iJRASET
International Journal For Research in
Applied Science and Engineering Technology

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Volume: 2 **Issue:** IV **Month of publication:** April 2014

DOI:

Vol. 2 Issue IV, April 2014
ISSN: 2321-9653

INTERNATIONAL JOURNAL FOR RESEARCH IN APPLIED SCIENCE AND ENGINEERING TECHNOLOGY (IJRASET)

Load Balancing In Cloud Computing Environment Using Pso Algorithm

Anju Baby

Karunya University

Abstract: Cloud computing is an entirely internet-based approach where all the applications and files are hosted on a cloud which consists of thousands of computers interlinked together in a complex manner. The HBB load balancing model is not assign task to proper virtual machine and also it does not consider the Quality of Service. In order to overcome the drawback of honeybee algorithm another algorithm called PSO algorithm is used. In PSO algorithm task will be assigned to the virtual machine in best fit manner. i.e task will check all the virtual machine and assigns the task to proper virtual machine which will have least memory wastage as we have taken that as the QoS In this project comprehensive multi-objective model for optimized task scheduling to minimize task completion time and task response time. However, the objective functions in this model are in conflict with one another. The simulation results show that the proposed method has the ability to find optimal trade-off solutions for task scheduling problems that represent the best possible compromises among the conflicting objectives.

Keywords: Load Balancing,PSO algorithm, HBB-LB algorithm,trust, resource prediction

1. INTRODUCTION

Cloud computing is an entirely internet-based approach where all the applications and files are hosted on a cloud which consists of thousands of computers interlinked together in a complex manner.Load balancing scheme mainly focuses on the following as their main criteria to be fulfilled:speed up the execution of applications on resources whose workload varies at run time in unpredictable way,Decrease in time of job execution and tardiness,Achieving distribution of load fairly, appropriately and optimally.Load balancing is defined as the enhancement of resources, parallel utilization, improvement in throughput, and cutting of response time via an appropriate distribution of load to the available resources as shown in Fig. 2.1. These computational environments are consists of multiple heterogeneous computing modules, these modules interact with each other to solve the problem. A proper scheduling policy attempts to assign these loads to available computing nodes so as to complete the processing of all loads in the shortest possible time. To improve the utilization of the processors, parallel computations require that processes be distributed to processors in such a way that the computational load is spread among the processors. Load balancing means shifting of tasks from one machine to another machine.

2. LITERATURE SURVEY

Narasimham et al.,2011) explains gradient based design methods are well developed for the design of adaptive FIR filters and widely applied to the distinct areas such as noise

INTERNATIONAL JOURNAL FOR RESEARCH IN APPLIED SCIENCE AND ENGINEERING TECHNOLOGY (IJRASET)

cancellation, system identification and channel equalization. Gradient based design approaches may often get stuck at a local minimum in a multi-modal error surface and the stability of the designed filter cannot be ensured. However, global optimization algorithms based approaches are able to converge to the global minimum in a multi-modal error surface and ensure the stability of the adaptive IIR filter. Proposed swarm intelligence based global optimization algorithms are the artificial bee colony algorithm, which simulates the intelligent foraging behavior of honeybee swarms. In this work, a novel approach based on artificial bee colony algorithm is introduced for the design of adaptive FIR and adaptive IIR filters.

(D. Emerson et al.,1998) explains the problem of redistributing the workload on parallel computer is considered.an optimal redistribution algorithm, which minimizes the Euclidean norm of the migrating load is derived.in order to achievegood performance on parallel computer, it is essential and necessaryto balance the work load among the processors.

(BibhudaSahoo et al.,2007) says as, Load balancing is a crucial issue in parallel and distributed systems to ensure fast processing and optimum utilization of computing resources. Load balancing strategies try to ensure that every processor in the system does almost the same amount of work at any point of time. The load-balancing problem, aim to compute the assignment with smallest possible makespan The load distribution problem is known to be NP-hard in most cases and therefore intractable with number of tasks and/or the computing node exceeds few units. Here, the load balancing is a job scheduling policy which takes a job as a whole and assign it to a computing node. Two job classes are considered for the study, the jobs of first class are dedicated to fast processors. While second job classes are generic in the sense they can be allocated to any processor. The performance of the scheduler has been verified under scalability.

