ISSN 2394-3777 (Print)
ISSN 2394-3785 (Online)
Available online at www.ijartet.com

International Journal of Advanced Research Trends in Engineering and Technology (IJARTET)
Vol. 4, Special Issue 14, March 2017

CLUSTER BASED TASK SCHEDULING IN CLOUD ENVIRONMENT USING PSO FRAMEWORK

S.Asha, P.Deepa, R.Gowtham, Dr. K. Kousalya

Department of Computer Science, Kongu Engineering College, Erode, Tamilnadu

Abstract
Cloud computing is an emerging computing technology with a large collection of heterogeneous autonomous systems with flexible computational architecture. Main challenge in cloud computing is resource management to improve Resource utilization. One of the way to improve resource utilization in cloud computing is Virtual machine Clustering Method. K-means clustering algorithm is used for virtual machine Clustering method. k-means clustering algorithm clusters the Virtual machines based on RAM size. Task scheduling is an important step to improve the overall performance of the cloud computing. Task scheduling is also essential to reduce processing time. This paper focuses on task scheduling by clustering VMs using Particle Swarm Optimization(PSO) to optimize processing time. The result obtained by cluster based PSO was simulated by an open source cloud platform (CloudSim). Finally, the results were compared to existing scheduling algorithms and found that the proposed algorithm has reduced the makespan.

Keywords: *Task Scheduling, Cloud Computing, Clustering, CloudSim, PSO, K-Means, Virtual Machine*

Introduction

Cloud computing is the next generation computational paradigm. It is rapidly consolidating itself as the future of distributed on-demand computing .By using the concept of Virtualization, Cloud Computing is emerging as vital backbone for the varieties of internet businesses. On the other hand, Internet enabled business (e-Business) is becoming one of best business model in present era. To fulfill the need of internet enabled business, computing is being transformed to a model consisting of services that are commoditized and delivered in a manner similar to traditional utilities such as water. Users can access services based on their requirements without regard to where the services are hosted or how they are delivered. Several computing paradigms have promised to deliver this utility computing . Cloud computing is one such reliable computing paradigm. Cloud computing architecture consists of a front end and a back end. These two ends are

ISSN 2394-3777 (Print)
ISSN 2394-3785 (Online)
Available online at www.ijartet.com

International Journal of Advanced Research Trends in Engineering and Technology (IJARTET)
Vol. 4, Special Issue 14, March 2017

connected by Internet or Intranet. The front end comprises of client devices like thin client, fat client or mobile devices etc. The clients need some interface and applications for accessing the cloud computing system. The back end consists of the various servers and data storage systems. There is also a server called "Central Server". A central server is used for administering the cloud system. It also monitors the overall traffic and fulfilling the client demands in real time. The main objective of cloud computing environment is to optimally use the available resources. Scheduling algorithms play an important role in optimization process. Therefore user tasks are scheduled using efficient scheduling algorithm. The scheduling algorithms usually have the goals of spreading the load on available processors and maximizing their utilization while minimizing the total execution time. Task scheduling is one of the most famous combinatorial NP complete problem problems. The main purpose of scheduling is to schedule the tasks in a proper sequence in which tasks can be executed under problem specific constraints. This paper presents an optimization algorithm for Task scheduling to achieve Minimization of overall computation time.

Model Development

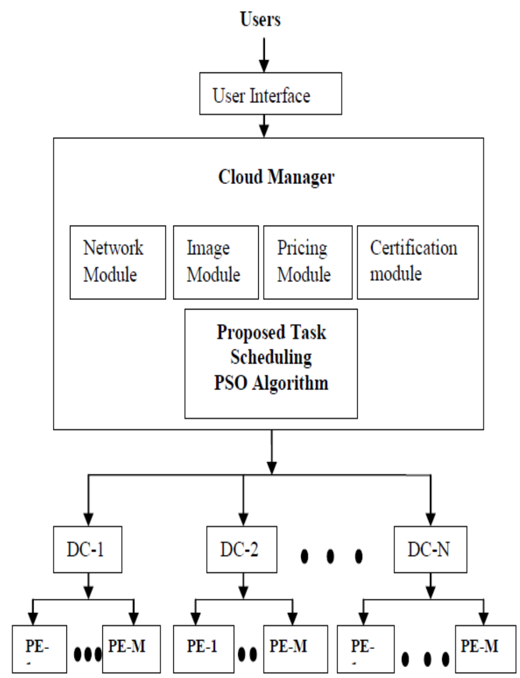

Fig. 1. Cloud Scheduling Environment

To solve the problem of resource optimization using PSO algorithm within the cloud framework, a typical cloud computing model is proposed as shown in fig. 1. The cloud system consists of many data center that are distributed geographically all over globe and are accessible using internet. Each data center consists of many computing, saving elements and other resources. Processing Elements (PEs) in each data center are connected by a high bandwidth communication network. Therefore negligible communication delay is considered in this model. In the proposed model, user can access the cloud resources using user interface. The proposed task scheduling module in the framework is responsible for efficient allocation of user tasks into different available PE with an objective to optimize energy consumption and time. In fig. 1, 'DC' indicates the Data

