International Journal of Innovation in Computer Science and Information Technology
Vol.1, NO.3 , P:144 - 154
Received : 13 April 2019
Accepted : 20 June 2019

Cuckoo Algorithm for Distribution of Load in Cloud Computing

Roshanak Nouri
Master of Computer Software, Islamic Azad University, Kermanshah Branch
rooshana862@gmail.com

Abstract

Cloud computing is a computational way to deliver and deliver resources and services on a virtual basis over the Internet. Cloud computing is a new concept that divides resources and information so that users can be eliminated at any time. Cloud computing is challenging, like any other technology, as it reduces costs and delivers faster applications. One of these important challenges is the distribution of load among virtual machines. A technical load distribution that distributes workloads between multiple servers, network interfaces, or other computing resources. Load distribution will increase efficiency, reduce response time, and optimize resource utilization. This operation, by dividing the load between virtual machines in the cloud, makes it evenly distributed among virtual machines, and prevents idle time or load on one of the virtual machines. In this thesis, the goal is to use the Cuckoo algorithm to create the distribution of load among virtual machines. The Cuckoo algorithm calculates for each virtual machine a cost function that each of the machines has a lower cost function, which indicates that the virtual machine uses more efficient resources. In this thesis, we compare the results of simulation of Cuckoo algorithm and honeybee algorithm, which shows that Cuckoo's algorithm has faster convergence than honeycomb algorithm, optimal load distribution, low cost, and time. Fewer executives.

Keywords: cloud computing, load distribution, cuckoo algorithm.

1.Introduction

Cloud computing was first introduced in 1950 by Jack McCarthy [1]. The offer was able to address many of the problems that computer users faced such as lack of storage space, the need for multiple upgraded software, hardware requirements, and so on. Cloud computing offers these services to customers using the Internet, making it easier and cheaper for customers to access their needs. Cloud computing generally consists of a number of processors, and the main problem is how to coordinate and manage these processors. All of them can efficiently use the resources available in the cloud to meet the needs of users. This is known as load distribution. Load distribution leads to efficient use of resources, idle processors, customer satisfaction, and so on. For which different algorithms were proposed by experts. Applying each of these algorithms also has advantages and disadvantages. Which helps with load distribution[2].

2 .problem statement

Cloud computing is a large-scale network-based computing model that provides a new paradigm for the provision, consumption and delivery of information technology services (including hardware, software, information and other computing sharing resources) using the Internet. Some of the benefits cloud computing can bring are cost savings, increased efficiency, time savings and more. Cloud computing is a two-word combination of cloud computing and cloud computing. The cloud here is a metaphor for a network or a network of broad networks such as the Internet, which usually gives the user a behind-the-scenes look at what is happening. According to the US National Institute of Standards and Technology (NIST)[1], "Cloud computing is a model for comprehensive, easy, and order-based network access to a set of interchangeable and configurable computing resources (such as networks, storage, servers, applications) Applications and services) that can be quickly provided or released with minimal work without the need for direct service interference. " The core idea of cloud computing is to have software installed on an online server rather than being installed on users' computers. Programs are generally run in a web browser, so we can run programs from any computer that has internet access. But the ability to manage resources in the cloud due to the large scale of data, heterogeneity of data, and different goals of users in the cloud is a complex issue that also requires different ways to manage them [3]. This system is loaded with the mechanisms that cloud computing needs to become robust. It is possible to remain idle or overload on a virtual machine.

3. Load distribution

Load distribution for cloud computing is an interesting concept that offers better resources with maximum throughput with minimum response time. Load distribution means assigning each workflow to a virtual machine in the cloud. Different algorithms have been proposed to perform the load distribution. These models can be divided into two categories: static load distribution algorithms and dynamic load distribution algorithms. Static algorithms do not consider the current state of the system and perform load distribution based on prior knowledge of the system and are incompatible with changes. In this type of algorithm, a processor is known as the Boss processor, which divides the tasks between the slave processors according to their abilities and returns the results to the Boss processor after executing its tasks. The purpose of these algorithms is to minimize work execution time and limit communication overhead and latency. The dynamic algorithm makes decisions about the load based on the system's dynamic state and allows processes to migrate from a high-load machine to a lighter-weight machine, which means that the dynamic load distribution algorithm Exclusive. These algorithms are more accurate and can perform better load distribution[4].

