International Journal of Database Theory and Application
Vol. 10, No.11 (2017), pp. 15-26
hhtp://dx.doi.org/10.14257/ijdta.2017.10.11.02

Theoretical Analysis of Bio-Inspired Load Balancing Approach in Cloud Computing Environment

Damodar Tiwari[1], Shailendra Singh[2] and Sanjeev Sharma[3]

[1]Bansal Institute of Science and Technology, Bhopal, India
[2]CSE department, NITTTR, Bhopal, India
[3]SOIT- RGPV, Bhopal
[1]damodarptiwari21@gmail.com, [2]ssingh@nitttrbpl.ac.in, [3]sanjeev@rgtu.net

Abstract

Cloud computing is one of the emerging technology in the field of IT industries. It enables to access IT related resources with the help of the high-speed network. It's become so popular in the IT industries because of their attractive features. Due to the popularity, demand for the cloud resources increases exponentially in the past few decades. To fulfill this demands numbers of datacenter is deployed which consume huge amount of energy. One of the main challenging issue for the provider is to utilize their hardware and software resources effectively with minimum response time. The performance of the system mainly depends on the server load. As the load on the server increases, performance of the cloud system leads to decreases. Hence, provider needs to balance the tradeoff between server load and performance. In order to deal with load balancing several loads balancing approach have been proposed in the past few decades. Bio-inspired approaches are also used to deal with the load imbalancing, resource allocation, performance optimization etc., because they provide the optimized solutions. In spite of the fact that it is not used in cloud computing to a more noteworthy degree, it is mainly used in data mining, networking, grid computing, pattern recognition etc. The paper gives a broad study of bio-inspired algorithms which are used to resolve various issues like load balancing, energy consumption, performance improvement etc.

***Keywords**: Cloud computing, virtualization, bio-inspired, ant colony, swan optimization, energy efficient*

1. Introduction

Cloud is one of the fastest emerging technologies in the field of IT industries as well as from the research perspective [1]. Over the past few years, cloud is one of favorite era of the research because it has several striking features like easy to use, pay for use *etc.*, [2]. Cloud offers number of services to the users and user can access these services from anywhere at any time depends on the deployment model. Cloud can be deploy or implemented in three different way *i.e.* private, public and hybrid which defines the accessibility scope of the service [3].

Virtualization [4, 5] is the driving technology in the cloud which allows application isolation and enables the service provider to relocate the virtual machine (VM) from one server or physical machine (PM) to another knows as virtual machine migration. Hypervisor is the software which is use to implement the concept of the virtualization that create the VM and assign to the user for performing the user task. Figure 1 show the virtualization stack of the PM.

Received (May 31, 2017), Review Result (October 30, 2017), Accepted (November 20, 2017)

ISSN: 2005-4270 IJDTA

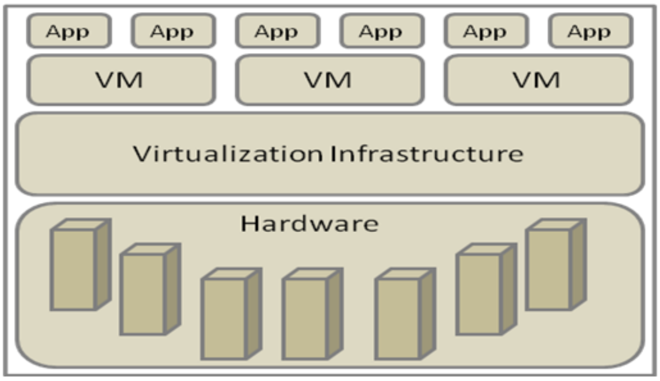


Figure 1. Virtualization Stack

In cloud number of user can use the single PM due to the virtualization, which may overload the PM. Depending on the user demand load of the server can be change at any time which results server overload or underload situation may occur [6]. To deal with these overload or underload state threshold is used which may be static or dynamic. All scheduling approaches which uses the static lower and upper threshold can't change the value of these thresholds during the entire scheduling process whereas dynamic allow user to change the threshold values.

Load balancing is one of the center issue of the cloud computing. In order to deal with the situation some virtual machine has to be migrated. VM migration [6, 7] is the key features of the virtualization. It facilitates load balancing, hot spot mitigation, server consolidation, fault management, and power saving. When the PM is overloaded or underloaded the overall performance of the PM is degrades. To mitigate the overloaded or underloaded situation VM migration is use. Since overload and underload situation may decrease the performance of the PM, so every cloud provider want to reduce the number of migration with maximum resource utilization. Due to this reason it received much attention in the last few years. Numerous load balancing algorithm have been proposed in the past. These approaches are mainly focus to increase the resource utilization by placing the VM in such a way that all VM place to the minimum number of servers. This is one of the approach by which provider can increase their profit by reducing the number of active servers. Power consumption is propositional to the number active server so reducing in number of active server will lead to the green computing. Proper placement of the VM is the way which can play down the number of migrations. To scale up the resource utilization, an efficient load balancing approach is required that behave dynamically according to the need. In the past, several load balancing approach have been proposed like foraging behavior, random sampling, biased random sampling and active clustering *etc*. [8],but now a day's researchers have been engrossed very much in the direction of real-time stochastic behaviour of social creatures like birds, ant, swam *etc*. [9, 10]. Hence, bio inspired algorithms are widely used to deal with the various situations. Bio-inspired techniques are divided into three categories named neural networks, evolutionary computation and swarm intelligence [9]. These all techniques mimics the behavior of some living species like birds, animal, insects *etc*.

