is the most suitable algorithm as the cloud is dynamic nature.

Keywords: cloud computing, load balacing, Honey bee ,Genetic Algorithm,ACO,virtual machine etc..

1. INTRODUCTION

1.1Cloud computing

A cloud is a collection of physical machines and virtual machines connected together to provide services to the users on pay-per -service model. Cloud architecture consists of Data centers which are the key resources to provide services through virtual machines connected to the end user. data centers provide services through virtualized cloud platforms. Cloud computing is providing majorly three types of services.

1. IaaS(Infrastructure as a Service)
2. PasS(Platform as a Service)\
3. Saas(Sofware as a Service)

The cloud service provider (CSP) provides on demand provisioning of Hardware like processing power, I/O, large amounts of storage etc.By utilizing virtualization, each user accesses the services of cloud through a virtual machine, where number of virtual machines shares a single physical server. cloud computing provides a service oriented platform for cloud users.

1.2 Load balancing in Cloud

As the cloud is dynamic , where number of tasks are to be scheduled dynamically by the available virtual machines The conventional algorithms like FIFO , Round Robin may not distribute the load efficiently. The load balancing algorithm must be dynamic I.e the load on each resource must be updated periodically and check the availability of the resources whenever there is a need to allocate a resource to the task. Cloud is heterogeneous in nature, servers of different capacity may be available at data center. The following challenges are need to be considered for efficient load balancing in cloud.

Challenges in Load Balancing

1. Overhead Associated: Overhead due to movement of tasks, inter process communication, overhead should be reduced so that load balancing algorithm works well.

2. Throughput : It is the number of tasks executed in fixed interval of time.

3.Performance : Performance can be defined as efficiency of the system,and it must be improved

4.Resource Utilization: This is used to test the utilization of resources. This should be maximumfor an efficient load balancing algorithm

5.Scalability:the quality of service should be same if the number of users increases. The more number of nodes can be added without affecting the service.

6.Response time: can be defined as the amount of time taken to react by a load balancing algorithm in a distributed system. For better performance, This parameter should be reduced.

7.Fault tolerance: In spite of the node failure, the ability of a systemto perform uniform load
balancing.

Load balancing can be categorized as

1.centralized load balancing

2. distributed load balancing

In centralized load balancing the balancing is done by a centralized server(single machine).the complete analysis of the overall system periodically should take care by the centralized server itself. And all other machines connected to it must inform to the availability of the resources at their locations and the balancing of the load must be done by the central server only. This is a challenging task in cloud as the cloud is an enormous collection of resources so controlling in such a huge collection and dynamic updation may not be feasible and also the risk is high when there is a failure of the central server.

The distributed load balancing is the alternative approach where the load balancing is the responsibility of each server connected to the cloud.

Advantages of Load balancing

1. scalable performance
2. secure application delivery
3. stress free deployment

Every node in the network maintains local knowledge base to ensure efficient distribution of tasks in static environment and re-distribution in dynamic environment. In distributed scenario, failure intensity of a node is not neglected. Hence, the system is fault tolerant and balanced as well as no single node is overloaded to make load balancing decision.

