Hybrid task scheduling approach using Gravitational and ACO search algorithm

Er. Partibha Rani [1], Er. Akhilesh K. Bhardwaj[2]

[1]M.Tech. Student, Computer Science & Engineering, Shri Krishan Institute of Engineering & Technology, Kurukshetra, Haryana, India
[2]Assistant Professor, Computer Science & Engineering, Shri Krishan Institute of Engineering & Technology, Kurukshetra, Haryana, India

---***---

Abstract - *Due to flexibility and timely availability of desired resources, cloud computing is getting more and more popular day by day. Easy to use and anywhere access like potential of cloud computing has made it more attractive relative to other technologies. This has resulted in reduction of deployment cost on user side. It has also allowed the big companies to lease their infrastructure to recover the installation cost for the organization. It has inherited some default issues from its parent technologies. Load balancing is one of them. The main question in this context is connected to task scheduling. This paper offered a hybrid task scheduling approach named as ACGSA by combining the ant colony and gravitational search approach. Proposed approach has been tested using cloudsim simulator. The results of this proposed apparatus have compared with Ant colony optimization algorithm (ACO). It has shown better performance relative to basic approach on the relevant parameter.*

***Key Words*: Cloud Computing, Load Balancing, Cloudsim, Ant Colony Optimization, Gravitational Search**

1. INTRODUCTION

On demand computing is recognized as cloud computing considering as computing which is based on internet in which processing resources and computers are shared by administer and other devices on internet. Worldwide enabling of this model for appetite of computer resources is organized with the shared pool of data. Cloud computing provides users and enterprises with storage solutions, various capabilities like processing and storage of data in data centers of third party. Today, cloud computing is considered as a progressive area that can dynamically supply flexible services over the internet with the help of hardware and software virtualization. Nowadays, the new growing area is cloud computing, which adapt services that are dynamically delivered over internet through hardware and software virtualization on demand. The biggest advantage of cloud computing is flexible lease and release of resources as per the requirement of the user. It means that cloud computing is a kind of business activity where the consumers need to pay only for their subscriptions. Other benefits encompass betterment in efficiency, compensating the costs in operations. It curtails down the high prices of hardware and software. [1]

In cloud, load balancing is a technique that distributes dynamic load across multiple systems to ensure no single system is overloaded. It helps in excellent utilization of resources and improves the execution of the system. The main goal of load balancing is to reduce the resource consumption that will further minimize the carbon emission rate and energy consumption rate that is an extreme need of cloud computing. It avoids a situation in a cloud where some systems are free or have no burden in work while others are heavily loaded. High consumer ease and resource usage rate is achieved by using load balancing, hence overall performance and resource utilization of system is improved. It is a technique that provides the huge dynamic local workload across all the Nodes. By using load balancing a high consumer ease and resource usage rate can be achieved, be assured that all nodes are balanced, hence maintaining the total execution of the system. It can help in optimally utilizing the available resources, thus minimizing the resource consumption. Load balancing also helps in performing failover, allowing scalability, declining bottlenecks and over-provisioning, decreasing response time etc [2].

The rest of this paper has organized as follows: Section II discusses the basics of load balancing. Simulator introduction has discussed in section III. Related work has discussed in section IV. Proposed hybrid load balancing approach is given in section V. Results and analysis have been given in section VI. Conclusion and enhancement scope is given in section VII.

2. LOAD BALANCING BASICS

Load balancing can be defined as the process of task distribution among multiple computers, processes, disk, or other resources in order to get optimal resource utilization and to reduce the computation time. Load balancing is an important means to achieve effective resource sharing and utilization [3]. In general, load balancing algorithms can be divided into following three types:

- **Centralized approach:** In this approach, a single node is responsible for managing the distribution within the whole system.
- **Distributed approach:** In this approach, each node independently builds its own load vector by collecting

the load information of other nodes. Decisions are made locally using local load vectors. This approach is more suitable for widely distributed systems such as cloud computing.

- **Mixed approach**: A combination between the two approaches to take advantage of each approach [4].