(A.Shanmugam et al.,2011) says as, cloud Computing is a form of distributed computing that involves coordinating and sharing computing, application, network resources across dynamic and geographically dispersed organizations. The primary issue associated with the efficient utilization of heterogeneous resources in a cloud is cloud scheduling. The main objective of cloud scheduling is to get the best optimal machine to each task, which makes scheduling a complex problem. Heuristic approach is developed to obtain optimal solution. In this paper, a Hybrid Ant Colony Optimization (HACO) scheduling algorithm is proposed. Experiments are conducted with different data series and conditions. The experimental results reveal that the proposed algorithm produces better results when compared with the existing ant algorithm. The proposed scheduler proves that best suitable resource is allocated to each task with reduced make span and execution time when compared with the existing algorithm.

3. PROPOSED APPROACH

There are various algorithms designed for balancing the load among different tasks. After completing the literature survey we are able to conclude that most of the load balancing algorithms proposed so far is complex, and not able to implement. Therefore in order to solve these problems load balancing using an algorithm Honey bee Inspired Load Balancing of tasks is proposed. This algorithm maximizes the throughput. It uses makespan and response time as performance metrics also this algorithm is more effective than any other load

Vol. 2 Issue IV, April 2014
ISSN: 2321-9653

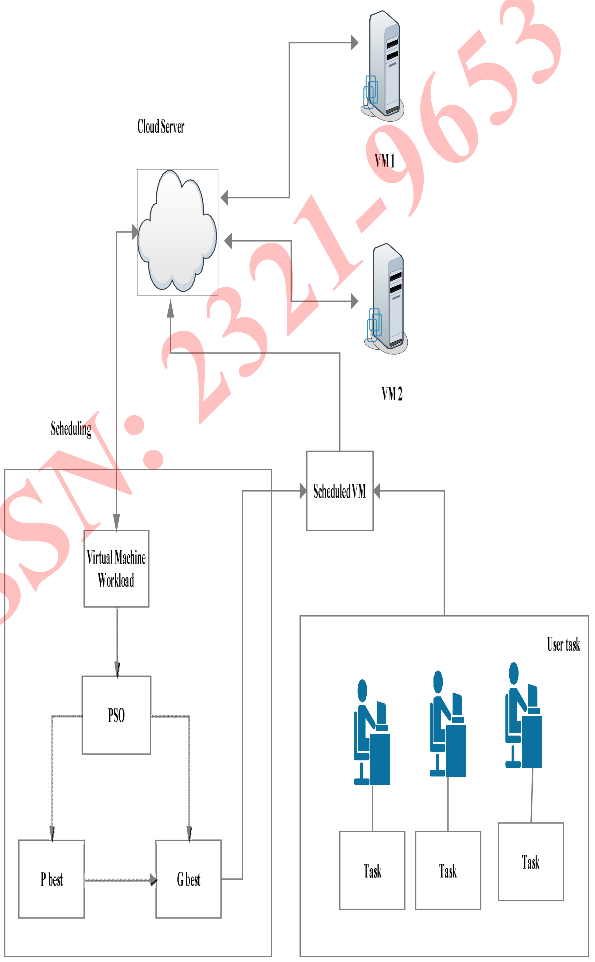
INTERNATIONAL JOURNAL FOR RESEARCH IN APPLIED SCIENCE AND ENGINEERING TECHNOLOGY (IJRASET)

balancing algorithm. Honeybee algorithm assigns every task into virtual machine in first come manner. It won't leave the virtual machine as idle.

Main disadvantage of honeybee algorithm is, it assigns the tasks in first come manner. i.e it won't check all the free virtual machine. It will directly assign to the first virtual machine. Sometimes there is wastage of memory in virtual machine. For example if are planning to assign 20 MB task, then we will assign the task into first free virtual machine. Sometime that virtual machine of size 500 MB and there may be another 25 MB virtual machine. Then this may causes the wastage of memory. In order to overcome the drawback of honeybee algorithm another algorithm called PSO(particle swarm optimization) algorithm.

(Zhen Xiao.,2013) says in PSO algorithm task will assign to the virtual machine in best fit manner. i.e task will check all the virtual machine and assigns the task to proper virtual machine which will have least memory wastage. User sends their task request to the cloud server. And this cloud server will decide virtual machine to store that task. Cloud server will select the virtual machine based on the particle swarm optimization algorithm. Initially we are creating an account in cloudme. After creating the account next create virtual machine and upload some files in it.

Our aim is to balance the load when there is an overload in virtual machine. First step is to upload the file and cloud server will accept the request and it will transfer that request to virtual machine. User control will initiate the process and give control to the vm scheduler. Main function of vm scheduler is performs the load balancing using PSO algorithm. Based on the threshold value only we are finding the overloaded virtual machine. After finding the overloaded virtual machine, next step is to migrate the task from overloaded virtual to under loaded virtual machine.



