EAI Endorsed Transactions
on Scalable Information Systems Research Article **EAI.EU**

Enhanced Task Scheduling Using Optimized Particle Swarm Optimization Algorithm in Cloud Computing Environment

Sirisha Potluri[1, *], Abdulsattar Abdullah Hamad[2], Deepthi Godavarthi[3] and Santi Swarup Basa[4]

[1]Department of Computer Science and Engineering, Faculty of Science and Technology, (IcfaiTech), ICFAI Foundation for Higher Education, Hyderabad, India, sirisha.vegunta@gmail.com
[2]Samarra University, College of Education, Department of Physics, Iraq, a.a.hamad@uosamarra.edu.iq
[3]School of Computer Science & Engineering VIT-AP University, Amaravati, Andhra Pradesh, India, deepthi.g@vitap.ac.in
[4]Maharaja Sriram Chandra Bhanjadeo University, Baripada, Odisha, India, santiswarup.basa@gmail.com

Abstract

The most significant constraint in cloud computing infrastructure is the job/task scheduling which affords the vital role of efficiency of the entire cloud computing services and offerings. Job/ task scheduling in cloud infrastructure means that to assign best appropriate cloud resources for the given job/task by considering of different factors: execution time and cost, infrastructure scalability and reliability, platform availability and throughput, resource utilization and makespan. The proposed enhanced task scheduling algorithm using particle swarm optimization considers optimization of makespan and scheduling time. We propose the proposed model by using dynamic adjustment of parameters with discrete positioning (DAPDP) based algorithm to schedule and allocate cloud jobs/tasks that ensues optimized makespan and scheduling time. DAPDP can witness a substantial role in attaining reliability in by seeing the available, scheduled and allocated cloud resources. Our approach DAPDP compared with other existing particle swarm and optimization job/task scheduling algorithms to prove that DAPDP can save in makespan, scheduling and execution time.

Keywords: Cloud Computing, Load Balancing, High-Performance Computing, Task Scheduling, Job Scheduling, Particle Swarm Optimization

Received on 22 June 2023, accepted on 08 August 2023, published on 02 August 2023

doi: 10.4108/eetsis.4042

*Corresponding author. Email: sirisha.vegunta@gmail.com

1. Introduction

The Cloud computing paradigm allows on-demand delivery of computing resources- computing capabilities, database management, storage abilities, application support, etc. over a dynamic platform through the Internet connectivity by using the pay-as-you-go model [1-2]. With this technology, organizations are released from high investments in software/ hardware procurement, organization, and management [3-4].

In its place, the right kind and size of the computing resources are provisioned from the cloud service provider [5]. Cloud computing delivers a simple approach to access the various services- servers, networks, storage, processing, databases, and application services from the vendors over the Internet as shown in Figure 1.

2. Related work

The NP-hard problem of load balancing is a disadvantageous event on the cloud service provider's side that degrades the cloud performance and efficiency [6]. To maintain guaranteed Quality of Service on promised service level agreements between cloud consumers and cloud providers continuous audits takes place. Here, there arises the requirement of an efficient load balancing algorithm which is a robust topic of research among several researchers as given in Table 1.

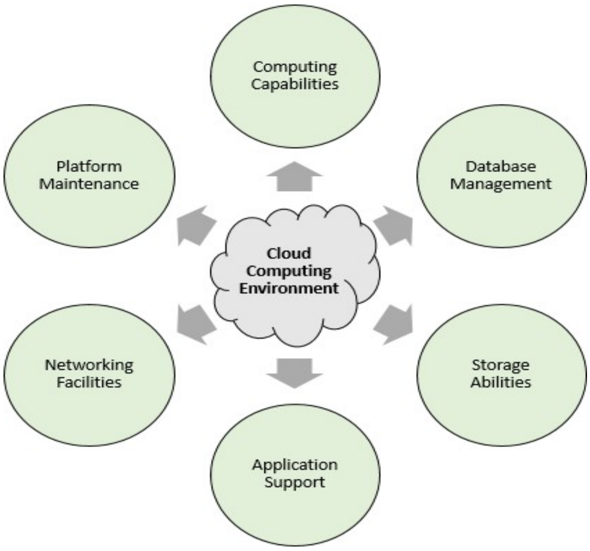

Figure 1. A primer on cloud computing environment

Table 1. State of the art load balancing algorithms in the cloud computing environment