3. CLOUDSIM SIMULATOR

Cloudsim is free and open source software available at http://www.cloudbus.org/CloudSim/. It is a code library based on Java. This library can be directly used by integrating with the JDK to compile and execute the code. To develop and test the applications quickly, Cloudsim is joined with Java-based IDEs (Integrated Development Environment) including Eclipse or NetBeans. Using Eclipse or NetBeans IDE, the Cloudsim library can be accessed and the cloud algorithm can be implemented. The Cloudsim library is used for the simulation of the following operations:

- Large scale cloud computing at data centers
- Virtualized server hosts with customizable policies.
- Support for modeling and simulation of large scale cloud computing data centers.
- Support for modeling and simulation of virtualized server hosts, with customizable policies for provisioning host resources to VMs.
- Support for modeling and simulation of energy-aware computational resources.
- Provide facility of simulation and modeling of data centre network topologies and message-passing applications.
- Support for modeling and simulation of federated clouds.
- Support for dynamic insertion of simulation elements, as well as stopping and resuming simulation.
- Support for user-defined policies to allot hosts to VMs, and policies for allotting host resources to VMs.
- User-defined policies for allocation of hosts to virtual machines [5].

4. RELATED WORK

In distributed computing, load management is needed to allocate the dynamic native work equally among all the machines. It assists in attaining a great client fulfillment and resource consumption ratio by guaranteeing an effective and impartial distribution of every computing resource. Accurate load balancing helps in reducing resource usage, minimizing failure, allowing scalability and dodging jams, etc. In this section, a systematic analysis of current load balancing methods is discussed. This analysis accomplishes that all the current methods, chiefly emphasis on decreasing related overhead, service reaction time and refining performance, etc. A number of parameters are also recognized, & these are utilized to analyze the existing techniques.

- Jiang, Ji & Shen et al. in [6] introduced a hybrid PSO and GSA (HPSO–GSA) in which position updates of particles are based on PSO velocity and GSA acceleration. Economic emission load dispatch problems were the main issue. HPSO-GSA gives better performance compared to PSO and GSA.
- Han et al. in [7] introduced a QBGSA–K-NN algorithm in which they combined the quantum inspired Binary GSA (QBGSA) with K-nearest neighbor (K-NN) method with leave-one-out cross validation (LOOCV). It is used for improving classification accuracy with an appropriate feature subset in binary problems. It selects the discriminating input features correctly and achievesa high classification accuracy.
- Sombra et al. in [8] introduced a new GSA in which changes were applied in alpha parameter throughout the iterations. It helps in improving performance of GSA. It helps to achieve better convergence.
- Gu & Pan in [9] proposed a modified a GSA (GSA &PSO) in which PSO features of saving previous local optimum and global optimum solutions are implemented into GSA. The particle memory ability in GSA is modified to remember its own local optimum and global optimum solutions in the updating process. It gives better performance than SVM and SGSA-SVM in classification accuracy and feature selection ability.
- Sharma in [10] has discussed the concept of gravitational search in task scheduling for cloud computing. Concept of mass and gravity was used in task scheduling. The algorithm includes a collection of searcher agents that communicate with others via the weight force. The agents are denoted as things and their operation is measured by their masses. The gravity force causes a complete united attempt to get something done where all things move towards other things with greater weight masses. The slow attempt of greater weight masses gives good results
- The Ant Colony System algorithm is an example of an Ant Colony Optimization method from the field of Swarm Intelligence, Metaheuristics and Computational Intelligence. Ant Colony System is an extension to the Ant System algorithm and is related to other Ant Colony Optimization methods such as Elite Ant System, and Rank-based Ant System [11].
- L.Agostinho, G. Feliciano et. al in [12] implements a cloud scheduler, based on Ant Colony Optimization (ACO) to allocate VMs to the physical resources belonging to a cloud. Experiments were carried out using Cloudsim and comparisons had been done with alternative cloud schedulers including random a scheduler based on genetic algorithms. The scheduler allows fair assignment of VMs and delivers competitive performance with re spect to the number of executed jobs per user.
- Pandey S et al. in [13] have described that the Particle Swarm Optimization technique for Cloud Computing

which is being inspired by the movement of birds or fishes in school. In PSO, each single solution is a "bird" in the search space which is called as "particle". Fitness values of all the particles have evaluated by the fitness function to be optimized, and have velocities which express the flying of the particles. In each iteration, each particle is updated by following two "best" values pbest and gbest and these two values are need not be evaluated in Ant Colony Optimization algorithm. Moreover ACO is more applicable for problems that require crisps results and PSO is applicable for problems that are fuzzy in nature.