: Overall architecture of the proposed system

3.1 PSO ALGORITHM

Begin

Calculate the load, capacity of a virtual machine

Calculate pbest and gbest for each machine .

Do

Update load and capacity of virtual machine
Calculate VM future resource need value of each machine. Update pbest for each machine.
Update gbest for each machine
Choose the low loaded machine and migrate task from overloaded machine

While

Termination criterion is not violated.

End

Termination criterion is not violated.

End

Predict the future resource needs of VMs. As said earlier, our focus is on Internet applications. One solution is to look inside a VM for application level statistics, e.g., by parsing logs of pending requests.

Cpu usage=file size+bandwidth

Doing so requires modification of the VM which may not always be possible. Instead, we make our prediction based on the past external behaviors of VMs.

Aim to migrate the task from VM of overloaded to under loaded VM, that can reduce the server's overload.

Trust = min(load)

The HBB load balancing model has not assign task to proper virtual machine and does not consider the Quality of Service. A population of candidate solutions and particles are moved around in the search-space according to a few simple formulae. The movements of the particles are guided by their own best known position in the search-space as well as the entire swarm's best known Virtual machine based HBB load.

Underloadd VM= average(load in other virtual machine)

(Zhen Xiao.,2013) says when improved positions are being discovered these will then come to guide the movements of the swarm. The process is repeated and by doing so it is hoped, but not guaranteed, that a satisfactory solution will eventually be discovered.

Our algorithm executes periodically to evaluate the resource allocation status based on the predicted future resource demands of VMs. We define a server as a hot spot if the utilization of any of its resources is above a hot threshold. This indicates that the server is overloaded and hence some VMs running on it should be migrated away. We define a server as a cold spot if the utilizations of all its resources are below a cold threshold. This indicates that the server is mostly idle.

The physical machines provide a set of virtual machines which are configured dynamically according to user requests. When the limited physical machines are provided to users from a pool of resources, the provided resources have two types; one is the dedicated resources and the other is the undedicated resources to give some extra margin in case of sudden request In this Cloud system environment, if a new user requests resources when all of the resources are already assigned, then the undedicated resources allocated to others are provided to the new users via dynamic reconfiguration. Here we calculate the trust model based on the historical information. By analyzing the vm load and the load of resource the trust value is calculated.

3.2 MATHEMATICAL MODEL

Let VM = {VM1,VM2,VM3,VM4} be the set of 4 virtual machines which should process n tasks represented by the set T = {T1, T2,. . ., Tn}. All the machines are unrelated and parallel and are denoted as R in the model. We schedule non-preemptive independent tasks to these VMs. Non-preemptive tasks are denoted as npmtn. Non-preemption of a task means that processing of that task on a virtual machine cannot be interrupted.We denote finishing time of a task Ti by CTi. Our aim is to reduce the memory usage which can be denoted as Mem_{max}. So our model is R|npmtn|Mem_{max}. Processing time of a task Ti on virtual machine VMj can be denoted as Pij.

$P_j = \sum_{i=0}^{n} P_{ij}$ j=1, . .4

By minimizingMem_{max} ,$\sum_{i=1} P_{ij} \leq Mem_{max}$j= 1,..4

$\rightarrow P_j \leq Mem_{max}$ j= 1,..4

(DhineshBabuet.al.,2013) saysAt the time of load balancing, the tasks will be transferred from one VM to other in order to reduce Mem_{max} as well as response time. Processing time of a task varies from one VM to other based on VM's capacity. In case of transferring, completion time of a task may vary because of load balancing. Optimally,

Mem_{max}={ $max_{i=1}^{n} Mem_i$, $max_{j=1}^{n} \sum_{i=1}^{n} P_{ij}$}

(Zhen Xiao.,2013) saysCurrent workload of all available VMs can be calculated based on the information received from the datacenter.

1) Capacity of a virtual machine

Ci= file size + cpu usage

2) Capacity of all VMs

C= $\sum_{i=n}^{n} C_i$

(DhineshBabu et.al.,2013) saysSummation of capacity of all VMs is the capacity of data center.

3) Load on a VM

Total length of tasks that are assigned to a VM is called load. Load of a VM can be calculated as the Number of tasks at time t on service queue of VMi divided by the service rate of VMi at time t.