The neural network approaches are inspired from the human brain and mainly used in Artificial Intelligence (AI) to solve several problems like security, decision making, optimization *etc*. Evolutionary strategies are based on the meta-heuristic population whereas Swarm intelligence approaches mimic the behavior of birds and insects and used to deal with the issues like scheduling and load balancing. Genetic algorithms (GAs) adopt the behavior of evolutionary strategy and swarm intelligence, particle swarm optimization (PSO), cuckoo search (CS), honey bee algorithms adopt the behavior of swarm intelligence. Table 1 represents the bio-inspired behavior used in cloud.

Table 1. Bio-inspired Behavior and their use in Cloud

Natural behavior	Bio-inspired Approaches	Relevance in cloud computing
Foraging (food searching)	Ant colony optimization, artificial bee colony	Load balancing, resource scheduling
Swarming (combined nature of flying birds	Particle swarm optimization, artificial bee colony	Resource migration (virtual machine placement)
Evolution-based (generates offspring)	Evolutionary, genetic algorithm approaches	VM management, VM scheduling
Breeding (lays egg in other bird nests)	Cuckoo search	Resource arrangement, finest node selection
Flashing	Firefly approaches	Ordered tasks in proper order

Main objective of this paper is to provide a broad study of bio-inspired algorithms and how they are used to resolve various issues like load balancing, energy consumption, performance improvement *etc*.

2. Resource Scheduling in Cloud

The main goal of the resource scheduling is to utilize the cloud resources with maximum rate. As cloud provides three types of services named SaaS, PaaS and IaaS, so resource scheduling is also required at the different level. Figure 2 illustrate the cloud scheduling at different layer of stack.

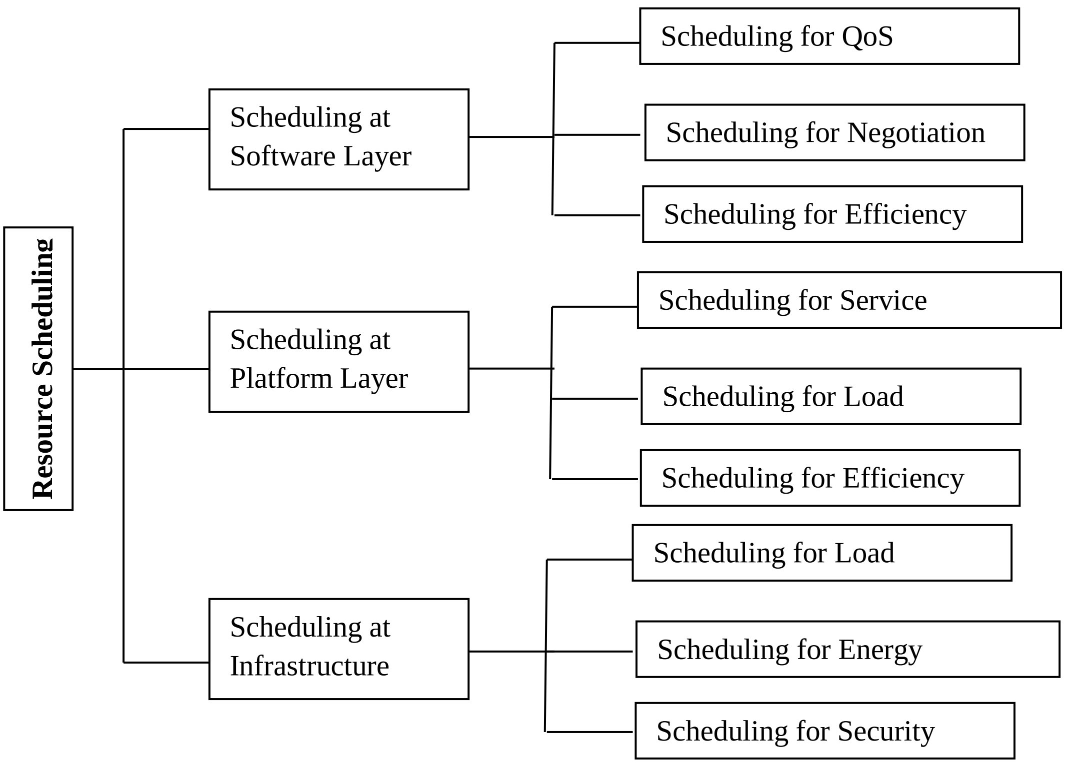


Figure 2. Cloud Scheduling at Different Layer of Stack

In cloud scheduling m task T= (t_1, t_2, t_3---------- t_m) is assign to n virtual machine VM= (vm_1, vm_2, vm_3-------- vm_n) and these VM is created in p available physical machine PM=