- Kumar Nishant et al. in [14] have described that an Ant Colony Optimization has been connected from the point of view of cloud with the primary point of Load Balancing of nodes. This modified algorithm has an edge over the first approach in which every ant constructs their own particular individual result set and it is later on assembled into a complete arrangement. Further, as it is realized that a cloud is the gathering of numerous nodes, which can strengthen different sorts of application that is utilized by the customers on a premise of pay for each utilization. In proposed approach the ants constantly update a single result set rather than updating their own result set. As cloud support various types of application that is used by the clients on a basis of pay peruse. Therefore, such a system should function smoothly and should have algorithms that can continue the proper system functioning even at peak usage hours

- Mishra Ratan et al. in [15] has shown that the Ant Colony Optimization technique which is a Load Balancing technique is based on Swarm Intelligence. The motive of this paper is to produce an effective Load Balancing algorithm using Ant Colony Optimization technique to enhance or decrease different performance parameters like CPU load, Memory capacity, Delay or network load for the clouds of different sizes. They have also explained the working of a load balancer that how it works and what the various phases of the load balancer. In this paper, a heuristic algorithm is proposed which is dependent on Ant Colony Optimization.

- T. Liao et al [16] proposed a new approach for Ant colony optimization. ACO is a probabilistic technique for solving computational problems. They used ACO in cloud and grid computing task scheduling, etc, but it doesn't get a good performance since there are still some problems in pheromone update and the parameters selection. However, PACO also has some problems, such as the selection of the parameters and the way getting pheromone. In order to let Ant colony optimization get a better performance, a self adaptive ant colony optimization has be proposed in this paper which improves PACO.

- C. Y Liu [17] represented a task scheduling based strategy on genetic ant colony algorithm in cloud computing was proposed, which used the global searchability of GA to find the optimal solution, and converted to the initial pheromone of ACO. Task scheduling problem in cloud computing environment is NP-hard problem, which is difficult to obtain exact optimal solution and is suitable for using intelligent optimization algorithms to approximate the optimal solution. Meanwhile, quality of service (QoS) is an important indicator to measure the performance of task scheduling.

- X. F. Liu et al [18] proposed that Cloud computing resources scheduling is essential for executing workflows in the cloud platform because it relates to both the execution time and execution costs. For scheduling T tasks on R resources, an ant in ACS represents a solution with T dimensions, in solving the problem of optimizing the execution costs while meeting deadline constraints, they developed an efficient approach based on ant colony system (ACS). Compare the results with those of particle swarm optimization (PSO) and dynamic objective genetic algorithm (DOGA) approaches.

- Jinhua Hu et al [19] gave explanation about the current virtual machine(VM) resources scheduling in cloud computing environment mainly considers the current state of the system but seldom considers system variation and historical data, which always leads to load imbalance of the system. In view of the load balancing problem in VM resources scheduling, this paper presents a scheduling strategy on load balancing of VM resources based on genetic algorithm.

- U. K. Rout et al [20] A novel Economic Operation of Two Area Power System through Gravitational Search Algorithm has been proposed, which is based on the AGC. This method is explained with an example and the result obtained by the proposed method is compared with by particle swarm optimization as reported in literature. It has been shown that this method is more efficient and takes less computation time than PSO. The economic operation of two area (Grid) power system by GSA is presented.