Li= $\frac{No\ of\ tasks}{service\ rate}$

(DhineshBabu et.al.,2013) saysLoad of all VMs in a data center is calculated as

L= $\sum_{i=1}^{n} L_i$

4) Calculate VM future need

Future need= ((total length/(1024*1024))* alpha)+(1-alpha);

(Zhen Xiao.,2013) saysAlpha value is a randomly selected value below 1.total length is the total length of files in each virtual machine.

5) Load balancing decision.

(DhineshBabu .,2013) saysAfter finding the workload and future need, the system should decide whether to do load balancing or not. For this, there are two possible situations i.e., (1)Finding whether the system is balanced (2) Finding whether thewhole system is saturated or not

6) Task migration

(Zhen Xiao.,2013) saysAfter finding the lower loaded VM migrate the tasks of particular size from overloaded to underloaded.

4 IMPLEMENTATION

4.1 Prediction

Predict the future resource needs of VMs. As said earlier, our focus is on Internet applications. One solution is to look inside a VM for application level statistics, e.g., by parsing logs of pending requests. Doing so requires modification of the VM which may not always be possible. Instead, we make our prediction based on the past external behaviors of VMs.

4.2 PSO

INTERNATIONAL JOURNAL FOR RESEARCH IN APPLIED SCIENCE AND ENGINEERING TECHNOLOGY (IJRASET)

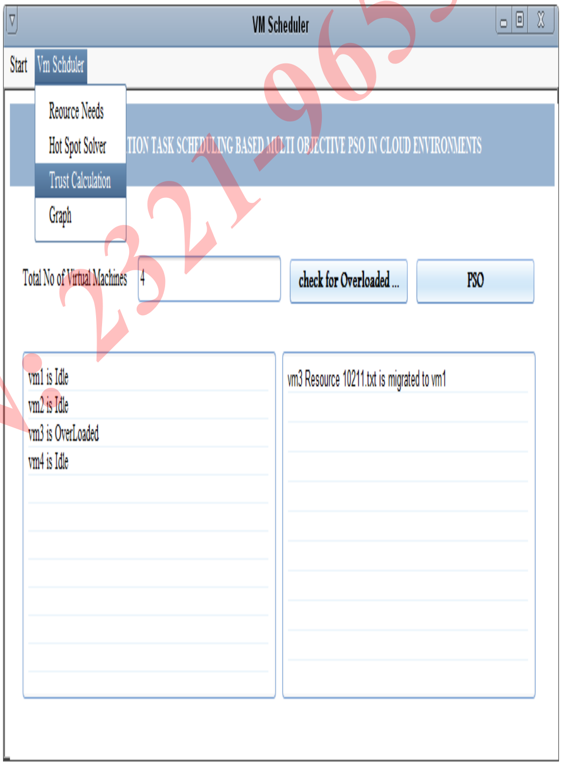
The HBB load balancing model has not assign task to proper virtual machine and does not consider the Quality of Service. A population of candidate solutions and particles are moved around in the search-space according to a few simple formulae. The movements of the particles are guided by their own best known position in the search-space as well as the entire swarm's best known Virtual machine based HBB load. When improved positions are being discovered these will then come to guide the movements of the swarm. The process is repeated and by doing so it is hoped, but not guaranteed, that a satisfactory solution will eventually be discovered.

4.3 Migration

Our algorithm executes periodically to evaluate the resource allocation status based on the predicted future resource demands of VMs. We define a server as a hot spot if the utilization of any of its resources is above a hot threshold. This indicates that the server is overloaded and hence some VMs running on it should be migrated away. The temperature of a hot spot reflects its degree of overload. If a server is not a hot spot, its temperature is zero. We define a server as a cold spot if the utilizations of all its resources are below a cold threshold. This indicates that the server is mostly idle and a potential candidate to turn off to save energy

4.4 Trust Allocation

The physical machines provide a set of virtual machines which are configured dynamically according to user requests. When the limited physical machines are provided to users from a pool of resources, the provided resources have two types; one is the dedicated resources and the other is the undedicated resources to give some extra margin in case of sudden request In this Cloud system environment, if a new user requests resources when all of the resources are already assigned, then the undedicated resources allocated to others are provided to the new users via dynamic reconfiguration. Here we calculate the trust model based on the historical information. By analyzing the vm load and the load of resource the trust value is calculated.



