An Upgraded Algorithm of Resource Scheduling using PSO and SA in Cloud Computing

Talwinder Kaur
M.Tech Student (CSE)
SSIET, Dera Bassi
PTU, Jalandhar

Seema Pahwa
Professor, CSE Deptt.
SSIET, Dera Bassi
PTU, Jalandhar

ABSTRACT

Cloud computing is a subscription-based service whose primary benefit is application scalability which allows real-time provisioning of resources to meet application requirements. Scheduling is the most prominent issue in cloud computing. Generally the goal of scheduling is to properly dispatch parallel jobs to slave node machines according to different scheduling policies. In this paper previously existing algorithms i.e. Particle Swarm Optimization (PSO), Improved Particle Swarm Optimization (IPSO), Simulated Annealing (SA) Algorithm, and Hybrid Particle Swarm Optimization-Simulated Annealing based on utilization time are studied which were proposed to handle problems posed by network properties between user and resources. A new algorithm is designed using shortest path theory, Particle Swarm Optimization and Simulated Annealing technique which achieve the target consuming less average execution time to obtain more efficiency in resource utilization and minimize the cost of the processing.

Keywords

Scheduling, IPSO-Improved Particle Swarm Optimization, PSO-Particle Swarm Optimization, SA-Simulated Annealing.

1. INTRODUCTION

1.1 Cloud Computing

A 'cloud' is an elastic execution environment of resources involving multiple stakeholders and providing a metered service at multiple granularities for a specified level of quality (of service). Specifically a cloud is a platform or infrastructure that enables execution of code (services, applications etc.), in a managed and elastic fashion, whereas "managed" means that reliability according to pre-defined quality parameters is automatically ensured and "elastic" implies that the resources are put to use according to actual current requirements [1].

1.2 Scheduling

With the drastic increase in number of users, the stakeholders have to satisfy all of them at the same time. They must provide the users with resources in such a way that all resources are utilized in minimum possible time and cost effectively. This process is known as scheduling. There are number of algorithms in cloud computing developed by the researchers to schedule the tasks based on various parameters. Hence scheduling becomes an important issue in cloud computing.

1.3 Particle Swarm Optimization (PSO)

PSO is a swarm based intelligent algorithm. It is self adaptive global search optimization technique introduced by Kennedy and Eberhart [2][3]. The particle swarm paradigm is inspired by the social behaviour patterns of organisms that live and interact within large groups. It incorporates swarming behaviours observed in flocks of birds, school of fish, or swarms of bees, and even human social behaviour. It can be easily implemented and applied to solve various function optimization problems and, by extension, to problems that can be transformed into function optimization problems. As an algorithm its main strength is its fast convergence, which compares favourably with many other global optimization algorithms [2][4].

1.4 Simulated Annealing (SA)

Simulated Annealing is a random-search technique which exploits an analogy between the way in which a metal cools and freezes into a minimum energy crystalline structure and the search for a minimum in a more general system. It forms the basis of an optimization technique for combinatorial and other problems. It was developed in 1983 to deal with highly non linear problems [4]. Simulated Annealing is one of the most popular meta-heuristics providing a means to escape local optima by considering moves which worsen the objective function value known as jumping mechanism. Towards the end of computation, when the temperature or probability of accepting a worse solution, is nearly zero, this simply seeks the bottom of the local optima [4].

2. LITERATURE REVIEW

Shaobin Zhan et al. [2] proposed the improved particle swarm optimization algorithm in resources scheduling strategy of cloud computing. Through experimental results, author show that this method can reduce the task average running time, and raised the rate of availability of resources. Simulated annealing algorithm was added into PSO algorithm, and mixed scheduling algorithm was proposed, which not only increased the convergence and speed but also avoid sinking into local optima. The results indicated that improved particle swarm optimization algorithm shortens the average operation time of tasks, supplied proper resources to user task efficiently in the environment, increased utilization ratio of resources. **Amin Jamili** et al. [4] proposed a hybrid algorithm, namely PSO-SA, based on particle swarm optimization (PSO) and simulated annealing (SA) algorithms. Some randomly constructed instances were carried out to evaluate this proposed PSO-SA by which the related results were compared with the proposed SA and PSO algorithms as well as a branch-and-bound algorithm. In addition the results were compared with the hybrid algorithm embedded with electromagnetic-like mechanism and SA. The three lower bounds were studied, and the gap between the found LBs and the best found solutions were reported. The outcomes prove