- P. K. Roy [21] represented a Meta heuristic method for optimization known as GSA In this paper GSA is implemented to economic operation of a two area power system and computes how much power has to be generated internally in an area and how much power has to be borrowed from other area through tie-line for a specified load in most economical sense.

- G -B. Wang et al [22] proposed there are many researches on VM scheduling strategy. One well-known strategy is centralized VM live migration scheduling scheme. The scheduler mainly consists of two components: central controller and local migration controller. The central controller obtains all the physical resources utilization on the whole, and then initializes the VM migration based on pre-specified policies so as to achieve load balancing and high resource utilization.

5. PROPOSED WORK

In the proposed approach, authors have proposed a new load balancing approach for cloud environment by combining the ant colony optimization and gravitation search approach.

Basic ACO

In this approach, natural behavior of ants is used to identify the best task machine mapping. While searching the food, ant leaves the pheromone and other ants follow the path based upon the concentration of pheromone. Similar to that in task scheduling, foraging and trailing pheromone are used to convey overloaded and under loaded machines. Limitations of this approach was associated extra overhead in terms of pheromone and chances of local convergence [23].

Basic Gravitation Search Approach

Idea of gravitational attraction was used in implementing the scheduling approach. Newton's formula of gravitation force between two object is used in scheduling approach.
$F=6.67*m_1*m_2/R^2$, here m_1 and m_2 are the weight of two objects and R is representing the distance between those two objects.
In cloud task scheduling based upon the task completion time of a machine, a force value is assigned to every machine and it attract task as per the force value [24].

Proposed Hybrid Gravitational & ACO approach

1. In the beginning, all virtual machine and tasks are configured. This includes the setting of number of physical machines, no of virtual machines, ram capacity, secondary storage capacity, no of processors and their capacity. Task configuration includes the size of task, arrival time and specific resource requirements.
2. After this concept of ACO is used. In the beginning when no pheromone has been placed, random task machine mapping strategy is used.
3. Based upon the capacity of machines different pheromones are placed at different positions.
4. When scheduler is searching the best pheromone, which path to be followed is decided using gravitation search strategy.
5. A path from pheromone X to pheromone Y should be their iff attracting force of pheromone Y is more than pheromone X. Attracting force of pheromone is calculated on the basis of gravitation search strategy.
6. Best pheromone is selected and on that basis mapping of task with machine is done.
7. Go back to step 4 if any task is pending.

Flowchart of proposed approach has been shown below in figure 1.