4.Previous studies

In this section, we try to examine the proposed algorithm of the past in cloud computing:

4.1.Round Robin Algorithm, This algorithm was developed in 2012 by Ahmad and Singh. Assigns the first request to a randomly selected virtual machine and rotates the next request[5].

4.2.Min-Min algorithm, was proposed in 2009 by Saeed Parsa and Reza Entezari. The shortest query is allocated to the fastest available resource and is removed from the task set and the same process is repeated[6].

4.3.Max-Maine algorithm, is divided into 2013 by heads and blinds. It is similar to the Maine-Maine algorithm, except that the one with the highest execution time first allocates the resource with the least execution time[7].

4.4.Ant's algorithm, Marco Dorigio proposed in 1992, is inspired by real-life observations of the ant. This algorithm works by running a cloud service over a limited bandwidth network and by setting an appropriate threshold near the minimum cloud computing capacity used to search the shortest path[8].

4.5.bee colony algorithm, was proposed in 2013 by Danesh and Krishna, which was modeled on the basis of bee behavior for the

1 . National Institute of Standards and Technology

search of the bee. In this way, each server calculates the benefits and benefits of the queue by processing and applying for a request from the queue, releasing it in the absence of the necessary benefits, and using the bee dance method that acts as an ad board. Displays this panel for whole colon use. Here, each server plays the role of a parasitoid bee[9].

5.Proposed Algorithm

In this paper, using the Cuckoo Algorithm (CS), It has tried to identify the optimal points for feeding virtual machines. The cuckoo algorithm calculates a cost function for each virtual machine, each of which has a lower cost function, indicating that the virtual machine is using more efficient resources and later on the rest of the virtual machine to that point. They optimally migrate and use the resources there. The interesting thing about cuckoo migration is that these birds use the Levy flight, which is a kind of random walk that makes the cuckoo arrive earlier[1o]. in continuation, We compare the simulation results of the cuckoo algorithm and the bee algorithm which show that the cuckoo algorithm has faster convergence, better load distribution, lower cost and less running time than the honeycomb algorithm.

5.1.Describes the proposed algorith

In Figure 1. The flowchart of this algorithm is shown. The algorithm was developed in 2009 by Zain-Xi Young and Swash and was developed in 2011 by Rajibon. One of the newest and most powerful algorithms of evolutionary optimization methods is the cuckoo algorithm, which is more capable of finding global optimal points than other algorithms. Cuckoo Algorithm Like other evolutionary algorithms, it starts with a population of cuckoo. Cuckoos lay some of their eggs in the nests of some other host birds. Some eggs that resemble host bird eggs are more likely to grow and mature. Other eggs less similar to those of host birds are identified and killed by host birds. Other eggs are likely to be P% of eggs (usually 10%) identified by the host birds or the host bird itself will leave the nest. The rest of the eggs are raised in the host nests, some of the chickens left in the nest will hatch and feed on the host birds and a few will have a lower chance of survival. Eggs grown in one area show the suitability of the nests there. The more eggs that survive in an area, the greater the profit that is obtained in the area [11]. So, A situation where more eggs survive, This is a situation in which the COA is improving. Cuckoos are searching for the most suitable area for egg hatching in order to increase the survival rate of their eggs. After the remaining eggs have grown and become a cuckoo bird, Make up the community. Each group has its own area of residence, and the best residential area of all groups will be the next destination for caterers in the other group. They will reside somewhere near the best habitat. Given the number of eggs each cuckoo has and the cuckoo's distance to the target point (best habitat), a number of egg-laying radii are identified. Then the bird begins to randomly lay in nests within the radius. This process continues until the best value-maximizing position is obtained and most of the population is gathered around the same position, so this algorithm will be one of the dynamic algorithms in cloud computing if it is high. The ability and optimality of a task node will be transferred to that node. It is also focused because it only recognizes one point as the optimal point and the rest interacts with it.