(pm_1, pm_2, pm_3-------- pm_p). Single task can be assign in multiple VM and multiple virtual resource can be scheduled in single PM. Figure 3 illustrate the VM scheduling problem. The primarily responsibility of cloud resource scheduling problem is to find the optimal mapping of the task, virtual resources and physical resources, which minimize the number of running PM. It is define as:

$$T \times VM \rightarrow Min \sum_{i \varepsilon P} n_p$$

Where $n_p = \begin{cases} 1 \text{ If PM is active} \\ 0 \text{ Otherwise} \end{cases}$

Table 2. Nomenclature Used in the Survey

Symbol	Description
m	Number of task need to be scheduled
n	Number of VM created to schedule m task
p	Number of PM available in the cloud data center
n_p	Binary value which represent the active or passive state of the PM
T_i	i^{th} task
VM_j	j^{th} virtual machine
PM_k	k^{th} physical machine
$\sum_{i \varepsilon P} n_p$	Total number of active PM

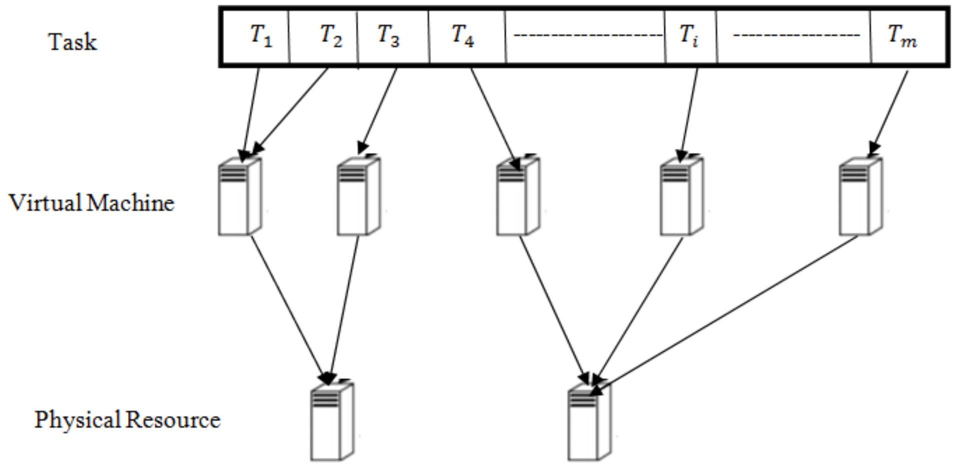


Figure 3. Task Scheduling in Cloud

3. Literature Survey

Dhinesh Babu L.D. and P. Venkata Krishna [11], proposed a load balancing approach which is inspired by the honey bee behavior. Main objective of this approach is to increase the throughput and minimize the response time of task in the queue. In this approach task is considered as honey bee and VM is considered as food source. Similarly, assigning task to the VM is similar to the honey bee foraging a food source. All VM are arranged in the ascending order of their load and when the VM is overloaded, task is migrated to the underloaded VM. This approach also considers the priority of the task during the assignment. When any higher priority task arrived in the queue than this task is assign to the VM which have the lesser number of higher priority task, so that higher priority task executed earlier. Following equation are used to calculate the capacity of the VM:

$$C^j = pe^j_{num} * pe^j_{mips} + VM^j_{bw}$$

where pe^j_{num} is the number of processing element, pe^j_{mips}j is the number of MIPS and VM^j_{bw} is the bandwidth available in Jth VM. The main limitation of this approach is that, RAM is not considered as a metrics during the load calculation which is also one of the critical element in the server.

Nishant *et al*. [12], proposed a load balancing algorithm which is inspired by ant behavior named Ant colony optimization (ACO). In this approach, VM is considered as ant and PM are considered as a nest. A number of load balancing approaches have been introduced so far which are based on the ant behavior. In the cloud, researcher is encouraged to use the ant behavior during the resource allocation because ants try to find the optimal smallest path between their nest and food source and return to their colony. It is a randomized searching algorithm, where initially ants start searching their food randomly by leaving a chemical material called pheromone. Ant used this pheromone to find the shortest path based on the pheromone intensity. All ants follow the path where pheromone intensity is higher. Figure 4 shows ant colony optimization