that the proposed hybrid algorithm was efficient and effective tool to solve the PJSSP. **Suraj Pandey** et al. [3] presented that particle swarm optimization (PSO) based scheduling heuristic for data intensive applications that takes into account both computational cost and data transmission cost. Cost savings when using PSO as compared to using existing 'Best Resource Selection' algorithm was analyzed. It was shown that: a) as much as 3 times cost savings cost savings as compared to BRS, b) good distribution of workload onto resources, when using PSO based scheduling heuristic. **Sujit Tilak** et al. [5] surveyed the various existing scheduling algorithms in cloud computing and tabulated their various parameters along with tools. It was noticed that disk space management was critical in virtual environments. When virtual image was created, the size of disk was fixed. Having too small initial virtual disk size can adversely affect the execution of the application. Therefore it was concluded that there was a need to implement a scheduling algorithm that can improve the availability and reliability in cloud environment.

3. PRESENT WORK

3.1 Objective

- To study and analyze various scheduling algorithm in cloud computing.

- To design a new algorithm using Particle Swarm optimization based task scheduling algorithm and improve the average time taken.

- To implement the designed algorithm using Java Netbeans IDE 7.3.

- To compare the designed algorithm with existing algorithm.

3.2 Proposed Work

In this paper a new algorithm is proposed that consumes very less time for achieving the target of scheduling the job in cloud computing. First of all we need to select N initial solutions randomly. Then we need to calculate the center position of the target position and first randomly selected particle's position using the following formula:

Midpoint = (Present position of randomly selected particle + Target position)/2

Next we have to find out the nearest particle to the midpoint to initialize it as pBest. SA algorithm will be implemented for each particle to generate the gBest. Now if it is the end of the iteration, the gbest will the final output but in case it is not the end of iterations the velocity and position of every particle are need to be updated. To calculate the position and velocity of particles according to the pBest and gBest will be updated using the following two equations:

$$v_{id}(t+1)=wv_{id}(t)+c_1r_1(pBest_{id}-x_{id}(t)+)+c_2r_2(gBest_d-x_{id}(t))$$

$$x_{id}(t+1)=x_{id}(t)+v_{id}(t+1)$$

$pBest_{id}$= the best solution that can the ith particle has been explored so far, $gBest_d$= Best value of $pBest_{id}$ among the whole group or swarm of particles, i= particle, d= dimension of space, v= velocity of the particle, x= position of the particle, w= inertia weight. Its value is set to 1, c1 and c2= acceleration constant. It is a cognitive and social component which leads each particle towards $pBest_{id}$ and $gBest_i$ position. Its value is set to 2 according to PSO algorithm [2], r1 and r2= random values of particles in range [0, 1], t= iteration number.