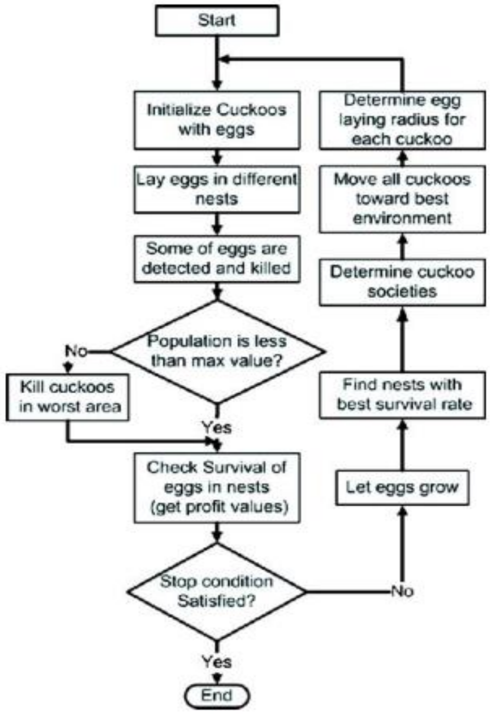


figure 1. Flowchart Cuckoo Algorithm[12].

5.2.Implementation of cuckoo algorithm in cloud load distribution

First, to implement the cuckoo algorithm for cloud balancing, simulate it as the virtual machines and their eggs as the requests sent by users or loads in the cloud. In this algorithm we have a set of virtual machines based on each of the user requests. As we know, the cloud is a collection of data centers. Each of these virtual machines also uses these data centers to respond to customer requests. Using the profit function we will calculate the amount of profit gained from using any of these data center resources and each having a lower profit function, Indicates that data center resources are optimized, which is the destination of the rest of the virtual machines. When we allocate resources based on the cuckoo algorithm we get the most benefit and this behavior of the cuckoo or virtual machines based on cost and time functions can achieve a better utilization of resources. After calculating the profit, it is best chosen as the next destination for the rest of the virtual machines.

5.3. Steps of Cuckoo Algorithm for Load Distribution of Virtual Machines

1)Creating a cuckoo's early life

Before solving an optimization problem, the problem needs to be arrayed. In the cuckoo optimization algorithm, this array is called habitat. In a subsequent Nvar problem, the habitat of the next NvarX1 arrays will be defined as relation 1:

Relation1:

Habitat = $[vm_1, vm_2, vm_3 \ldots vm_n]$

Relation2:

Xi = $[j1, j2, j3 \ldots . j_n]$

In relation 2, xi is a set of tasks devoted to virtual machines.

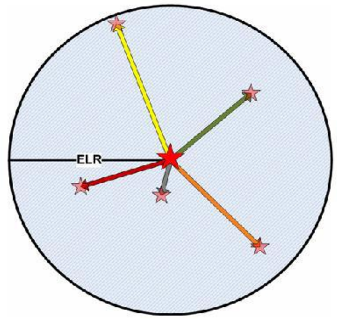


figure 2. Random spawning in ELR[13].

2) Calculate profit

The initial location of the virtual machines is the same cache generated randomly. Once we get the initial position of virtual machines we can calculate the profit for each of them based on time and cost functions. The profit of a cuckoo living is calculated by the profit function fp in a living place ($x_1, x_2, x_3, \ldots, x_n$).

As can be seen, COA is an algorithm that maximizes the profit function. To use this algorithm in cost minimization problems, one can simply maximize the following profit function:

Profit=Cost (habitat) = $f_c(x_1, x_2, x_3, \ldots, x_n)$

To use the cuckoo algorithm to solve the minimization problem, it is enough to multiply a negative sign in the cost function.

3)Calculate the number of eggs

To calculate the number of eggs per cucumber, we will use the formula 4:

Floor ((Maxeggs – Mineggs) * round) + Minegg

Where round is a random number between 0 and 1, and Floor is also a smaller integer than the expression. Maxeggs and Mineggs also have the largest and smallest number of jobs on virtual machines, respectively.

4) Laying radius of cuckoos

After calculating the cost function, the query for the virtual machine that has the least cost function is used from the resources available at that optimal point to respond to the query sent and that request will be removed from the waiting list. After specifying the optimal area, it is necessary to calculate the egg radius or the range in which the cuckoo puts its eggs by the formula of relation 5. In an optimization problem, the upper bound of var_{hi} and the lower bound of var_{low} are considered for the virtual machine deployment range [14].