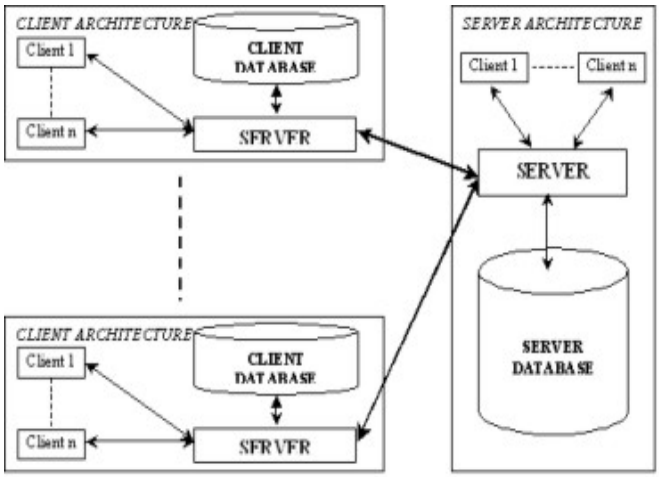
1.3 Distributed system architecture

Distributed computing is a field of computer science that studies distributed systems. A distributed system is a software system in which components located on networked computers communicate and coordinate their actions by passing messages. The components interact with each other in order to achieve a common goal. Three significant characteristics of distributed systems are: concurrency of components, lack of a goal clock and independent failure of components.

The following figure shows the distributed system architecture where n number of clients are connected to the server

2. RELATED WORK

The load balancing can be formulated as optimization problem where the optimum resource is selected locally and assigned the corresponding task so that the overall performance of the system must be increased.



2.1 Genetic Algorithm

Genetic algorithms are commonly used to generate high-quality solutions to optimization and search problems by relying on bio-inspired operators such as mutation, crossover and selection.

In a genetic algorithm, a population of candidate solutions (called individuals, creatures, or phenotypes) to an optimization problem is evolved toward better solutions. Each candidate solution has a set of properties (its chromosomes or genotype) which can be mutated and altered; traditionally, solutions are represented in binary as strings of 0s and 1s, but other encodings are also possible.

The following steps need to be followed in genetic algorithm

1. Initialization
2.fitness functionality
3.selection
4.crossover
5.mutation

Limitations of Genetic Algorithm in cloud environment

The genetic algorithm works well in dynamic environment with centralized load balancing but it is not suitable for dynamicenvironments like cloud where the load varies from time to time as the various users are connected to the cloud and demand for different types of services and every service need to be serviced by minimizing time with optimum number of resources , so the centralized server may not balance the load efficiently ,especially in huge computing such as cloud the distributed load balancing will give optimum results .

2.2 Ant colony optimization

The basic idea of ACO is to simulate the foraging behavior of ant colonies. When an ants groups try to search for the food, they use a special kind of chemical to communicate with each other. That chemical is referred to as pheromone. Initially ants start search their foods randomly. Once the ants find a path to food source, they leave pheromone on the path. An ant can follow the trails of the other ants to the food source by sensing pheromone on the ground,. As this process continues, most of the ants attract to choose the shortest path as there have been a huge amount of pheromones accumulated on this path. This collective pheromone depositing and pheromone following behavior of ants becomes the inspiring source of ACO.

ACO Algorithm approach

Initialization of algorithm : All pheromone values and parameters are initialized at the beginning of the algorithm.

Initialization of ants: M number of ants are initialised to select N number of task. Each ant build solution to M number of resources. In each iteration ants are randomly selected to build a constructive direction .

Local Pheromone Updating: After M ants map solution to M number of resources ,pheromone value is updated by local pheromone updating rule. The local pheromone update is performed by all the ants after each construction step. Each ant applies it only to the last edge traversed.

Limitations of ACO

1. Recruitment Strategies (methods used to communicate previous search experiences to other members of the colony) are indirect
2. Exploration may not be sufficient
3. It is more adaptive in nature.
4. Its convergence is guaranteed but time to convergence is uncertain.
5. Coding is not straightforward.
6. It is prone to falling in the local optimal solution.

8 PROPOSED WORK

8.1 Honey bee optimization algorithm

The idea behind the BCO(Bee Colony Optimization) is to create the multi agent system (colony of artificial bees) capable to successfully solve difficult combinatorial Optimization problems.