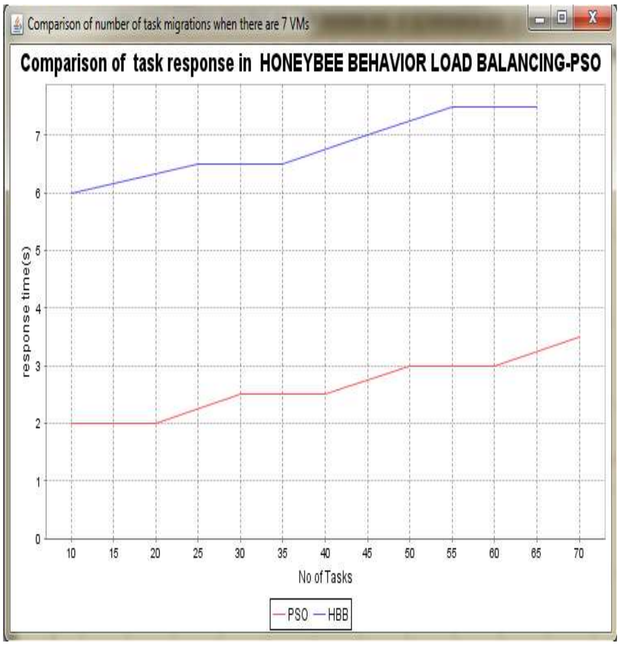
Apply PSO algorithm

5 RESULT AND DISCUSSION

The proposed algorithm is more efficient and reduces the response time and total task completion time compared to honey bee inspired algorithm. Particle swarm optimization is more efficient than honey bee inspired algorithm. Response time and make span are the main parameters in honey bee algorithm. By comparing with PSO algorithm, PSO is more efficient.

Vol. 2 Issue IV, April 2014
ISSN: 2321-9653

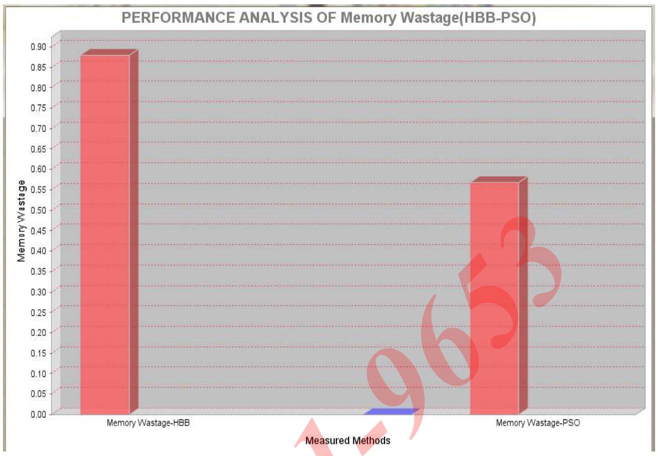
INTERNATIONAL JOURNAL FOR RESEARCH IN APPLIED SCIENCE AND ENGINEERING TECHNOLOGY (IJRASET)



Comparison of task response timeusing HBB-LB and PSO

Makespan can be defined as the overall task completion time.Wedenote completion time of task Ti on VMj as CTij. Makespan the time difference between the start and finish of a sequence of jobs or tasks.Response time is the amount of time taken between submission of a request and the first response that is produced. The reduction in waiting time is helpful in improving responsiveness of the VMs. In a data system, the system response time is the interval between the receipt of the end of transmission of an inquiry message and the beginning of the transmission of a response message to the station originating the inquiry.

By comparing the memory usage in Honey Bee Inspired Load balancing algorithm and PSO algorithm, PSO algorithm consumes less memory space.



performance analysis of memory usage

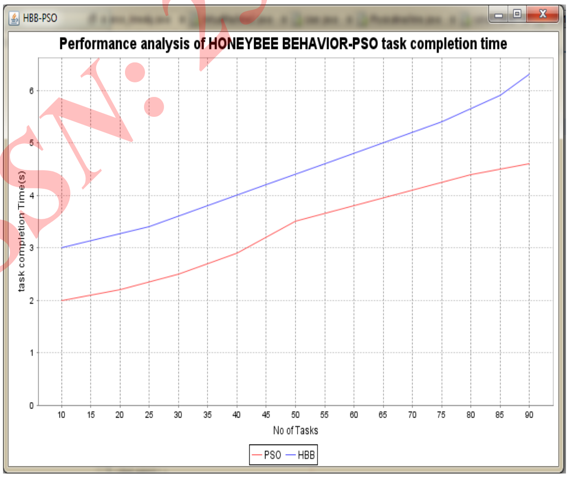


Fig 6.3: task completion time comparison using honeybee-pso

By analyzing all the above graph pso algorithm is more effective than honey bee inspire algorithm.