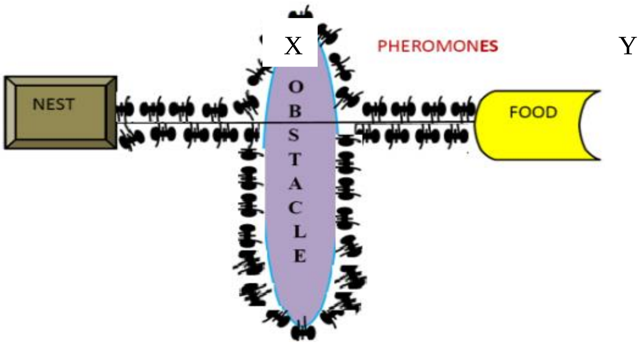


Figure 4. Ant Colony Optimization

The pheromone is evaporated in nature. So it starts evaporation with the time which results in pheromone on the largest will completely evaporate. The ants on the shortest path lay pheromones trail faster, making it additional attractive to the upcoming ants. The ants turn to following this shortest path. Hence, all ants follow the smallest path. Following equation is used to find the probability of next probable node by using pheromones values:

$$P_{xy} = \frac{[\tau_{xy}(t)]^{\alpha} * [\rho_{xy}(t)]^{\beta}}{\Sigma[\tau_{xy}(t)]^{\alpha} * [\rho_{xy}(t)]^{\beta}}$$

Where

τ_{xy} represents the quantity of pheromone on the edge x,y

α is a controlling parameter for the pheromone

ρ_{xy} represents the pleasant appearance of the edge x,y

β is a control parameter for the pheromone

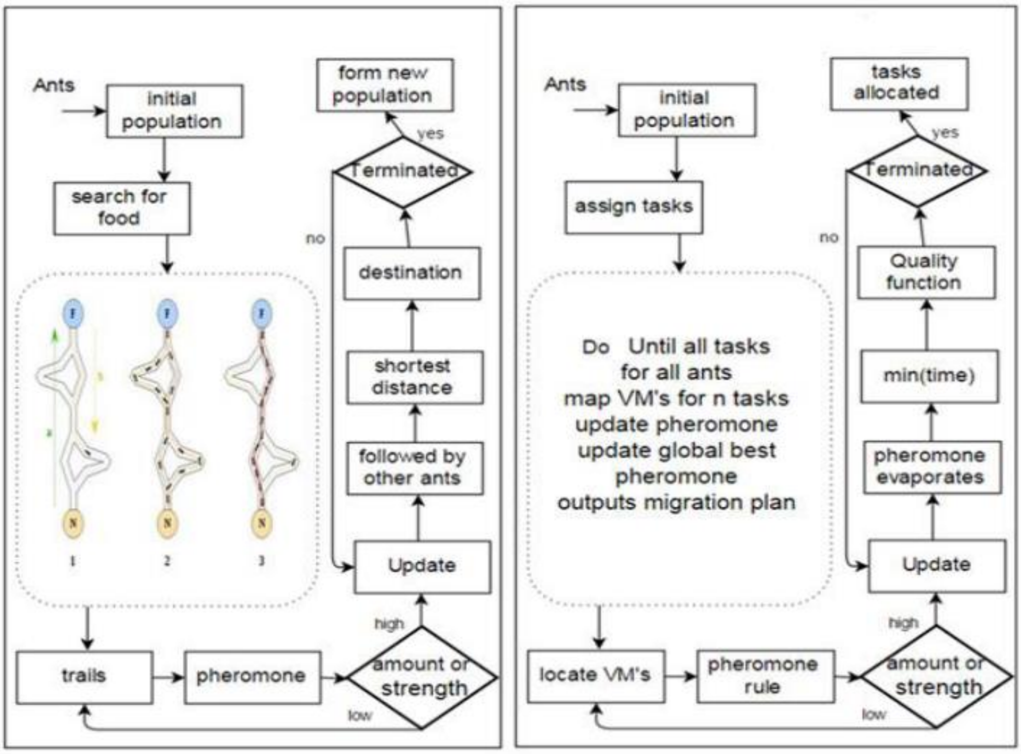
Amount of pheromone is updated according to the equation

$$\tau_{xy}(t) = (1 - \rho)\tau^k_{xy} + \Delta\tau^k_{xy}$$

$$\Delta\tau^k_{xy} = \begin{cases} \frac{Q}{L_k} \\ 0 \end{cases}$$

Where Q is the constant and L_k is the distance travel by k^{th} ant.

The main limitation of this approach is Research is experimental rather than theoretical. Moreover, this approach is focused on quality so the time required to find optimal path is uncertain.



(a) (b)

Figure 5. (a) ACO Behavior Inspired Model (b) ACO Behavior in Cloud

R. Kumar and T. Prashar [13], proposed a bio-inspired hybrid approach to deal with load balancing issue in cloud. This approach used the ant colony optimization (ACO) and priority-based artificial bee colony (ABC) to deal with the load. ACO approach was used for the load balancing and ABC approach was used to optimize the resource utilization by scheduling the virtual machine effectively. Cloud analyst simulation tool is used to simulate their approach. This approach claims that it reduce the average response time, processing cost and average processing time.

