Sub-Sahara African Academic Research Publications
June, 2021, Editions

JPAS
Journal of Pure and Applied Science
Vol..20, No.9, ISSN 2318-9315

A TASK SCHEDULING ALGORITHM BASED ON IMPROVED PSO LOAD BALANCING IN CLOUD COMPUTING ENVIRONMENT

1SABO ABDULHAFIZ, 2ISHAQ MUHAMMED, 3*HABIB SHEHU JIBRIL*
**Department of Mathematical sciences, Bauchi State University Gadau, **Computer Science Department, Federal Polytechnic Bauchi, ***Mathematical Sciences Department, Abubakar Tafawa Balewa University Bauchi*

Abstract
Cloud computing is emerging as a new paradigm that provide computing resources in a pay-as-you go model. Users can get satisfying services via cloud computing. The major challenges facing the cloud environment is the issue of load balance the work load on virtual machine on the cloud that are serving various user. Many researchers consider execution time to be the major factor when scheduling the cloud resources as task running time and system resource utilization which trigger more issues in load balancing. In order to solve this problem, this research proposes a model based on PSO to improve execution time in view of both task running time and system resources.

Keywords: Cloud Computing: Load balancing: PSO

Introduction
Cloud computing is emerging as a new paradigm that provide computing resources in a pay-as-you go model. The cloud computing is a combination of technologies where a large number of systems are connected in private or public networks [1]. The major potential of the cloud comes from ability to provide anything as service "Xaas". "Xaas" means one or more of these services such that Software as a Service (SaaS), Infrastructure as a Service (IaaS), and

Platform as a Service (PaaS). Due to the exponential growth of cloud computing it has been widely adopted by the industry and thus making a rapid expansion in availability of resource in the Internet. The aim of cloud computing environment is to optimally use the available computing resources [2]. Virtualization greatly helps in valuable utilization of resources and builds an effective system. Customers use cloud computing in the form of services like Gmail, Face book, YouTube, Yahoo, Hotmail etc. Thus in spite of glorious future of Cloud Computing, many critical problems still need to be explored for its perfect realization [3]. One of these issues is task scheduling.

The scheduling of tasks means an optimal usage of available resources. The main purpose of scheduling is to achieve the high performance, reduce the waiting time, increase system throughput and so on. Task scheduling is a challenging issue in cloud computing because it is parallel and distributed architecture. The task completion time determination is difficult in cloud because the tasks may be distributed between more than one Virtual machine. Virtual CPUs are assigned to each virtual machine. An effective task scheduling method requires not only meeting the user's needs but also improve the performance of the whole system. There have been few researches on task scheduling techniques in cloud computing environment.

Thomas et al. [4] presented scheduling as major task in a cloud computing environment. In Cloud computing environment datacenters take care of this task. The scheduling algorithm has been analyzed which were based on user priority and task length. The three cloud scheduling scenarios have been discussed. First scenario is based on the length of tasks. The second scenario is based on task priority. The third and the proposed approach works on both cloudlet priority and cloudlet length.

Al-Maamari and Omara [5] discussed that task scheduling is the most significant matter in the cloud computing because the user has to pay for resource using on the basis of time, which acts to distribute the load evenly among the system resources by maximizing utilization and reducing task execution Time. The work implemented a Dynamic Adaptive Particle Swarm Optimization algorithm (DAPSO) to enhance the performance of the basic PSO algorithm to optimize the task runtime by minimizing the makespan of a particular task set, and in the same time, maximizing resource utilization. Also a task scheduling algorithm has been proposed to schedule the independent task over the Cloud Computing.

Mandal and Acharyya [6] explained that task scheduling issue is mainly focus on to find the best or optimal resources in order to minimize the total processing time of Virtual Machines (VMs).The focus is on increasing the efficient use of the shared resources. The work three meta-heuristic techniques such as Simulated Annealing, Firefly Algorithm and Cuckoo Search Algorithm have been implemented to find an optimal solution. The study of these algorithms is to minimize the overall processing time of the VMs which execute a set of tasks.

Mehranzadeh and Hashemi [7] presented cloud computing one host as component that represents a physical computing and a datacenter is composed by a set of hosts, which are responsible for managing Virtual Machines during their life cycles. An important issue in cloud computing is the scheduling of virtual machines requests so that the requested tasks can be completed in a minimum time according to the user defined time and compared various scheduling techniques First come first serve (FCFS) and Round Robin (RR).