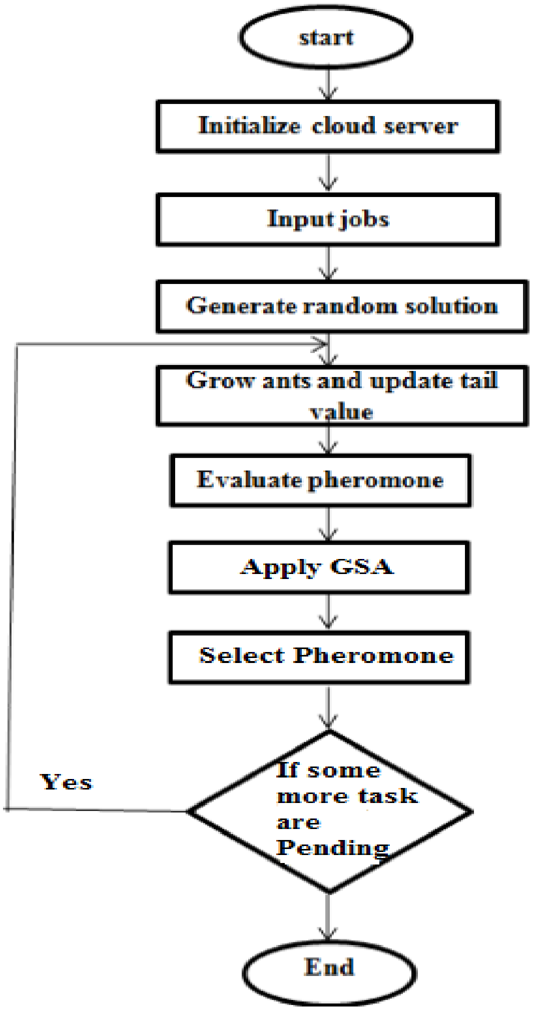

Fig-1: Proposed Hybrid ACO & Gravitational Approach

```
Algorithm HG_ACO( )
{
    Identify the host in the data center & store their
    number in the variable N.
    For (i=1 to N)
    {
        Initialize the processing capacity in MIPS,
        RAM, Data Storage & No of processing
        cores to machine $M_i$.
    }
    Generate an arbitrary set of tasks and store their
    number in the variable M.
    For (i=1 to M)
    {
        Initialize the task size, its arrival time,
        and resource requirement for task $t_i$.
    }
    Generate random task machine mapping pair.
    While (pending task is there in the data center)
    {
            ▪ Pheromones are launched.
            ▪ Based upon the remaining capacity of
              nearest machines pheromones are
              updated.
            ▪ Using Gravitation search algorithm, force
              of each pheromone is calculated.
            ▪ An edge from lower power pheromone to
              higher power pheromone is created
            ▪ While searching machine for a task,
              pheromone path is traced and wherever it
              ends task is mapped with that machine.
        }
}
```

6. RESULTS & ANALYSIS

Authors have tested the proposed ACO & Gravitational based hybrid approach and compare it with basic ACO approach in the simulation environment developed by cloudsim simulator [25].

Different parameters on which performance of the proposed approach has been compared with the existing one are as follows:

Makespan: It denotes the total schedule length. Lower value of this parameter is desirable.

Average Resource Utilization: It denotes up to what extent of makespan, resource was in usage. Higher value of this parameter is desirable.

Load balancing %age: This parameter denotes how efficiently tasks are distributed among different resources. Higher value of this factor is desirable.

Comparison of proposed ACO & Gravitational based hybrid approach with the basic ACO task scheduling approach on the above discussed parameter in the cloudsim [25] based simulation environment are as follows:

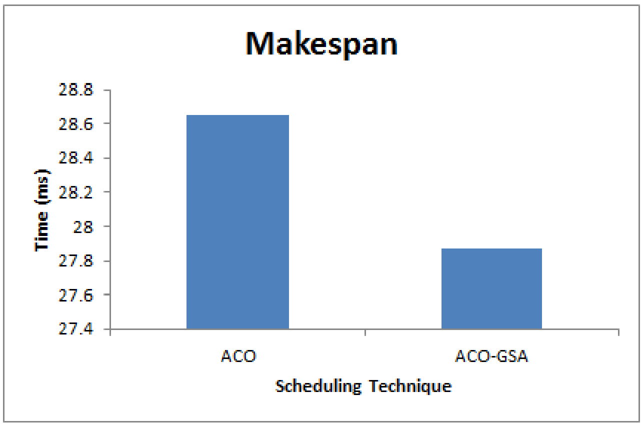

Chart-1: Comparison of Hybrid with basic approach on the scale of Makespan

Chart 1 shows the comparison of Hybrid ACO & GSA with basic ACO on the scale of makespan. It has been identified that Hybrid ACO & GSA is showing better makespan relative basic ACO.

Chart 2 shows the comparison of Hybrid ACO & GSA with basic ACO on the scale of resource utilization. It has been identified that Hybrid ACO & GSA is showing better resource utilization relative basic ACO.

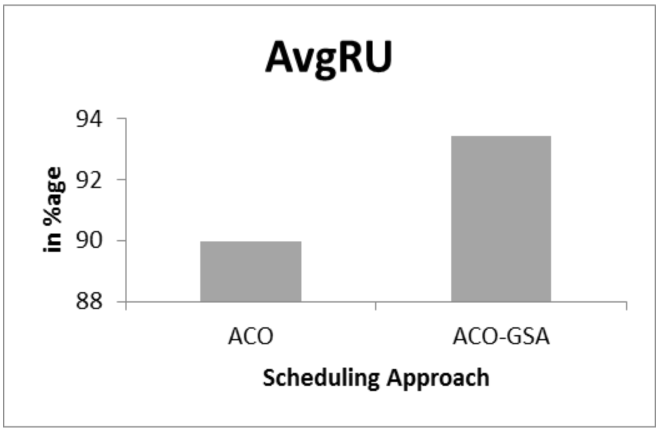

Chart-2: Comparison of Hybrid with basic approach on the scale of Resource utilization