Table 1. Evaluation of Different ACO based Scheduling Approaches

Improvement Strategy	Evaluation Metrics	Task Nature	Simulation Environment
Basic ACO [18]	Total simulation time	Self-Governing	CloudSim simulator toll is used in cloud environment.
Modified pheromone updation policy [19]	Total simulation time	Self-Governing	GridSim simulation tool is used in grid environment.
Based on the pheromone evaporation rate Modified pheromone updation policy [20]	Load Balancing, Total simulation time	Self-Governing	GridSim simulation tool is used in grid environment
Combined ACO	Ratio of physical	Workflow	Matlab tool in cloud

scheduling approach with PSO approach [21]	resource utilization, Total simulation time		environment.
Update load balancing policy by proper setting of lower and upper thresholds are used to recognized overloaded and underloaded nodes [22]	Power consumption, Load balancing, Total simulation time, SLA violation	Self-Governing	CloudSim simulator toll is used in cloud environment.

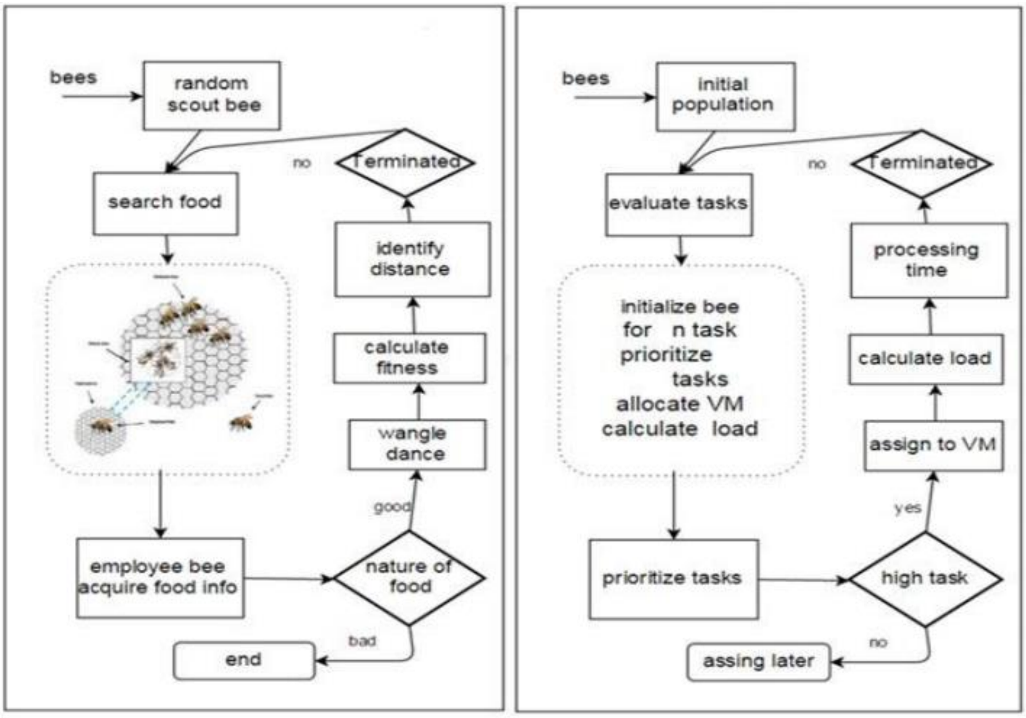
Karaboga and Basturk [14], introduced a bio-inspired load balancing approach which mimics the behavior of bee known as Artificial Bee Optimization (ABC). It is a relatively new member of swarm intelligence, which can mimic bee behavior to find food source. Bee does the wangles dance to find the optimal location of nectar (food) source. by performing this dance, foragers sharing the information about the direction, amount of food and distance of source food. In this approach bee categorized into three types based on their behavior *i.e.*, employed bee, scout bee and onlooker bee. Scout bee is responsible for finding new food source and get the information from the employed bee in the hive. It is responsible for selecting a food source to deposit their food. Scout bee move randomly to find the food source and when it find the food source, calculate the fitness value. Employed bee stays on the food and provides information to the neighborhood and estimate new fitness value. Onlooker bee compare these two fitness value and if the fitness value calculate by scout is lesser than by the current fitness value (calculated by employed bee) then its send the onlooker bee to the food source which is suggested by employed bee, again calculate the fitness value and select the path and source with best fitness value. The employed bee whose source food is finished becomes a scout bee and starts searching for new source food. The main objective of the bee is to maximize the E/T ratio, where E represents the amount of food discovered by bee and T is the time spends to search food. If θ_i represent the position of i[th] food source and F(θ_i) is the amount of food available at θ_i then the probability to find the food source available at θ_i can be calculated as

$$P_i = \frac{F(\theta_i)}{\sum_{k=1}^{s} F(\theta_k)}$$

where S represents the number of food source near the hive. Onlooker bee calculates the fitness value of all available food source around hive and select one of the source food with higher fitness value. Following equation is used to find the position of the neighbor food source around hive