Relation5:

$$\mathbf{ELR} = \alpha \frac{\textit{Number of eggs per cuckoo}}{\textit{Total number of eggs}} \times (var_{hi} - var_{low})$$

Here, variable a determines the maximum value of the egg radius. After calculating the hatching radius, each cuckoo can lay only one egg in its own radial nest, with the rest of the eggs destroyed and only one remaining.

5) Cuckoos migration

After specifying the optimal area, all virtual machines try to migrate to that area. In this region, it is difficult to identify which group each cuckoo belongs to, which will be clustered by K-Means method to solve this problem. We calculate the average profit for the formed groups. Ultimately, the most promising place to live is the new destination for migrant cuckoo. As seen in the figure, cuckoo will not fly all the way all the way to the goal. Each cuckoo will fly only λ% of its total distance to its destination. λ is a parameter that causes deviation of motion. Now that the cuckoo has reached the optimum area it should be in its new position. The following equation can be used to calculate its new position in the optimal region:

Relation6:

$X_{NextHbita} = X_{CurrentPosition} + \lambda(X_{Gol} - X_{CurrentPosition})$

$x_{CurrentPosition}$ is the previous location of the virtual machine. λ Specifies the angle of deviation of motion. x_{Gool} is our target point, for example we consider 1 that this region has the most optimal resource. This is shown in Figure 3.

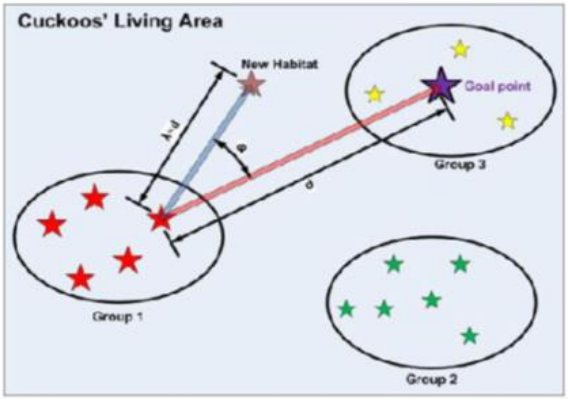


Figure 3. Migration of a cuckoo to the target's place of residence [14].

6.Convergence

After a few iterations, all cucumbers migrate to the best habitat similar to the eggs of host birds and most food sources. This place makes the most profit possible. The least egg loss occurs in this habitat. The convergence of %95 of the cuckoo to one place will be the end of the cuckoo optimization algorithm.

7. Remove cuckoo

Due to the fact that there is always equilibrium in the bird population, so N_{max} controls and limits the maximum number of live cages in the environment. This balance is due to dietary restrictions, killing by hunters, and the inability to find a suitable nest for eggs. To implement this natural equilibrium, the N_{max} of the cuckoo survives to be the most stable.

8.cost dependent

The cost function is a measure of the suitability of a solution. In fact, when we have an optimization problem, we are going to optimize one or more parameters (find the best values or choices for them). To find the optimal solution, a number of answers must be generated by the optimization algorithm. After generating these answers, the cost function is used to see which one is closest to our optimal solution. Here we have to define the cost function as Divide each workload into existing virtual machines so that the task is processed at the lowest cost possible. We obtain the optimality of an area in the cloud using the relation 7:

Relation7:

$F = x \times \sin(4x) + 1.1y \times \sin(2y)$

$X > -\infty,\ y < \infty$

We need to calculate the fit of a point first to obtain its x and y and then to calculate the fit function using the formula above [15].

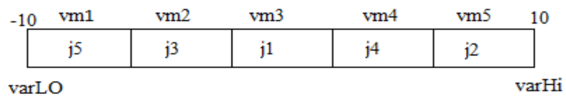
9. Simulation

To simulate the cuckoo algorithm in cloud load distribution, we will use a Claude Simulator. Cloudsim is a simulation tool for executing some cloud-related scenarios. The tool is available as a software library written in Java. To use it, you had to use a development environment like NetBeans and write your own simulation scenarios in a program using the Cloudsim library.