The velocity and the position of particle will be updated in every iteration using the above equations. And hence we get the desired results.
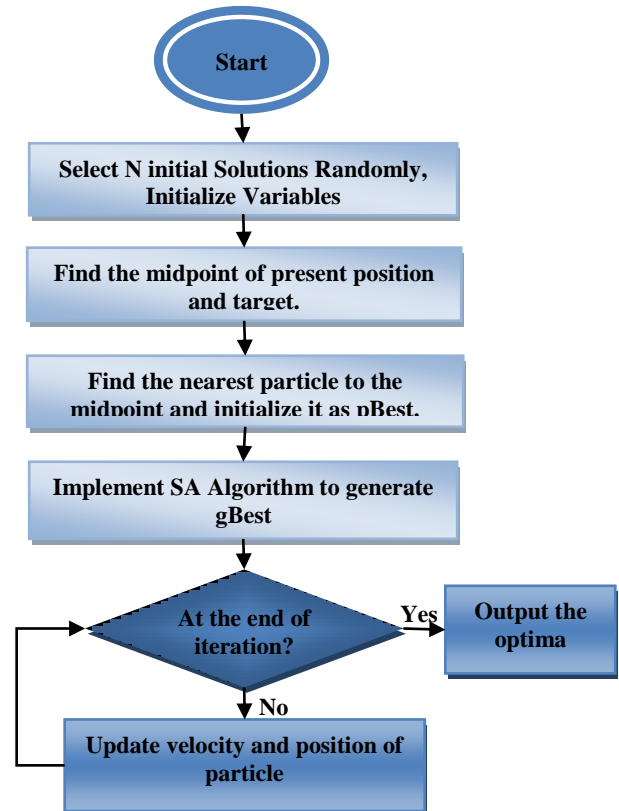

Fig 1: Flowchart of Proposed Algorithm

Algorithm for Shortest Path Mixed PSO-SA algorithm

Step 1. Randomly Initialize variables for PSO.

Step 2. Find the Center Position b/w the Present particle and target.

Center position = (Present position+Target)/2

Step 3. Find the nearest particle to the center position and initialize it as pBest.

Step 4. For each particle implement SA (Stimulated annealing) algorithm to generate gBest.

Step 5. If the end of iteration is reached, the optima will be returned otherwise next step will be followed.

Step 6. Update velocity and position according to pBest using PSO and repeat step 5.

4. RESULTS AND DISCUSSIONS

The average execution time to achieve the target is calculated by manually updating the number of particle and number of inputs of each particle by using both previously existing algorithm and the proposed algorithm i.e. Shortest Path Mixed PSO-SA.

Case 1:

When the number of particles is set to 100, the number of inputs each particle has is 20, and then average execution time of both the algorithms is as shown in Table 1:

Table 1. Shows the average execution time of PSO and Shortest Path mixed PSO-SA in Case 1

Execution Time in seconds		
Algorithm→	PSO Algorithm	Shortest Path Mixed PSO-SA
1	0.064	0.077
2	0.046	0.036
3	0.344	0.040
4	0.037	0.044
5	0.054	0.074
6	0.036	0.36
7	0.120	0.051
8	0.067	0.051
9	0.040	0.044
10	0.116	0.146
Average	**0.0924**	**0.0923**

Case 2:

Number of particles is set to 200,

Number of inputs each particle has is 40.

Then the average execution time of both the algorithms is shown in Table33:

Table 2. Shows the average execution time of PSO and Shortest Path mixed PSO-SA in Case 2

Execution Time in seconds		
Algorithm→	PSO Algorithm	Shortest Path Mixed PSO-SA
1	0.070	0.088
2	0.073	0.231
3	0.110	0.228
4	0.651	0.151
5	0.109	0.101
6	1.598	0.122
7	0.090	0.139
8	0.071	0.082
9	0.325	0.074
10	0.071	0.174
Average	**0.2168**	**0.139**

Case 3:

Number of particles is set to 300,

Number of inputs each particle has is 100, and

Then the average execution time of both the algorithms is shown in Table 3:

Table 3. Shows the average execution time of PSO and Shortest Path mixed PSO-SA in Case 3

Execution Time in seconds		
Algorithm→	PSO Algorithm	Shortest Path Mixed PSO-SA
1	2.368	0.872
2	4.841	0.467
3	4.188	0.687
4	2.880	0.282
5	0.206	0.687
6	3.428	1.405
7	3.447	0.303
8	2.753	1.213
9	0.314	0.308
10	2.574	1.497
Average	**2.6993**	**1.497**

Case 4:

Number of particles is set to 400,

Number of inputs each particle has is 150, and

Then the average execution time of both the algorithms is shown in Table 3:

Table 4. Shows the average execution time of PSO and Shortest Path mixed PSO-SA in Case 4

Execution Time in seconds		
Algorithm→	PSO Algorithm	Shortest Path Mixed PSO-SA
1	2.165	1.378
2	9.721	0.536
3	3.168	0.639
4	0.648	0.609
5	0.407	2.669
6	2.711	0.663
7	4.598	0.685
8	7.101	1.951
9	0.728	0.388
10	0.513	0.574
Average	**3.176**	**1.009**

Case 5:

Number of particles is set to 500,

Number of inputs each particle has is 200

Then the average execution time of both the algorithms is shown in Table 5:

Table 5. Shows the average execution time of PSO and Shortest Path mixed PSO-SA in Case 5

Execution Time in seconds		
Algorithm→	PSO Algorithm	Shortest Path Mixed PSO-SA
1	5.290	1.094
2	0.601	0.712
3	2.881	1.055
4	0.631	3.118
5	10.32	0.597
6	5.385	0.596
7	12.66	1.149
8	2.844	1.381
9	18.975	2.513
10	4.663	0.845
Average	6.4251	1.309

Case 6:

When the number of particles is set to 600, the number of inputs each particle has is 250, and then the average execution time of both the algorithms is shown in Table 6:

Table 6. Shows the average execution time of PSO and Shortest Path mixed PSO-SA in Case 6

Execution Time in seconds		
Algorithm→	PSO Algorithm	Shortest Path Mixed PSO-SA
1	1.244	1.467
2	35.528	5.951
3	0.861	1.369
4	74.749	2.293
5	74.640	1.972
6	62.483	1.588
7	69.327	0.771
8	8.013	1.815
9	0.934	0.853
10	8.900	0.805
Average	29.548	1.8884

All the above cases show that the new algorithm i.e. Shortest Path Mixed PSO-SA is much more efficient as it consumes very less time as compared to Particle Swarm Optimization algorithm. As the average execution time decrease the cost of execution also decreases The results are also compared graphically in the Fig. 2. In this graph it is worth noting that as the number of particle increases the difference between the average execution time (in second) of both the algorithms increases. Shortest Path Mixed PSO-SA algorithm consumes less time as compared to the Particle Swarm Optimization algorithm.
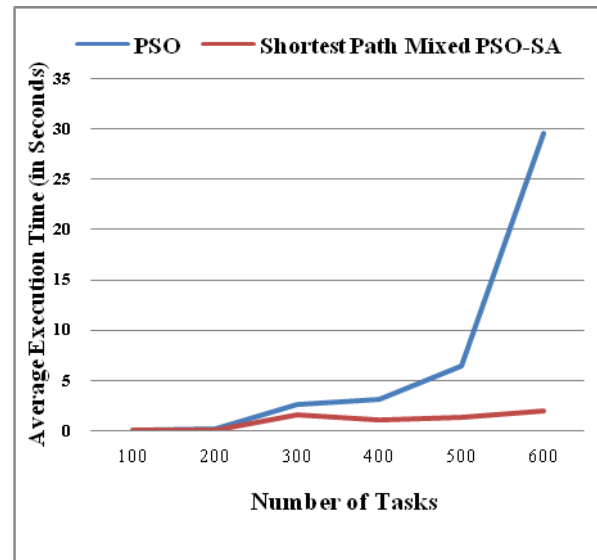

Fig 2: Flowchart of Proposed Algorithm

5. CONCLUSION

In this paper author achieved the need of user for management and execution of application on virtual machines in cloud computing in minimum possible time. The experimental evaluation is performed to prove that proposed algorithm consumes very less time as compared to the existing algorithm. This algorithm is very efficient as it follows the shortest path to achieve the target. It is also clear from the graph and tabular results by changing the number of parameters in Section 4.

6. REFERENCES

[1] Keith Jeffery, Burkhard Neidecker-Lutz, "The Future of cloud computing opportunities for European Cloud Computing Beyond*", European Commission Information Society and Media,* 2010.