Ant Colony Optimization (LBACO). Vijayalakshmi et al. [8] described that allocating the resources efficiently is a challenging job. Service providers need to ensure that their resources are utilized properly. The new scheduling algorithm in this make an address the challenge of task scheduling in cloud. The user tasks are prioritized. Based on the priority, the tasks are assigned to the Virtual Machines. The task with highest priority is assigned to a Virtual Machine with highest processing power and minimize the execution time.

Sun et al. [9] presented priority-based task scheduling algorithm (P-TSA) in grid. In this kind of priority-based algorithm, tasks are scheduled according to the priority order firstly. And then assign processors. The work Comparing P-TSA with existed grid scheduling algorithms on scheduling length and resource utilization rates. The performance of P-TSA is better than other scheduling algorithms such as Min-min and Max-min.

In this paper MAPSO has been used to improve the SPSO algorithms, which uses the mechanism of mutation operator and a self adaptation inertia weight. CloudSim has been used for simulation and analysis of the algorithm. The performance of the algorithm is compared with SPSO. The rest of paper is organized as follows. Section 2 proposes the MAPSO algorithm for task scheduling. Section 3 presents the simulation results and its analysis. Finally, Section 4 concludes this paper.

Basic Particle Swarm Optimization Algorithm.

In the basic particle swarm optimization algorithm, particle swarm consists of "n" particles, and the position of each particle stands for the potential solution in D-dimensional space. The particles change its condition according to the following three principles: (1) to keep its inertia (2) to change the condition according to its most optimist position (3) to change the condition according to the swarm's most optimist position. The position of each particle in the swarm is affected both by the most optimist position during its movement (individual experience) and the position of the most optimist particle in its surrounding (near experience). When the whole particle swarm is surrounding the particle, the most optimist position of the surrounding is equal to the one of the whole most optimist particle; this algorithm is called the whole PSO. If the narrow surrounding is used in the algorithm, this algorithm is called the partial PSO. Each particle can be shown by its current speed and position, the most optimist position of each individual and the most optimist position of the surrounding. In the partial PSO, the speed and position of each particle change according the following equality [10]

$$VX_i^{k+1} = w.vx_i^k + c_1.rand(\quad).(p_i^k - x_i^k) + C_2.rand(\quad).\left(f_g^k - x_i^k\right) \ldots\ldots\ldots\ldots(1)$$

$$vx_i^{k+1} = \begin{pmatrix} (vx_i^{k+1}) \leq vmax \\ vmax, else \end{pmatrix} \ldots\ldots\ldots\ldots(2)$$

In this equality, *kid v* and *k id x* stand for separately the speed of the particle "i" at its "k" times and the d-dimension quantity of its position; *k id pbest* represents the d-dimension quantity of the individual "i" at its most optimist position at its "k" times. *gbest* is the d-dimension quantity of the swarm at its most optimist position. In order to avoid particle being far away from the searching space, the speed of the particle created at its each direction is confined between -vdmax, and vdmax. If the number of vdmax is too big, the solution is far from the best, if the number of vdmax is too small, the solution will be the local optimism; c1 and c2 represent the speeding figure, regulating the length when flying to the most particle of the whole swarm and to the most optimist individual particle. If the figure is too small, the particle is probably far away from the target field, if the figure is too big, the particle will may be fly to the target field suddenly or fly beyond the target field. The proper figures for c1 and c2 can control the speed of the particle's flying and the solution will not be the partial optimism. Usually, c1 is equal to c2 and they are equal to 2; r1 and r2 represent random fiction, and 0-1 is a random number. In local PSO, instead of persuading the optimist particle of the swarm, each particle will pursuit the optimist particle in its surrounding to regulate its speed and position. Formally, the formula for the speed and the position of the particle is completely identical to the one in the whole PSO.

Proposed Algorithm

Inertia weights are put forward by [11]. An Inertia weight ω is a proportional agent that is related with the speed of last time, and the formula for the change of the speed is the following:

$$VX_i^{k+1} = w.vx_i^k + c_1.rand(\quad).(p_i^k - x_i^k) + C_2.rand(\quad).\left(f_g^{\ k} - x_i^k\right))\text{..........}(3)$$

The influence that the last speed has on the current speed can be controlled by inertia weights. The bigger ω is, the bigger the PSO's searching ability for the whole is, and the smaller ω is, the bigger the PSO's searching ability for the partial. Generally, ω is equal to 1, so at the later period of the several generations, there is a lack of the searching ability for the partial. Experimental results show that PSO has the biggest speed of convergence when ω is between 0.8 and 1.2. While experimenting, ω is confined from 0.9 to 0.4 according to the linear decrease, which makes PSO search for the bigger space at the beginning and locate the position quickly where there is the most optimist solution. As ω is decreasing, the speed of the particle will also slow down to search for the delicate partial. The method quickens the speed of the convergence, and the function of the PSO is improved. When the problem that is to be solved is very complex, this method makes PSO's searching ability for the whole at the later period after several generation is not adequate, the most optimist solution cannot be found, so the inertia weights can be used to work out the problem.