Reference	Algorithm proposed	Advantages	Limitations
[7]	Heuristic task load balancing algorithm	Achieved enhanced elasticity with the capability of handling dynamics within deadline by reducing makespan	The task rejection ratio is high as tasks whose execution time is higher than the defined deadline are rejected
[8]	Integrated Harries Hawks Optimization and Pigeon inspired Optimization Algorithm are proposed	Optimal load balancing is achieved through optimal response time by using an optimal routing policy for each user response	Makespan and throughput of the tasks are not considered
[9]	Integrated Support Vector Machine and Ant colony Optimization Algorithm is proposed for scalable and robust load balancing	The proposed metaheuristic algorithm implemented an efficient multi-objective scheduling algorithm using which throughput time, and overhead time are optimized	Makespan, execution time, and energy consumption of task execution are not considered
[10]	The proposed design is to optimize resources and improve virtual machine load balancing given the quality of service metrics, the significance of virtual machines, and resource allocation	The proposed model achieved load balancing in less execution time and makespan	Virtual machine migration and dynamic load balancing can be improved for better performance
[11]	The gene expression programming for managing virtual machine hosts	The proposed model achieved power saving and maximum utilization of resources through efficient load consolidation and balancing	The trade-offs between efficiency and measurement need to be balanced
[12]	Maximum-minimum and round- robin load balancing algorithm	The proposed model improved response time and cost effectiveness	Maximum-minimum and round-robin load balancing algorithms can be enhanced by improving user satisfaction in the cloud service
	New static resource allocation strategy based on particle swarm optimization algorithm	The proposed model provided improved convergence and diversity strategy to reduce the degree of imbalance in load balancing	Can improve in random movement of particles to avoid improper discretization strategies
[14]	A hybrid meta heuristic with Genetic Algorithm and Particle Swarm	The proposed model achieved load	Can improve the efficiency to decrease energy intake

	Optimization approaches	balancing with greater performance	and carbon release
[15]	Bird Swarm Optimization Load Balancing algorithm	The proposed model obtained improved resource utilization and makespan of tasks	Can improve in terms of energy consumption of resources and reliability performance by using heterogeneous resources
[16]	Enhanced hybrid algorithm with recent antlion optimization algorithm and particle swarm optimization algorithm	Achieved enhanced workflow scheduling with improved Data Encryption Standard	Can improve in terms of success rate and trust management

3. Proposed model

The load balancing in cloud computing can be done at the physical machine level or VM level. The proposed framework recommends an efficient hybrid meta-heuristic load balancing algorithm to ensure a better quality of service through the optimal use of cloud resources. It is implemented in two main modules.

The first module is having two phases. In the first phase, it optimizes the make span and scheduling time of the tasks by using improved particle swarm optimization (PSO) algorithm. In the second phase, it reduces execution costs by optimizing the fitness function of PSO.

In the second module, it achieves improved load prediction for self-adaptive clouds by using a non-linear support vector machine algorithm for better load balancing in a cloud computing environment.

Through the improved fitness and kernel mathematical functions of the proposed model, the data center provides several virtual machines to allocate and execute tasks among them.

Our new load balancing model lets cloud services providers optimize and maximize their resource usage. Algorithm of the proposed dynamic adjustment of parameters with discrete positioning (DAPDP) based particle swarm optimization algorithm is given below.

Step1- Start
Step 2- Initialize DAPDP parameters
{Maximum number of iterations as Imax, size of the population as P, inertia weight (ω) range [max-min], learning factors l1 and l2}
Step 3- Initialize P, initial positions and particle velocities randomly
Step 4- Get fitness value of each particle by using following.

System Fairness (F) is defined, for n tasks {t1, t2, ……tn} and their corresponding fairness values {f1, f2, ……fn} as given in equation-1.

$$F = \sum_{i=1}^{n} | F_i | \text{ -(1)}$$

Step 5- For each particle pi, we compare the values of the fitness function in search of best position pbest
Step 6- For each particle pi, we compare the values of the fitness function in search of global best position pgbest
Step 7- For each particle pi, update the velocity v by using:
Particles pi = {p1, p2…...pn}, Particles pbesti = {pbest1, pbest2,….pbestn}, Particles pgbesti = {pgbest1, pgbest2,….pgbestn}. As given in equation-2.

$$vi = \omega * vi\text{-}1 + l1 * rand() * (pbest[] - vi\text{-}1) + l2 * random() * (pgbest[] - vi\text{-}1) \quad \text{-(2)}$$

Step 8- Check whether the current iteration reaches the best value, if it is not executing Step-3 for the next iteration and obtain optimal value.
Step 9- Obtain and output optimal results.
Step 10- Stop

4. Execution Environment and Parameter Setting

Cloud computing offers computing, storage, and software as a service. Through cloud computing provides different types of services over the internet, from software and analytics to secure and safe data storage and networking resources, everything can be delivered through the cloud framework.