[2] Shivkumar Buyya, "Applications scheduling and Management system for cloud infrastructure", 2010.

[3] Wei-Neng Chen; Jun Zhang, "A set-based discrete PSO for cloud workflow scheduling with user-defined QoS constraints," *Systems, Man, and Cybernetics (SMC), 2012 IEEE International Conference on* , vol., no., pp.773,778, 14-17 Oct. 2012

[4] Amin Jamili, Mohammad Ali Shafia, Reza Tauakkoli-Moghaddam, "A hybrid algorithm based on Particle Swarm Optimization and Simulated Annealing for a periodic job shop scheduling problem", *International Journal of Advancement in Manufacturing Technology*, Springer 2010.

[5] Suraj. Pandey, Linlin. Wu, Siddeswara. Guru, Rajkumar. Buyya, "A Particle Swarm Optimazation (PSO)-based Heuristic for Sceduling Workflow Application in Cloud Computing Environments", 2009.

[6] A. Alex, S. Harm, T.Q. Guo, L.G. Ning, "Infrastructure as a Service Cloud Concepts". Developing and Hosting Applications on the Cloud. *IBM Press*. ISBN 978-0-13-306684-5, 2012.

[7] Alexa Huth and James Cebula, "The Basics of Cloud Computing", *Carneige Mellon University, USCER, a government organization* © 2011.

[8] Amid Khatibi Bardsiri, Seyyed Mohsen Hashemi, "A review of workflow scheduling in cloud computing Environment", Vol 1, Issue 3, 2012.

[9] David Bufford, "Cloud Computing: A brief Introduction", LAD Enterprizes, 2010.

[10] Fei Teng, "Resource Allocation and Scheduling Models for Cloud Computing", tel-00659303, version 1- 12, 2012.

[11] Gunho Lee, "Resource Allocation and Scheduling in Heterogeneous Cloud Environments", Technical Report No. UCB/EECS-2012-78 http://www.eecs.berkeley.edu/ Pubs/ TechRpts/2012/EECS-2012-78.html, 2012.

[12] Hadi Salimi, Mahsa Najafzadeh, and Mohsen Sharifi, "Advantages, Challenges and Optimization of Virtual Machine Scheduling in Cloud Computing Environments", *International Journal of Computer Theory and Engineering* Vol.4,No.2, 2012.

[13] Hamdaqa M., Livogiannis T., Tahvildari L., "A Reference Model for Developing Cloud Applications", *International Conference on Cloud Computing and Services Science*, 98-103, 2011.

[14] http://www.boingboing.net/2009/09/02/cloud-computing-skep.html

[15] http://www.wikipedia.org.

[16] K. Dinesh, G. Poornima, K. Kiruthika, "Efficient Resource Allocation for Different Jobs in cloud", IGCA,Vol. 56, No. 10, 2012.

[17] K. Thanashkodi, K.Deeba, "A new improved Particle Swarm Optimization Algorithm for Multiprocessor Job Scheduling", *IJCSI*, vol. 8, No.1, 2011.

[18] Lizhe Wang, Gregon Von Laszewski, Marcel Kunze, Jie Tao, "Cloud Compuitng: a perspective study", 2008.

[19] Luiz F. Bitttencourt, Nelson L.S. da Fonseca, "Scheduling in Hybrid Clouds", 2012. http://www.gartner.com/id=914826 , http://assets1.csc.com/newsroom/downloads/CSC Cloud Usage Index Report.pdf , http://pegasus.isi.edu/.

[20] Mark F Tompkins, "Optimization Techniques for Task Allocation and Scheduling in Distributed Multi-Agent Operation", 2003.

[21] M. Kaladevi, S. Sathiyabama, "A comparative study of scheduling Algorithm for Real Time Task", *International Journal of Advances in Science and Technology*, vol. 1, No.4, 2010.