A colony of honey bee can extend itself over long distances as to find many food sources such as flower patches and then these bees harvests nectar or pollen from these sources. A small fraction of the colony finds the environment looking for new flower patches. When food source is encountered the scout bees go in the field surrounding the hive and check for quality beneficial. When they return to the hive, the scouts collect the food harvested. There is an area in the hive called as the “dance floor”, where waggle dance is performed by the bees that found a very beneficial food. Through the waggle dance a scout bee passes the position of its search to idle spectator, which helps in the using of the flower patch. Here the duration of the dance is according to the scout’s rating of the food source, to harvest the best rated flower patches more foragers get recruited. When dance is done, the scout return to the food source it found to see more food. Till the food is profitable, food sources will be posted by the scouts when they return to their hive. Foragers who are recruited recently may waggle dance as well, which will step-up the recruitment for highly profitable flower patches. This auto catalytic process will go on to find most beneficial flower patches .

The Honey Bee behavior inspired load balancing algorithm was proposed, which aims to achieve well balanced load across VMs to maximize the throughput and to balance the priorities of tasks on the VMs. Hence, the amount of waiting time of the tasks in the queue is minimal. Using this algorithm average execution time and reduction in waiting time of tasks on queue were improved. This algorithm works for heterogeneous type of systems and for balancing non preemptive independent tasks.

Algorithmic Approach

1. Cloud Setup : The cloud is set up by using data centers , physical machines and virtual machines

2. Initialization: All virtual machine loads are initialized to zero , The jobs need to be assigned are send as cloudlets.The threshold value is chosen based on the virtual machine capacity .

3. LoadBalancing: load balancing algorithm is described using following flow chart

The algorithm checks periodically load on each virtual machine and migration or tansfer of load takes place according to the following approach. This algorithm identifies the highly loaded VM and finds the less loaded VM then transferring followed, so that the loads are distributed evenly, the response time minimizes and resource utilization increases.

The task can be considered as a bee and it is searching for a less loaded VM(food source),when it finds the suitable VM assignment of task to VM takes place, and the next task also tries to assign to the same VM ,this assignment continues until the load on the VM reaches threshold

value. Once the threshold value is reached , the search starts to find another less loaded VM and the task is redirected to that VM.