$$\theta_i(c+1) = \theta_i(c) \pm \emptyset_i(c)$$

where $\theta_i(c)$ shows random movement of bee to find food source around the highest fitness value source food, $\emptyset_i(c)$ is the difference between $\emptyset_i(c)$ and $\emptyset_k(c)$ food position. If amount of food $F(\theta_i(c+1))$ at $\theta_i(c+1)$ is more than $\theta_i(c)$, then bee move to the hive and spread his information to all bee in the hive which results bee change their food source from location $\theta_i(c)$ to $\theta_i(c+1)$, otherwise $\theta_i(c)$ remains food source. This approach is different from the ACO. Which leave the pheromones, instead they communicate to each other by the wangle dance. Through the wangle dance s employed bee communicate the location of its discovery to idle onlooker bee. Main limitation of this approach is that search space is limited to the initial solution.



(a) (b)

Figure 6. (a) ABC Behavior Inspired Model (b) ABC Behavior in Cloud

S. Sharma *et al*, [15], proposed an approach for the load balancing in cloud which mimics the bats behavior. Initially bats fly randomly but they changed their velocity, frequency, position and pulse emission rate according to the distance between prey and itself. It is an iterative approach so after the multiple iteration bats will find optimal path between prey and itself. This bats behavior simulated in cloud to find the best suitable VM for scheduling the task. When any task reached to the job pool, VM scheduler call bats algorithm to locate suitable server which fulfills the job requirement. Once bats algorithm find the appropriate server, its place the VM to the selected PM. Then load is calculated and if load is greater than all another server then job is distributed to multiple servers. This approach was compared with Round Robin and Fuzzy GSO and implemented in Matlab. Response time is measured corresponding to the number of task in all approaches and claim that it has a better response time as compared to the competitive approaches.

S. Pandey *et al.* [16], developed a heuristic approach for scheduling the VM in cloud which is inspirited by the Particle Swarm. This technique is inspired from the natural social behavior as well as dynamic moments with the communication of insects and fishes. Flock of birds moving randomly for finding the source of food. The birds those find the food source or near to the food, chirps stridently. When other birds hear this chirps they are moving in this direction. If any of the other rotating birds find the food source, it peep (chirps) louder which results some birds start moving toward him. This process is continues until one of the birds reach up to the food. Following equation is used to find the optimal path

$$V_i^{k+1} = wv_i^k + c_1 rand_1 \times (pbest_i - x_i^k) + c_2 rand_2 \times (gbest - x_i^k)$$

$$x_i^{k+1} = x_i^k + v_i^{k+1}$$

where:

V_i^k i^{th} particle velocity at k iteration

V_i^{k+1} i^{th} particle velocity at k+1 iteration

w inertia weigh
c_j acceleration coefficients j=1,2
$rand_1$ random number between 0 and 1; i=1,2
x_i^k i[th] particle current position at k+1 iteration
$pbest_i$ particle i best position
$gbest$ best particle position in a population
x_i^{k+1} i[th] particle position at k+1 iteration

Main goal of this strategy is to reduces the computation cost and data transmission cost. This approach is similar to the other approaches which we have discussed so far like ACO, Genetic algorithm *etc*. This method is became so popular because of their wide application with minimum calculation and simplicity. Flowchart for the approach is given below