ISSN 2394-3777 (Print)
ISSN 2394-3785 (Online)
Available online at www.ijartet.com

International Journal of Advanced Research Trends in Engineering and Technology (IJARTET)
Vol. 4, Special Issue 14, March 2017

Center and 'PE' indicate the sets of Processing Elements.

PSO Approach

Over the last decades, heuristics that find approximate solutions have attracted great interest. From these heuristics, Particle Swarm Optimization (PSO) have been found to be very useful. The PSO algorithm was first proposed by J. Kennedy and R. Eberhart in 1995 and it was successfully used in several single-objective optimization problems. PSO is a population-based algorithm. Each individual (particle) represents a solution in a n-dimensional space. Each particle also has knowledge of its previous best experience and knows the global best experience (solution) found by the entire swarm. PSO simulates the behaviors of bird flocking. Suppose the following scenario: a group of birds are randomly searching food in an area. There is only one piece of food in the area being searched. All the birds do not know where the food is. But they know how far the food is in each iteration. So what's the best strategy to find the food? The effective one is to follow the bird which is nearest to the food.

PSO learned from the scenario and used it to solve the optimization problems. In PSO, each single solution is a "bird" in the search space. It is referred as "particle". All of particles have fitness values which are evaluated by the fitness function to be optimized, and have velocities which direct the flying of the particles. The particles fly through the problem space by following the current optimum particles.

PSO is initialized with a group of random particles (solutions) and then searches for optima by updating generations. In every iteration, each particle is updated by following two "best" values. The first one is the best solution (fitness) it has achieved so far. (The fitness value is also stored.) This value is called pbest. Another "best" value that is tracked by the particle swarm optimizer is the best value, obtained so far by any particle in the population. This best value is a global best and called gbest. When a particle takes part of the population as its topological neighbors, the best value is a local best and is called lbest.

After finding the two best values, the particle updates its velocity and positions by using equation (1) and (2).

$$V_{i,j} = W \text{ x } V_{i,j} + C_1 \text{ x } R_1 \text{ x } (P_{i,j} - X_{i,j}) + C_2 \text{ x } R_2 \text{ x } (P_{g,j} - X_{i,j}) \quad (1)$$

$$P_{i,j} = P_{i,,j} + V_{i,j} \quad (2)$$

Where 'w' is the inertia factor influencing the local and global abilities of the algorithm, $x_{i,j}$ is the velocity of the particle 'I' in the jth dimension, c1 and c2 are weights affecting the cognitive and social factors, respectively. r1 and r2 ~υ (0,1), p_i stands for the best value found by particle 'i' (pbest) and p_g denotes the global best found by the entire swarm (gbest). After the velocity is updated, the new position i in its jth dimension is calculated. This process is repeated for every dimension and for all the particles in the swarm.

ISSN 2394-3777 (Print)
ISSN 2394-3785 (Online)
Available online at www.ijartet.com

International Journal of Advanced Research Trends in Engineering and Technology (IJARTET)
Vol. 4, Special Issue 14, March 2017

PSO Algorithm
Step 1: Initialize each particle. **Step2:**Evaluate each particle's position according to the objective function. **Step 3:**If a particle's current position is better than its previous best position, update current position as new pBest. **Step 4:**Assign best particles pBest value to Gbest. **Step 5:**Update particles position and velocity by using Equation (1) and (2)**Step 6:** Repeat the above process until reach the stopping criteria.