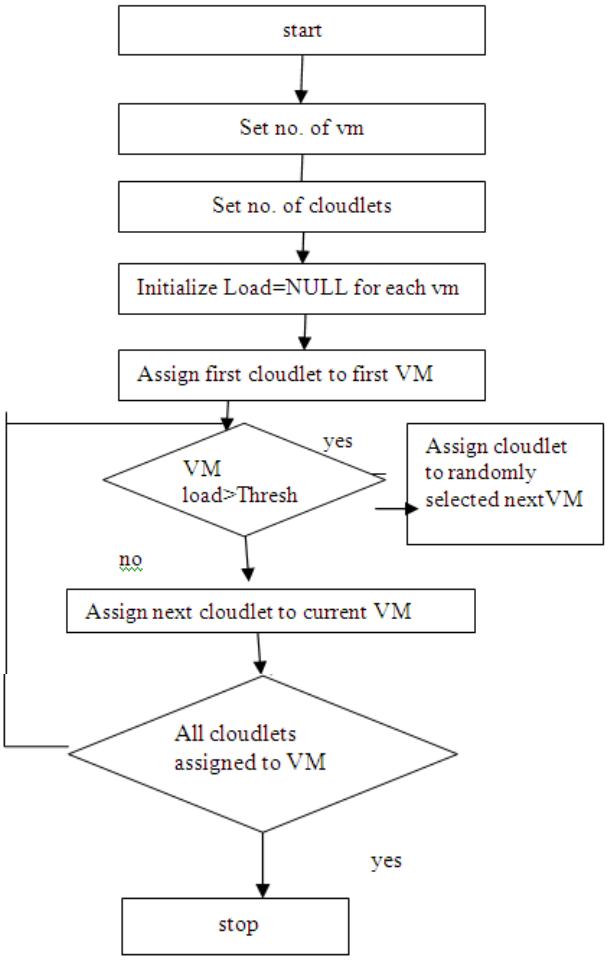


Figure 1 Honey bee algorithm flowchart

9 IMPLEMENTATION

In the dynamic load balancing algorithms, there is continuous variation in the workload. So there were some decision making algorithms are required. In this decision making system, there were firstly central decision maker, so no other node is decision maker except central node in view of that that if the central node becomes fail, subsequently thetotal system fails and hence the reliability becomes less. Secondly organization decision maker in which the total system should be not speaking in to groups hence that the communication cost becomes shortened. But taking decision without considered the mass system load therefore that global optimization explored a major suffering. The advantage of proposed algorithm is efficient load addition mechanism, excited distributed decision maker, migration selection model and full of zip file migration algorithm for a bigger load balancing. The disadvantage is degradation of the amass system due to the migration side effect.

```
HoneyBee()
  {
       set no_of_vm=2;
       set  no_of_cloudlets=4;
            set threshold=5;
     for each task in task queue
       init  walklength = 0;
       while (task is assigned to a vm) or (walklength
> threshold)
            {
              increment walklength
              assign task to vm if indegree >0;
decrement indegree
            }
       remove task from task queque
  }

  Process completed tasks()
  {
     increment degreee of vm assigned to the task
  }
```

Advantages of Honey bee over other Algorithms

1. The communication is **d**irect. Bees employ a direct strategy by dancing in the nest (waggle dance). Their dance communicates distance and direction towards a destination and Quality and Quantity of food.

2. Bees are able to compute their present location from their past trajectory continuously. They can return to their starting point by choosing the direct route rather than retracing their outbound trajectory. (Path Integration).

3. For navigation they wander randomly and record the direction and other details along the path.

4.BCO algorithm is strong robustness, fast convergence and high flexible which allows adjustments and it represent specific knowledge of the problem by observing nature.

5. It can be used for solving multidimensional and multimodal optimization problems.
6. It is efficient when finding and collecting food, as it takes less number of steps.
4. It has ability to explore local solutions.
5. It is time saving process by structuring favourable parallel processing algorithm.

10 CONCLUSION

summary

In this paper, we have proposed a working model for load balancing in cloud computing environments based on behavior of honey bee foraging strategy. Honey bee optimization is an emerging field for researchers in the field of optimization problems because it provides immense problem solving scope for combinatorial and NP-hard problems. The tasks are to be send to the underloaded machine and like foraging bee the next tasks are also sent to that virtual machine till the machine gets overloaded as flower patches exploitation is done by scout bees. Honey bee behavior inspired load balancing improves the overall throughput of processing and priority based balancing focuses on reducing the amount of time a task has to wait on a queue of the VM. Thus, it reduces the response of time of VMs. We have compared our proposed algorithm with other existing techniques. Results show that our algorithm stands good without increasing additional overheads.

References

[1]. Harshith gupta, kalicharan sahu, International Journal of Science and Research (IJSR) ,"Honey Bee Behavior Based Load Balancing of Tasks in Cloud Computing ",
[2]. Kai Hwang. Geoffrey C.Fox. Jack J.Dongarra (2012) - Distributed and cloud computing, Mogan Kaufmann publishers
[3]. Mayanka Katyal*, Atul Mishra," A Comparative Study of Load Balancing Algorithms in Cloud Computing Environment"
[4]. Aditi Chikhalikar,Avanti Darade,"Swarm Intelligence Techniques: Comparative Study of ACO and BCO"
[5]. Farzad Sabahi, Member, IEEE, "Secure Virtualization for Cloud Environment Using Hypervisor-based Technology" .
[6]. Ramya Jayaram Masti, Claudio Marforio, Srdjan Capkun,"An Architecture for Concurrent Execution of Secure environments in cloud"
[7]. http://www.cpd.iit.edu/netsecure08/ROBERT_RANDELL.pdf
[8]. https://www.ernw.de/download/ERNW_DCVI-HypervisorsToClouds.pdf