9.1.Proposed Algorithm Scenario

In this article, we have 6 virtual machines. We have an $N_{pop} \times N_{var}$ matrix in the range [-10,10] as follows.

We count 30 requests. The maximum number of jobs that can be assigned to any virtual machine is 25 and the minimum is 10, each of which has a random number of jobs. The length of work on each machine is Consider the following:



We first need to calculate the position of each machine in the cloud using the coordinate axis:

Virtual Machines	Location
VM1	-7
VM2	4
VM3	-4
VM4	-5.5
VM5	0.8

Table 1. Virtual Machines Location

Now, using the relation formula 7 (cost function formula), we calculate the cost function for each virtual machine:

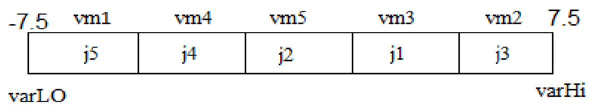
Virtual Machines	cost dependant
VM1	0.35
VM2	0.85
VM3	1.3
VM4	2.84
VM5	4.79

Table 2. cost dependant

The lowest cost function available at this stage belongs to the VM3 on which the request is executed. In this way, all tasks in the queue with the lowest cost function at each stage are selected as the destination of other virtual machines. This algorithm is executed for all the tasks in the queue and then stops performing them. We will use Formula 5 to determine the next egg laying radius:

$$ELR = 3 \times (\frac{10}{20}) \times (10 - (-10)) = 15$$

The $N_{pop} \times N_{var}$ matrix in the second step in the interval [-7.5,7.5] is as follows:



Cuckoo lays on this radius. To obtain the position of the virtual machines within this radius, we use the relation formula 6. This algorithm is executed 100 times and then terminated.

10.Comparison of the proposed algorithm

For a more detailed evaluation of the proposed algorithm, in this section we compare the performance of the scenario presented in the cuckoo algorithm with the bee algorithm.

10.1Time

The following table and figure illustrate the comparison of this algorithm in terms of time spent in both algorithms. In this experiment, different input requests are considered to obtain the runtime and 30 commands are calculated to calculate the load distribution of the algorithms:

Number of incoming requests	Cuckoo algorithm	Bee Algorithm
40	23	28
35	21	27
30	20	21
25	17	20
20	13	15

Table 3. Completion time of all algorithms in seconds

10.2.Load distribution

In this section, we compare load distribution in two bee algorithms and cuckoo algorithms with different vertexes:

Virtual Machines	Cuckoo algorithm	Bee Algorithm
VM1	900	700
VM2	800	750
VM3	900	600
VM4	550	650
VM5	440	570

Table 4. Load distribution of algorithms in watts

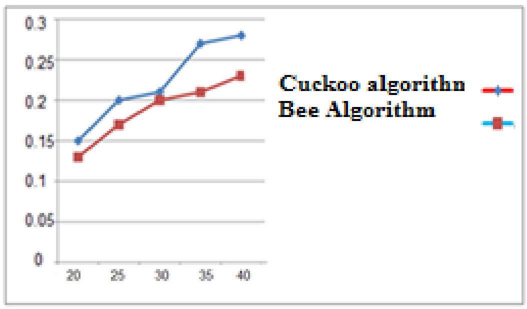


Figure 5. Timeout of each algorithm in seconds

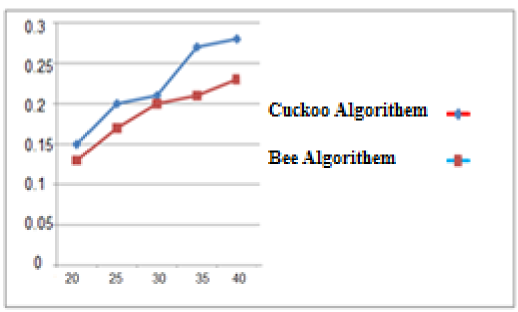


Figure 6. Load distribution algorithms in watts

As can be seen in the diagrams and figures above, the cuckoo algorithm distributes the load better and more optimally than the bee algorithm, and the speed of completion is faster. In the diagrams, taken from the first tier, vm1 performs faster and has better load distribution, proving that the region's resources are optimal. This will execute the request located on vm1 and will be removed from the existing request list.The other machines then migrate to the specified radius and at each stage the request will be executed on a less costly machine