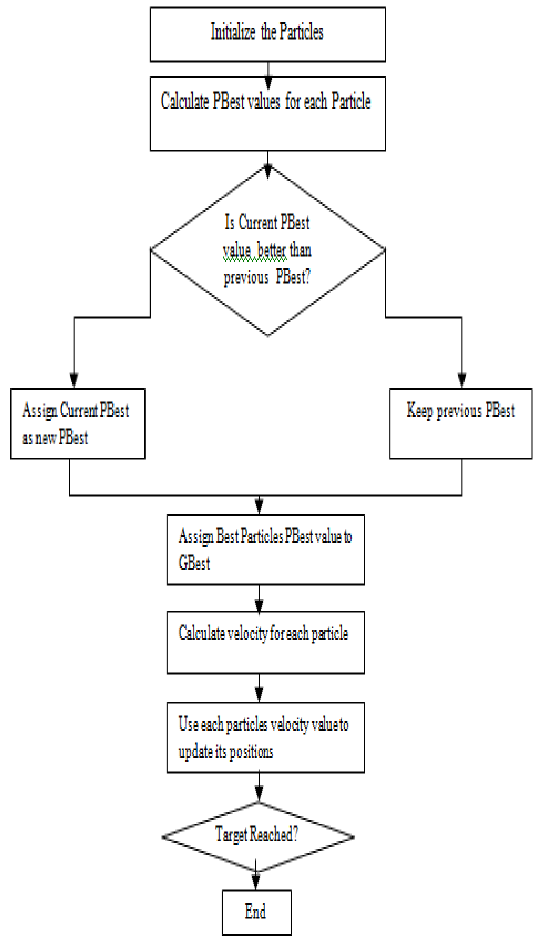

Fig. 2. Flow Diagram for PSO Algorithm

ISSN 2394-3777 (Print)
ISSN 2394-3785 (Online)
Available online at www.ijartet.com

International Journal of Advanced Research Trends in Engineering and Technology (IJARTET)
Vol. 4, Special Issue 14, March 2017

K-Means Clustering Algorithm

K-Means Algorithm
Step 1: Initialize the number of Clusters(k). **Step 2:**Randomly select k centroids. **Step 3:**from k clusters by assigning each point to its closest centroid. **Step 4**:Recompute the centroid of each cluster. **Step 5**:Repeat step4 until there is no change in centroids.

Clustering of the virtual machine is one of the keys in cloud computing to virtual sharing of physical machines in the datacenter and the efficiency of the Clustering of virtual machines based on the algorithm used . Each algorithm for Clustering of a virtual machine has its advantages and limitations on their own in a cloud computing environment.

In K-Means Clustering algorithm, clustering is done based on the RAM size of the Virtual Machine. Clustering is a method to divide a large number of VMs into small groups that have similar characteristics.

K-means clustering algorithm is a simple technique in clustering, starting from the selection of points as initial centroids K, where K is the number of clusters specified by the user. Every VM in Cloud made assignment to the nearest centroid and each set of VMs that have been made is called cluster. Centroid of each cluster then be updated based on the number of VMs contained in the cluster. Assignment process and the update process will be repeated until the centroid position has not changed or the same as the previous iteration. One way to calculate the distance between the centroid and points is the Euclidian distance as in the below equation.

$$d(X,C) = \sqrt{\sum_{i=1}^{d} |X_i - C_i|^2} \quad (3)$$

In Equation (3)**, X** is the RAM values , **C** is the cluster centroids and d is the number of Virtual Machines.

New calculations on each centroid cluster can be done with the equation (4).

$$C_i = (1/m_i) \sum_{x \in C} X \quad (4)$$

In Equation (4), *ci* is the new centroid from cluster Ci, then mi is number of objects in i-th cluster, x is an object, and Ci is ith cluster.

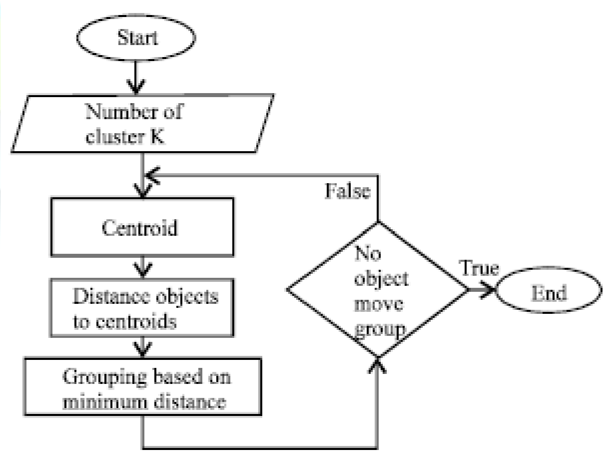

Fig. 3. Flow Diagram for K-Means Algorithm

ISSN 2394-3777 (Print)
ISSN 2394-3785 (Online)
Available online at www.ijartet.com

International Journal of Advanced Research Trends in Engineering and Technology (IJARTET)
Vol. 4, Special Issue 14, March 2017