The proposed algorithm is as given below:

i. Initialize the particle swarm randomly.
ii. Evaluate the fitness function for each particle.
iii. Compare the optimization fitness value of each particle with its previous best position pbest, if the pbest is better than the previous pbest values, and then set the current pbest value as the best value of the particle.
iv. Compare the fitness function value of each particle with the previous gbest value, if the current value is better than the previous gbest values, and then reset the current value as the global best value of the swarm.
v. To all particles of particle swarm, execute the following operations:
    i. Update the position and velocity of particles by using equations (4).
    ii. Update the inertia weight by using equations (5).
vi. Check if a stop criterion is met. Fit is met, the execution is terminated. Otherwise, go to Step 2.

Simulation results and analysis

Cloudsim3.0.3 is an open source simulator which has been developed by Grid bus project team and the grid Laboratory of the University of Melbourne in Australia. The Cloudsim can run on Linux and Windows systems [12]. CloudSim has been used to implement the proposed MAPSO task scheduling algorithm. Also, a comparative study has been done to evaluate the performance of the proposed MAPSO, with respect to the SPSO algorithm [13]. This simulation mainly validates the advantage of the execution time and the resource utilization in the Cloud Computing environment.

Performance Results

To evaluate the performance of the two algorithms; original PSO and MAPSO, fifteen Virtual machines are considered with 50, 100, 150, 200, 250 and 300 cloudlets.

Tables 1 represent the execution time of SPSO and MAPSO algorithms using fifteen virtual machines and a set of various cloudlets respectively.

Table1. Completion Execution Time in sec Using 15 VMs

No. of Tasks	No. of VM	SPSO	MAPSO
50		2107.66	1610.434
100	15	2498	1717.408
150		2762.25	2027.655
200		3006.76	2180.052
250		3118.58	2294.558
300		3297.61	2402.806

According to the results in Figure1, the proposed MAPSO algorithm, with the respect to the execution time using 15 VMs, outperforms the default SPSO algorithms.

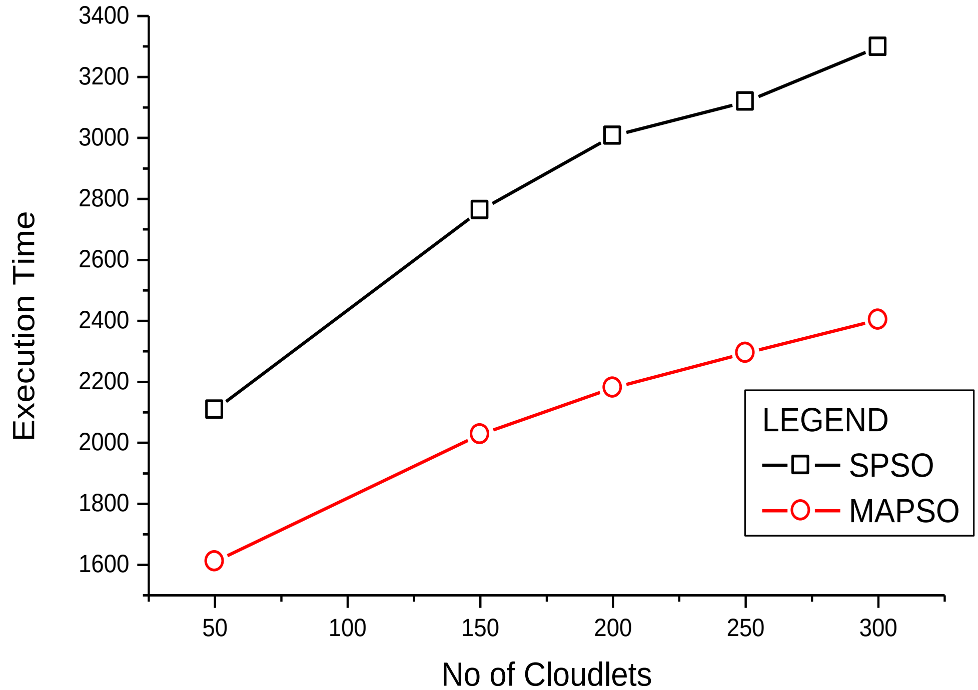

Figure1. The Execution Time of All Cloudlet when No. VMs (15)

The simulation results of the resource utilization of MAPSO and PSO algorithms using fifteen virtual machines and a set of various cloudlets respectively are described in Table2 and Figure2.