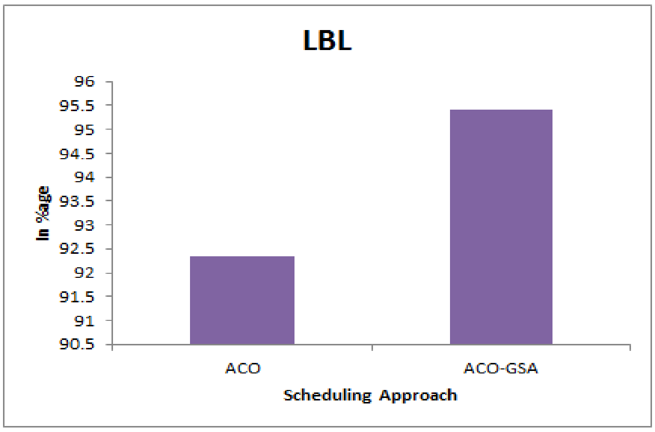

Chart-3: Comparison of Hybrid with basic approach on the scale of Load balance %age

Chart 3 shows the comparison of Hybrid ACO & GSA with basic ACO on the scale of load balance %age. It has been identified that Hybrid ACO & GSA is showing better distribution of load relative basic ACO. This has justified the result shown in chart 1 and 2.

7. CONCLUSIONS

Hybrid ACO & GSA approach has shown the better results relative to basic ACO on all the relevant parameter. By combining the GSA with ACO, it gets rid of local convergence limitation of ACO. This has been reflected through less makespan, better resource utilization and higher load balancing of hybrid approach relative to basic ACO approach. Moreover, proposed approach is distributed in nature. This makes it best suited for distributed environment like cloud. As a future scope authors have planned to test it in real cloud environment and test it on more parameters by comparing it with more task scheduling approaches.

REFERENCES

[1] A. Jain and R. Kumar, "A Taxonomy of Cloud Computing," International Journal of Scientific and Research Publications. vol. 4(7), Jul. 2014, pp. 1-5.

[2] P. Sasikala, "Cloud computing: Present Status and Future Implications," International Journal of Cloud Computing, vol. 1, no. 1, 2011, pp. 23-36.

[3] A. Jain and R. Kumar, "A multi stage load balancing technique for cloud environment," 2016 International Conference on Information Communication and Embedded Systems (ICICES), Chennai, 2016, pp. 1-7.doi: 10.1109/ICICES.2016.7518921.

[4] A. Khiyaita, M. Zbakh, H. El Bakkali, and D. El Kettani, "Load balancing Cloud Computing: State of Art," National Days of Network Security and Systems (JNS2), pp. 106-109, Apr. 2012.

[5] R. Buyya, R. Ranjan and R.N. Calheiros, "Modeling and Simulation of Scalable Cloud Computing Environments and the CloudSim Toolkit: Challenges and Opportunities," In Proceedings of the 7th High Performance Computing and Simulation Conference, pp.1-11, June 2009.

[6] Gu and F. Pan, "Modified Gravitational Search Algorithm with Particle Memory Ability and its Application," International Journal of Innovative Computing, Information and Control, vol. 9, pp. 4531–4544 2013.

[7] K. Das, "Cloud computing simulation," PhD diss., Indian Institute of Technology, Bombay Mumbai, 2011.

[8] K. Dasgupta, B. Mandal, P. Dutta, J.K. Mandal, and S. Dam, "A genetic algorithm based load balancing strategy for cloud computing," Procedia Technology, pp. 340-347, Jan 2013.