Proposed Work

In proposed work, the Virtual machines are clustered based on the RAM size of the Virtual machines. After clustering process PSO is implemented. In K-Means clustering algorithm determine the number of clusters. Generate the random centroids for each cluster. Then find the final centroids using K-Means algorithm. Based on the final centroids Virtual machines are clustered then PSO is implemented. In PSO, Virtual machines and tasks are taken as rows and columns in population, execution matrices. Generate Execution time for each Vm. Then allocate the task to corresponding Vm based on the minimum execution time. In population matrix generate the machine values and the machine values should be unique. Calculate personal best(PBest) and global best(GBest) based on minimum execution time. Then update position and velocity matrix using PSO algorithm. Repeat the updation process until we reach the stopping criteria. When Optimized Gbest is get take that particular Vms row. Then that Vms are clustered using K-Means algorithm. Finally Tasks are scheduled efficiently. Thus Makespan is reduced by combining K-Means and PSO algorithm.

Implementation and Result

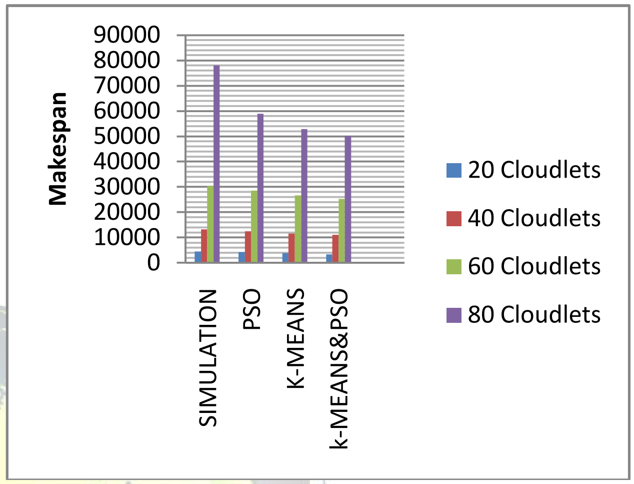

Fig. 4. Comparison of Different Approaches

In this paper, results of Cluster based Particle Swarm Optimization is compared with normal simulation, PSO and K-Means algorithms. Usage of cluster based Particle Swarm Optimization will reduce the Makespan when compared to other Algorithms.

Conclusion

This paper presented cluster based PSO optimization algorithm which can solve the task scheduling problem under the computing environment. But, in changing environment, cloud computing resources needs to be optimized. Therefore, cluster based PSO algorithm was suitable for cloud computing environment because the algorithm was able to effectively utilize the system resources to reduce makespan.

ISSN 2394-3777 (Print)
ISSN 2394-3785 (Online)
Available online at www.ijartet.com

International Journal of Advanced Research Trends in Engineering and Technology (IJARTET)
Vol. 4, Special Issue 14, March 2017

Likewise we can also cluster the tasks to reduce the makespan.

Reference

[1] J. Kennedy and R. Eberhart. **Particle swarm optimization**. In *International Conference on Neural Networks,Piscataway, NJ*. 1995.IEEE Sevice Center. p. 1942-1948.

[2] J. Kennedy and R. Mendes. **Population structure and particle swarm performance**. In *Congress on Evolutionary Computation* . **2**: 1671-1676.

[3] Khan, S., Ahmad, A., 2004**. Cluster centre initialization algorithm for k-means clustering**. Pattern Recognition Lett. 25, 1293–1302.

[4] S.Sindhu, and S. Mukherjee. **Efficient task scheduling algorithms for cloud computing environment**. In High Performance Architecture and Grid Computing, Communications in Computer and Information Science; 2011; **169:** 79-83.

[5] Rathore, S, "**Efficient allocation of virtual machine in cloud computing environment**" in *International Journal of Computer Science and Informatics,* ISSN:2231-5292, 2, 2012, 59-62.

[6] Y. C. Hsu, P. Liu, and J. J. Wu **Job sequence scheduling for cloud computing**. In Int. Conf. on Cloud and Service Computing (CSC 2011); 2011. p. 212-219 .

[7] Y. Fang, F. Wang, and J. Ge. **A task scheduling algorithm based on load balancing in cloud computing**. In Web Information Systems and Mining, Lecture Notes in Computer Science; 2010**, 6318**:271-277.

[8] J. Kennedy and R. Eberhart. **Swarm Intelligence**. Morgan Kaufmann Publishers, California,USA, 2001.

[9] K. Li, G. Xu, G. Zhao, Y. Dong, and D. Wang. **Cloud task scheduling based on load balancing Ant Colony Optimization**. In 6th Annual ChinaGrid Conf.; 2011. p. 3-9.