Table2. Resource Utilization using 15 VMs

No. of Tasks	No. of VM	SPSO	MAPSO
50		0.40124	0.43183
100	15	0.43064	0.46875
150		0.56298	0.61972
200		0.63243	0.72631
250		0.68519	0.80621
300		0.7229	0.89765

According to the results in Figure2, the proposed MAPSO algorithm, with the respect to the resource utilization using 15 VMs, outperforms the default SPSO algorithms.

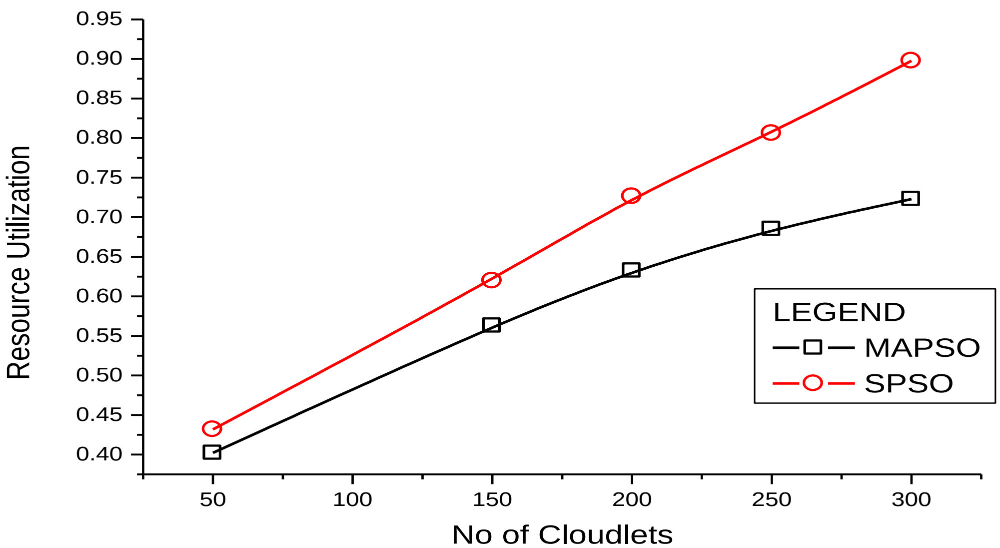

Figure2. Comparison Utilization of Number of cloudlet

Conclusion

In this paper, a novel task scheduling algorithm based on improved PSO for load balancing in Cloud Computing has been developed to provide minimum execution time and maximum utilization of resource in cloud environment. Analysis of the results indicates that the proposed MAPSO for load balancing outperforms the standard PSO algorithm.

References

[1] Kun Li, Gaochao Xu, Guangyu Zhao, Yushuang Dong, Dan Wang," Cloud Task scheduling based on Load Balancing Ant Colony Optimization", Sixth Annual China Grid Conference IEEE, Chang Chun, Liaoning,China,pp.3-9,2011.

[2] Antony Thomas, Krishnalal G, Jagathy Raj VP," Cried Based Scheduling Algorithm in Cloud Computing Enviorment", International Conference on Information and communication Technologies ,Elsevier ,Vol. 46,pp.913- 920,2015.

[3] A. Vouk, "Cloud computing- issues, research and implementations", in Proc. of Information Technology Interfaces, pp. 31-40, 2008.

[4] Antony Thomas, Krishnalal G, Jagathy Raj VP," Cried Based Scheduling Algorithm in Cloud Computing Enviorment", International Conference on Information and communication Technologies ,Elsevier ,Vol. 46,pp.913- 920,2015.

[5] Ali Al-maamari and Fatma A. Omara," Task Scheduling Using PSO Algorithm in Cloud Computing Environments", International Journal of Grid Distribution Computing, Vol.8,No.5, pp.245-256,2015.

[6] Tripti Mandal and Sriyankar Acharyya," Optimal Task Scheduling in Cloud Computing Environment: Meta Heuristic Approaches", International Conference on Electrical Information and Communication Technology, IEEE, pp.25- 28, 2015.