6.CONCLUSION

There are various algorithms designed for balancing the load among different tasks. Most of the load balancing

Vol. 2 Issue IV, April 2014
ISSN: 2321-9653

INTERNATIONAL JOURNAL FOR RESEARCH IN APPLIED SCIENCE AND ENGINEERING TECHNOLOGY (IJRASET)

algorithms is complex, and not able to implement. Honey bee Inspired Load Balancing algorithm overcomes this drawback. Main disadvantage of honeybee algorithm is, it assigns the tasks in first come manner. i.e it won't check all the free virtual machine. It will directly assign to the first virtual machine. Sometimes there is wastage of memory in virtual machine. Proposed algorithm overcomes the disadvantages of honey bee inspired algorithm. The main disadvantages of load balancing using an algorithm Honey bee Inspired Load Balancing of tasks is, it assigns the tasks in first come manner. i.e it won't check all the free virtual machine. It will directly assign to the first virtual machine.In order to overcome the drawback of honeybee algorithm another algorithm called PSO(particle swarm optimization) algorithm. In PSO algorithm task will assign to the virtual machine in best fit manner. i.e task will check all the virtual machine and assigns the task to proper virtual machine which will have least memory wastage.

7.REFERENCES

A. Revar, M. Andhariya, D. Sutariya, M. Bhavsar, Load balancing in grid environment using machine learning innovative approach, International Journal of Computer Applications 8 (10 (Oct)) (2010) 975–8887.

A.M. Bernardino, E.M. Bernardino, J.M. Sánchez-Pérez, M.A. Vega-Rodríguez, J.A. Gómez-Pulido, Efficient Load Balancing Using the Bees Algorithm, Trends Trends in Applied Intelligent Systems, Lecture Notes in Computer Science, Springer, Berlin, Heidelberg, 2011.

A.Singh, An artificial bee colony algorithm for the leaf constrained minimum spanning tree problem, Applied Soft Computing Journal 9 (2) .

B. Chang, H. Tsai, C.F. Huang, Z.Y. Lin, C.M. Chen, Fast access security on cloud computing: ubuntu enterprise server and cloud with face and fingerprint identification, in: Proceedings of the 2011 2nd International Congress on Computer Applications and

B. Yagoubi, Y. Slimani, Dynamic load balancing strategy for grid computing, transactions on engineering, Computing and Technology 13 (May) (2006) 260–265.

B. Yagoubi, Y. Slimani, Task load balancing strategy for grid computing, Journal of Computer Science 3 (3) (2007) 186–194.

B.R. Johnson, J.C. Nieh, Modeling the adaptive role of negative signaling in honey bee intraspecific competition, Journal of Insect Behavior 23 (2010) 459–471.Computing Research Repository, vol. abs/0903.2525, 2009.

D. Karaboga, An idea based on honey bee swarm for numerical optimization, Technical Report TR06, Computer Engineering Department, Erciyes University,Turkey, 2005.

D. Karaboga, B. Basturk, On the performance of artificial bee colony (ABC) algorithm, Applied Soft Computing 8 (1) (2008) 687–697.

D.L. Eager, E.D. Lazowska, J. Zahorjan, Adaptive load sharing in homogeneous distributed systems, The IEEE Transactions on Software Engineering (1986) 662–675.

E. Cuevas, D. Zaldívar, M. Pérez-Cisneros, H. Sossa, V. Osuna, Block matching algorithm for motion estimation based on Artificial Bee Colony (ABC), Applied Soft Computing (2012), http://dx.doi.org/10.1016/j.asoc.2012.09.020.

H. de Vries, J.C. Biesmeijer, Modelling collective foraging by means of in
Behavioral Ecology and Sociobiology 44 (1998) 109–124.

K. Li, G. Xu, G. Zhao, Y. Dong, D. Wang, Cloud task scheduling based on load balancing ant colony optimization, in: Chinagrid Conference (ChinaGrid), 2011.

K. Mukherjee, G. Sahoo, Mathematical model of cloud computing framework using fuzzy bee colony optimization technique, in: Proceedings of the 2009

N. Karaboga, M.B.C. etinkaya, A novel and efficient algorithm for adaptive filtering: artificial bee colony algorithm, Turkish Journal of Electrical Engineering & Computer Sciences Vol. 19 (2011) 175–190.

P. VenkataKrishnab,DhineshBabu L.D.," Honey bee behavior inspired load balancing of tasks in cloud computing(2013)