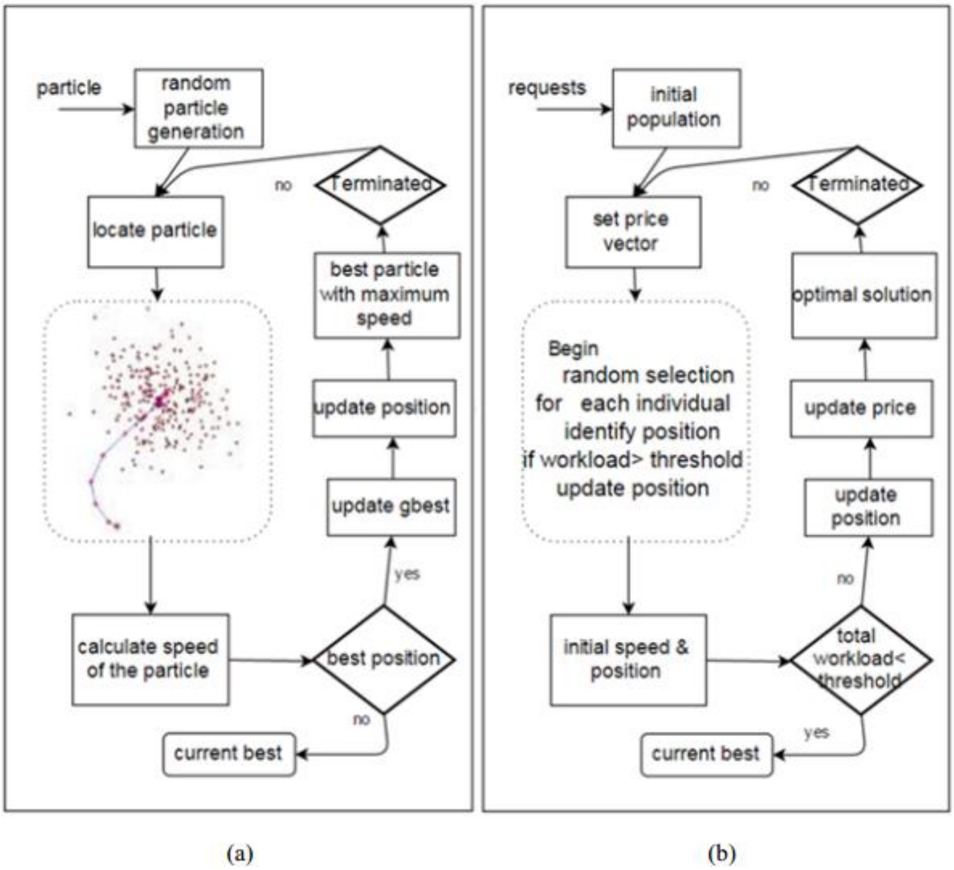


Figure 7. (a) Particle Swarm Behavior Inspired Model (b) Particle Swarm Behavior in Cloud

Dasgupta *et al*. [17], introduced a genetic algorithm for handling the load balancing issue in cloud. In GA we randomly select a processors and perform GA and processor having highest finess value select again and again this leads to starvation. To deal with the starvation this paper uses priority strategy which give some priority to each VM based on the fitness and then this prioritized input is given to the genetic algorithm which results improve response time.

For assigning priorities to VM's Logarithmic Least Square Matrix technique is used. This approach uses the processing unit vector (PUV) and job unit vector (JUV) to show the current utilization of the process and job requirement respectively. PUV and JUV is given byPUV = f (MIPS,α,L)

$JUV = f(t, NIC, AT, wc)$

where
MIPS represent million instruction per second
α Instruction execution cost
L is a delay cost
NIC Number of instruction in task
AT Task arrival time
wc is the worst completion time

In cloud, provider wants to allocate N task to the M number of processor. Following function is used to calculate the cost

$\zeta = w_1 * \alpha(NIC/MIPS) + w_2 * L$

where
w_1 and w_2 are the weighting coefficient
ζ represent the cost or fitness function

This approach assign the priority to each task and VM are assign to the task according to the priority. Main limitation of this approach is the starvation because lower priority task have to wait for a infinite time. Table 2 depicts the comparative analysis of different Bio-Inspirited algorithm in term of performance, scalability, time/cost fault tolerance and security.

Table 3. Comparative Analysis of Different Bio-Inspirited Algorithm

Algorithm	performance	Scalability	Time/Cost	Fault Tolerance	Security
Ant colony optimization (ACO)	Yes	Yes	Yes	Yes	Yes
Particle swarm optimization (PSO)	Yes	Yes	Yes	No	No
Evolutionary algorithm (EA)	Yes	No	Yes	No	No
Genetic algorithm (GA)	Yes	Yes	Yes	No	No
Artificial bee colony (ABC)	Yes	Yes	Yes	Yes	No
Hybrid algorithm	Yes	Yes	Yes	Yes	-

4. Conclusion

Cloud is become so popular in very short time which results in increasing the number of users. To fulfill this demand huge number of severs are installed in each data center. In order to increase the performance of the cloud services and resource utilization proper balancing of the physical resources are required. Several load balancing algorithms have been introduced in the last decades. Bio-inspired approaches are also used to deal with the load imbalancing, resource allocation, performance optimization *etc.*, because they provide the optimized solutions. In spite of the fact that it is not used in cloud computing to a more noteworthy degree. This paper, comprehensively surveys various bio-inspired algorithms which is used in the cloud environment to deal with the various critical issue like load balancing, VM scheduling *etc.* Bio-inspired techniques are divided into three categories named neural networks, evolutionary computation and swarm intelligence.

These all techniques mimics the behavior of some living species like birds, animal, insects *etc*. Table 1, show the comparison between swarm optimization, hybrid algorithms and evolutionary computations. In addition, performance investigation of different bio-inspired algorithms has been done.

Competing Interests

All authors declare that they have no competing interests.

Authors' Contributions

"Damodar Tiwari carried out the genetic studies, participated in the design and implementation of the proposed approach. He also participated in the sequence alignment and drafted the manuscript. Dr. Shalindra Singh and Dr. Sanjeev sharma both participated in its design and coordination, implementation and helped to draft the manuscript. All authors read and approved the final manuscript".

References

[1] R. Buyya, "Cloud Computing and Emerging IT Platforms: Vision, Hype and Reality for Delivering Computing as the 5th Utility", Future Generation Computer Systems, **(2011)**.

[2] J. Carretero and J. G. Blas, "Introduction to cloud computing: platforms and solutions" Int. Journal of Cluster Computing (Springer), vol. 17, issue 4, **(2014)**, pp. 1225-1229.