[7] Amin Mehranzadeh and Seyyed Mohsen Hashemi, "A Novel-Scheduling Algorithm for Cloud Computing based on Fuzzy Logic", International Journal of Applied Information Systems Foundation of Computer Science FCS, New York, USA,Vol.5,No.7, pp.2249-0868, 2013.

[6] Ratan Mishra and Anant Jaiswal, "Ant colony Optimization: A Solution of Load balancing in Cloud",in International Journal of Web & Semantic Technology (IJWesT), Vol.3, No.2, pp. 33–50, 2012.

[7] Eberhart R C, Shi Y. (1998). Comparison between genetic algorithms and Particle Swarm Optimization.Porto V W, Saravanan N, Waagen D, et al.Evolutionary Programming VII.[S.l.]:Springer, pp 1998:611-616.

[8] R. Vijayalakshmi, Mrs. Soma Prathibha," A Novel Approach for Task Scheduling in Cloud", Fourth International Conference on Computing, Communication, and Network Technologies, IEEE, chennai, India, pp.1-5,2013.

[9] Weifeng Sun, Yudan Zhu, Zhiyuan Su, Dong Jiao, Mingchu Li," A Priority-based Task Scheduling Algorithm in Grid", Third International Symposium on Parallel Architectures, Algorithms and Programming, IEEE Computer Society Washington, DC, USA,pp.311-315,2010.

[10] A.I. Awada, N.A. El-Hefnawy, H.M. Abdel_ kader,"Enhanced Particle Swarm Optimization For Task Scheduling In Cloud Computing Environments", International Conference on Communication Management and Information Technology, Elsevier,Vol.65,pp.920 – 929,2015.

[11] Nidhi Bansal, Amitab Maurya, Tarun Kumar, Manzeet Singh, Shruti Bansal,” Cost performance of QoS Driven task scheduling in cloud computing”, Third International Conference on Recent Trends in Computing, Elsevier,Vol.57,pp.126-130,2015.

[12] Liyun zuo, Lei shu shoubin dong, chunsheng zhu, and takahiro hara,” A Multi-Objective Optimization Scheduling Method Based on the Ant Colony Algorithm in Cloud Computing”, Computer Science and Engineering South China University of Technology, Guangzho China, IEEE,Vol.3, pp. 2687-2699,2015.

[13] Gaurang patel,Rutvik Mehta,Upandra Bhoi, ”Enhanced Load Balanced Min Min Algorithm For Static Meta Task Scheduling in cloud computing”, International Conference on Recent Trends in Computing Elsevier,Vol.57,pp.545- 553, 2015.

[14] A.I.Awada, N.A.El-Hefnawy, H.M.Abdel_kader,”Enhanced Particle Swarm Optimization For Task Scheduling In Cloud Computing Environments”, International Conference on Communication Management and Information Technology, Elsevier,Vol.65,pp.920 – 929,2015.

[15] Eakta Kumari and Monika,”Task Scheduling Algorithms in Cloud Computing:A Review”, International Journal of Science, Environment and Technology, Vol. 4, No. 2, pp.433-439,2015.

[16] Sonia Sindhu,”Task Scheduling in Cloud Computing”, International Journal of Advanced Research in Computer Engineering & Technology,Vol.4 No.6, pp.3019-3023, 2015.

[17] Eleonora Maria Mocanu, Mihai Florea, Mugurel Ionuţ Andreica, Nicolae Tapus, ”Cloud Computing –Task Scheduling based on Genetic Algorithms”, Computer Science Department, IEEE, Politehnica University of Bucharest, Bucharest, Romania,pp.1-6,2012.

[18] S.Sindhu, and Saswati Mukherjee, "Efficient task scheduling algorithms for cloud computing environment", High Performance Architecture and Grid Computing, Springer, Vol.169, pp.79-83, 2011.

[19] Amin Mehranzadeh and Seyyed Mohsen Hashemi,”A Novel-Scheduling Algorithm for Cloud Computing based on Fuzzy Logic”, International Journal of Applied Information Systems Foundation of Computer Science FCS, New York, USA,Vol.5,No.7,pp.2249-0868,2013.

[20]R. N. Calheiros, R. Ranjan, A. Beloglazov, C. A. De Rose and 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, pp. 23-50, 20011

[21] Eberhart R C, Shi Y. Comparing inertia weights and constriction factors in Particle Swarm Optimization. Proceedings of the Congress on Evolutionary Computating ,pp. 2000:84-88,2000