[22] M. Nadhini, S. Kanmani, "Comparison of Memetic Algorithm and PSO in Optimizing Multi Job Shop Scheduling", *International Journal of Advanced Research in Computing Engineering & Technology,* Vol. 1, Issue 5, 2012.

[23] M. Nandhini, S. Kanmani, "Enhanced PSO for Optimizing Multi Job Shop Scheduling", *International Journal of Multidisciplinary Educational Research* ISSN: Volume 1, Issue 3, 2277-7881, 2012.

[24] Mousumi Paul, Debabrata Samanta, Goutam Sanyal, "Dynamic Job Scheduling in Cloud Computing based on horizontal Load Balancing", *IJCTA,* Vol. 2(5), 2011.

[25] Mr. R. Gogulan, Ms. A. Kavitha, Mr. U.K. Kumar, "An Multiple Pheromone Algorithm for Cloud scheduling With Various QOS Requirements*", International Journal of Computer Science Issues*, Vol. 9, Issue 3, No 1, 1694-0814, 2012.

[26] Netjinda, N.; Sirinaovakul, B.; Achalakul, T., "Cost optimization in cloud provisioning using Particle Swarm Optimization," *Electrical Engineering/Electronics, Computer, Telecommunications and Information Technology (ECTI-CON), 2012 9th International Conference on* , vol., no., pp.1,4, 16-18 May 2012.

[27] Pooja.P.Vasani, Nishant S.Sanchani, "Literature review, various priority based task scheduling algorithms in cloud computing", *Journal of information, knowledge and research in computer engineering*, vol. 2, issue 2, 2012.

[28] Probir Roy, Md Mejbah Uh Alam, Nishit Das, "Heuristic based task scheduling in multiprocessor systems with Genetic Algorithm by choosing the Eligibal processor", *IJDPS,* Vol.3, No.4, 2012.

[29] Rajkumar Buyya, Chee Shin Yeo, Srikumar Venugopal, James Broberg, Ivona Brandic, " Cloud Computing and Emerging IT Platforms: Vision, Hype, and Reality for Delivering Computing as the 5th Utility", 2008.

[30] Ram Prasad Padhy, P. Goutam Prasad Rao, "Load Balancing in cloud computing systems",2011.

[31] Sandeep Tayal, "Task Scheduling optimization for the Cloud Computing Systems", *International journal of advanced engineering sciences and Technologie*, Vol No. 5, Issue No. 2, 111-115, 2012.

[32] Serafini P, Ukovich W, "A mathematical model for periodic scheduling problems". *SIAM J Discrete Math* 2(4):550–581.

[33] Shaobin Zhan, Hongying. Huo, "Improved PSO-based Task Scheduling Algorithm in Cloud Computing", *Journal of Information & Computational Science* 9: 13, 3821-3829, 2012.

[34] Sharat Chandra Racha, "Load Balancing Map-Reduce Communication for Efficient Execution of Applications in a cloud", 2012.

[35] Shuai Zhang, Shufen Zhang, Xuebin Chen, Xiuzhen Huo, "Cloud Computing Research and Development Trends", *Second International Conference on Future Networks*, 2010.

[36] Simsy Xavier, S. P. Jeno. Lovesum, "A survey of Various Workflow Scheduling Algorithms in Cloud Environment", *International Journal of Scientific and Research Publication*, Volume 3, Issue 2, ISSN 2250-3153, 2013.

[37] Sujit Tilak, Prof. Dipti. Patil, "A Survey of Various Scheduling Algorithms in Cloud Environment", *International Journal of Engineering Inventions* ISSN: Volume 1, Issue 2 2278-7461, PP: 36-39, 2012.

[38] Tsung-Lieh Lin, Shi Jinn Horng, Tzong-Wann Xao, Yuan-Hsin Chen, Ray-Shine Run, Rong-Jian Chen, Jui-Lin Lai, I-Hong Kuo, "An efficient job-shop scheduling based on particle swarm optimization",*Elsevier*,2010.