[3] R. K. Gupta, "A Complete Theoretical Review on Virtual Machine Migration in Cloud Environment", International Journal of Cloud Computing and Services Science (IJ-CLOSER), vol.3, no.3, **(2014)**, pp. 172-178.

[4] Barnham, "Xen and the art of virtualization", Proceedings in 19th ACM Symposium on Operating System Principles (SOSP'03), **(2003)**, pp. 164–177.

[5] VMware, VMware ESX and VMware ESXi. [Online]. Available: http://www.vmware.com/files/pdf/VMware-ESX-and-VMware-ESXi-DS-EN.pdf.

[6] R. W. Ahmad, "Virtual machine migration in cloud data centers: a review, taxonomy and open research issues", The Journal of Supercomputing (Springer), vol. 71, **(2015)**, pp. 2473-2515.

[7] R. W. Ahmad, "A survey on virtual machine migration and server consolidation frameworks for cloud data centers", Journal of Network and Computer Applications, **(2015)**, pp. 11–25.

[8] M. Randles, D. Lamb and Taleb-Bendiab, "A Comparative Study into Distributed Load Balancing Algorithms for Cloud Computing", IEEE 24th International Conference on Advanced Information Networking and Applications Workshops, **(2010)**.

[9] B. Balusamy, "Bio-inspired algorithms for cloud computing: a review", Int. J. Innovative Computing and Applications, vol. 6, no. 3, **(2015)**, pp. 181-202.

[10] D. Karaboga, "A comprehensive survey: artificial bee colony (ABC) algorithm and applications", in Artificial Intelligence Review, Springer Science Business Media B.V., **(2012)**.

[11] L.D. Dhinesh Babu and P. Venkata Krishna, "Honey Bee Behavior Inspired Load Balancing of Tasks in Cloud Computing Environments", Applied Soft Computing (Elsevier), vol. 13, **(2013)**, pp. 2292–2303.

[12] K. Nishant, P. Sharma and V. Krishna, "Load balancing of nodes in cloud using ant colony optimization", proceeding in 14th IEEE International Conference on Modelling and Simulation, **(2012)**.

[13] R. Kumar and T. Prashar, "A Bio-Inspired Hybrid Algorithm for Effective Load Balancing In Cloud Computing", Int. J. Cloud Computing, vol. 5, no. 3, **(2016)**, pp. 218-246.

[14] D. Karaboga and B. Basturk, "On the performance of artificial bee colony (ABC) algorithm", Journal of Applied Soft Computing, vol. 8, **(2008)**, pp.687–697.

[15] S. Sharma, "An Optimal Load Balancing Technique for Cloud Computing Environment using Bat Algorithm", Indian Journal of Science and Technology, vol. 9, **(2016)**, pp. 1-4.

[16] S. Pandey, "A Particle Swarm Optimization-based Heuristic for Scheduling Workflow Applications in Cloud Computing Environments", 24th IEEE International Conference on Advanced Information Networking and Applications (AINA), **(2010)**.

[17] K. Dasgupta, "A Genetic Algorithm (GA) based Load Balancing Strategy for Cloud Computing", proceeding of the First International Conference on Computational Intelligence: Modeling Techniques and Applications (CIMTA), **(2013)**, pp. 340-347.

[18] K. Kousalya, "To improve ant algorithm's grid scheduling using local search", International Journal of Computer Cogn., **(2009)**, pp. 47–57.

[19] P. Mathiyalagan, S. Suriya and S.N. Sivanandam, "Modified ant colony algorithm for grid scheduling", International Journal of Computer Science and Eng, **(2010)**, pp. 132–139.

[20] A.L.A. Liu and Z.W.Z. Wang, "Grid task scheduling based on adaptive ant colony algorithm", proceeding of International conf management e-commerce e-government. IEEE; **(2008)**, pp. 415–421.

[21] X. Wen, M. Huang and J. Shi, “Study on resources scheduling based on ACO algorithm and PSO algorithm in cloud computing”, Proc. of 11th int symp distribution computer application to business eng science, **(2012)**, pp. 219–225.
[22] S. Khan and N. Sharma, “Effective scheduling algorithm for load balancing (SALB) using ant colony optimization in cloud computing”, International Journal of Advance Res Computer Eng, **(2014)**, pp. 966–73.

Authors

Damodar Tiwari, he has been working at the Bansal Institute of Science and Technology, Bhopal since 2006. He holds a B. E. and M.Tech in Computer Science. His are of research is cloud computing, parallel computing.

Shailendra Singh, he is a professor and head of the department in CSE dept. at the NITTTR, Bhopal. He holds M.Tech and PhD in Computer Science. His are of research is Support Vector Machine, Machine Learning, Software Engineering, Network Security & Image Processing, cloud computing, parallel computing.

Sanjeev Sharma, he is a professor and head of SOIT- RGPV, Bhopal. He holds M.Tech and PhD in Computer Science. His are of research is Soft Computing, Machine Learning, Network Security & Image Processing, cloud computing, parallel computing.