[9] L.SV. Singh, J. Ahmed, "A Greedy Algorithm For Task Scheduling & Resource Allocation Problems In Cloud Computing," International Journal of Research & Development in Technology and Management Science–Kailash 21, 2014.

[10] Sharma, Aakanksha. "Differential evolution-gsa based optimal task scheduling in cloud computing" International Journal of Engineering Sciences & Research Technology 1, no. 5: 1447-1451.

[11] Jason Brownlee "Clever Algorithms: Nature-Inspired Programming Recipes" http://www.cleveralgorithms.com/nature-inspired/swarm/ant_colony_system.html

[12] L. Agostinho, G. Feliciano, L. Olivi, E. Cardozo, E. Guimaraes, "A Bio-inspired Approach to Provisioning of Virtual Resources in Federated Clouds", Ninth International Conference on Dependable, Autonomic and Secure Computing (DASC), DASC 11, IEEE Computer Society, Washington, DC, USA, 2011.

[13] Pandey S, Wu L, Guru S, Buyya R. "A particle swarm optimization-based heuristic for scheduling workflow applications in Cloud Computing environments", International conference on advanced information networking and applications, 2010.

[14] Kumar Nishant, Sharma Pratik, Krishna Vishal, "Load Balancing of Nodes in Cloud Using Ant Colony Optimization", 14th IEEE Conference on Modelling and Simulation, 2012.

[15] Ratan Mishra and Anant Jaiswal, "Ant colony Optimization: A Solution of Load balancing in Cloud", International Journal of Web & Semantic Technology, Vol. 3,2012.

[16] T. Liao, K. Socha, M. A. Montes de Oca, T. Stutzle, M. Dorigo, "Ant colony optimization for mixed-variable optimization problems," IEEE Transactions on Evolutionary Computation, vol 18, NO. 4, pp. 503-518, 2014.

[17] C. Y. Liu, "A task scheduling algorithm based on genetic algorithm and ant colony optimization in cloud computing," Proceeding of the 13th International Symposium on Distributed Computing and Applications to Business, Engineering and Science (DCABES), IEEE Press, Nov. 2014, pp.68-72.

[18] X. F. Liu, Z. H. Zhan, K. J. Du, and W. N. Chen, "Energy aware virtual machine placement scheduling in cloud computing based on ant colony optimization approach," in Proc. Genetic Evol. Comput. Conf., 2014, pp. 41-47.

[19] Jinhua Hu, JianhuaGu, Guofei Sun Tianhai Zhao "A scheduling strategy on load balancing of virtual machine resources in cloud computing environment",2010 IEEE 978-0-7695-4312-3/10.

[20] U. K. Rout., "Gravitational search algorithm based automatic generation control for interconnected power system", International Conerence on Circuits, Power and Computing Technologies (ICCPCT-2013), pp.558-563, Mar 2013.

[21] P. K. Roy, "Solution of unit commitment problem using gravitational search algorithm", International Journal of Electrical Power & EnergySystems, vol. 53, pp. 85-94, Dec 2013.

[22] G.-B. WANG, Z.-T. MA, L. SUN, "On load balancing-oriented distributed virtual machine migration in cloud environment," Computer Applications and Software, vol. 30, no. 30, pp. 1–2, 2013.

[23] S. Kumar, D. S. Rana and S. C. Dimri, " Fault tolerance and load Balancing Algorithm in Cloud Computing" International Journal of Advanced Research in Computer and Communication Engineering, Vol. 4(7), pp. 92-96, 2015

[24] B. Barzegar, A. M. Rahmani, K. Z. Far and A. Divsalar, "Gravitational Emulation local Search Algorithm for Advanced Reservation and Scheduling in Grid Computing Systems" in 4th International conference on Computer Sciences and Convergence Information Technology, IEEE 2009.

[25] R. N. Calheiros, R. Ranjan, A. Beloglazov, C. A. F. D. Rose, R. Buyya, "Cloudsim: a toolkit for modeling and simulation of cloud computing environments and evaluation of resource provisioning algorithms", Software: Practice and Experience, vol. 41, no. 1, pp. 23 – 50,2011.