11.Conclusion and suggestion

This article begins with cloud computing and its benefits and then addresses one of the major challenges of cloud computing, which is load distribution, and introduces a number of existing methods for load distribution. After studying these methods, a new method for load distribution based on the cuckoo algorithm was proposed. This algorithm is one of the most important criteria for determining the quality of service, is the cost function of the basis of load distribution among virtual machines. Using the cost function can find the optimal resources available in the cloud. A less costly virtual machine is an indication of the optimal resources of that cloud data center. Later, machines try to reduce their distance to the optimal source. The low-cost virtual machine request is executed and queued. This algorithm executes and then completes the number of all requests. The results of comparing this algorithm with the bee algorithm show that this algorithm has little time to perform its tasks on the other hand due to the use of better cost function in load distribution. For better performance of cuckoo algorithm in cloud load distribution It is suggested to combine the algorithm with the ant colony algorithm for future work that the ant colony algorithm can assist in the optimization of the machine path by using pheromone secretion.

Sources:

1. [1]. Kumar S. Goudar, R. H.(2012). “Cloud Computing – Research Issues, Challenges, Architecture, Platforms and Applications: A Survey. International Journal of Future Computer and Communication”, Vol. 1, No. 4.
2. [2]. Malhotra. L.Jaiswal,A and Agawal,D.(2014)." Virtualization in Cloud Computing”.
3. [3]. Chaczko,Z.Mahadvan,V . Aslanzdeh,S and Mcdermid,C.(2011).”Availabillty and Load Balacing in Cloud computing.Internatinal Conference on computer and Softwer Modeling”,14.
4. [4]. Tech,M.(2013). “Load Balancing In Cloud Computing:A Review. IOSR Journal of Computer Engineering (IOSR-JCE)”.
5. [5].Chen,h.Wang,F.Helian,N.Akanmu, G.(2013).“User-Priority Guided Min-Min Scheduling Algorithm For Load Balancing in Cloud Computing”. Parallel Computing Technologie, (PARCOMPTECH)PP.1.8.
6. [6]. kaur,R.Luthra,P.(2013). “Load Balancing in Cloud system using Max Min and MinMin Algorithm”. International Journal of Computer Applications 0975 – 8887 .
7. [7]. Begum,s. and C.S.R,P. (2013).” Review of Load Balancing in Cloud Computing”.IJCSI International Journal of Computer Science Issues, Vol. 10, Issue 1, No. 2.
8. [8]. Dhinesh, B. Krishna, L, D.(2013). “Honey bee behavior inspired load balancing of tasks in cloud computing environments”. In proc. Applied Soft Computing.
9. [9]. Begum,s. and C.S.R,P. (2013).” Review of Load Balancing in Cloud Computing”.IJCSI International Journal of Computer Science Issues, Vol. 10, Issue 1, No. 2.
10. [10]. X. S, Yang . S. Deb.(2014). *“*Cuckoo Search: Recent Advances and Applications”. Neural Computing and Applications, vol. 24, no. 1, 169-174.
11. [11]. Mohamad A, *&* Zain A M, *&* Bazin N E N, Udin A.*(*2013*).”* Cuckoo Search Algorithm for Optimization Problems – A literature Review“,pp 502-506 Applied Mechanics and Materials Vol. 421.
12. [12]. atarajan, A.,& Subramanian, P. K. (2012). "An Enhanced Cuckoo Search for Optimization of Bloom Filter in Spam Filtering". Global Journal of ComputerScience and Technology, 12(1).

13. [13].Natarajan, A., & Subramanian, S. (2012, January). Bloom filter optimization using Cuckoo Search. In Computer Communication and Informatics (ICCCI), 2012 International Conference

14. on (pp. 1-5). IEEE.

15. [14].Rani, K. A., & Malek, F. (2011, October). Symmetric linear antenna array geometry synthesis using cuckoo search metaheuristic algorithm. In Communications (APCC), 2011 17th Asia-Pacific Conference on (pp. 374-379).