To implement the proposed model, the CloudSim-cloud simulator is used to simulate the physical and logical structure of data centers in the cloud which contains thousands of physical machines and hundreds of virtual machines in an ordinary computer. Simulators may vary in functionalities such as availability of graphical interface, basic programming language, extensibility, etc.

CloudSim is a java based multi-layered cloud simulator. CloudSim can be used as a standalone simulator to study and analyze various existing policies and cloud entities also, it is possible to change the policies defined for various cloud operations depending on which class to extend. It provides essential classes for describing data centers, computational resources, virtual machines, applications, users, and policies for managing various parts of the system such as scheduling and provisioning. It can also be used to access the capabilities from various perspectives such as cost, application, execution, time, etc.

It also supports the evaluation of Green IT policies. CloudSim is flexible enough to be used as a library that allows you to add the desired scenario by writing a java program. Parameters of the proposed particle swarm optimization based dynamic adjustment of parameters with discrete positioning (DAPDP) is represented in Table 2.

Table 2. Algorithm Parameters

Name of the variable	Value/ Range
Population size	50
Virtual machines	10
Task size	30
Inertia	min=0.4, max=0.8
Learning factor l1	2
Learning factor l2	2
Number of iterations	200

5. Results and Discussions

Results of the proposed dynamic adjustment of parameters with discrete positioning (DAPDP) algorithm is compared with the existing state of the art algorithms using makespan and execution time as QoS parameters and are represented in Table 3. Makespan results are compared with Min-Min, Max-Min, QoS guided and PSO algorithms and plotted in Figure 2.

Table 3. comparison of makespan results of the proposed algorithm

Task Size/ Algorithm	Min-Min	Max-Min	QoS Guided	PSO	DAPDP
5	2.912	2.998	2.787	2.647	2.174
10	6.121	5.789	5.478	6.487	5.478
15	6.989	6.174	6.478	6.333	5.978
20	8.989	7.986	8.125	8.788	7.568
25	9.992	9.878	8.978	9.178	8.147
30	11.45	10.78	11.45	10.34	10.11

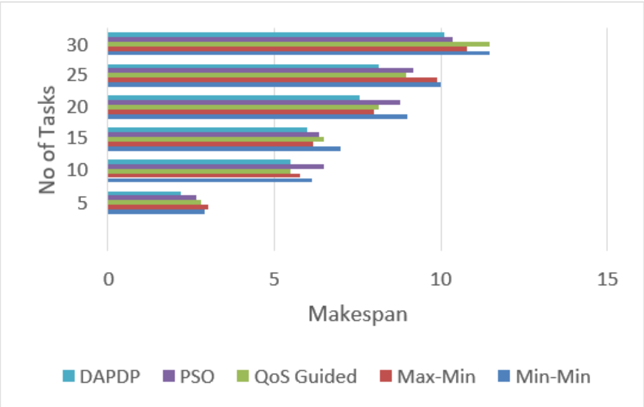

Figure 2. Makespan results of proposed algorithms vs existing algorithms

6. Conclusion and Future Work

In our research, to handle task scheduling problem in cloud computing environment, we proposed dynamic adjustment of parameters with discrete positioning (DAPDP) to satisfy QoS requirements. We improved particle swarm optimization by optimizing the inertia weight and allowing the position of the particle as coding discrete. And also we recommended a task scheduling algorithm to effectively handle task scheduling problem by improving the fairness function to optimize make span, scheduling and execution time. In future, the proposed model can be improved by reducing the execution cost by optimizing the fitness function of PSO with adaptive pbest updating through probability and similarity balanced discretization method. This approach will mark our method more expedient and effective.

References

[1]. Arunarani, A. R., Dhanabalachandran Manjula, and Vijayan Sugumaran. "Task scheduling techniques in cloud computing: A literature survey." Future Generation Computer Sys- tems 91 (2019): 407-415.

[2]. Kumar, Mohit, et al. "A comprehensive survey for scheduling techniques in cloud compu- ting." Journal of Network and Computer Applications 143 (2019): 1-33.

[3]. Rahimi, Morteza, et al. "Toward the efficient service selection approaches in cloud compu- ting." Kybernetes (2021).

[4]. Zhou, Zhou, et al. "A modified PSO algorithm for task scheduling optimization in cloud computing." Concurrency and Computation: Practice and Experience 30.24 (2018): e4970.

[5]. Junaid, Muhammad, et al. "Agile Support Vector Machine for Energy-efficient Resource Allocation in IoT-oriented Cloud using PSO." ACM Transactions on Internet Technology (TOIT) 22.1 (2021): 1-35.

[6]. Kumar M, Dubey K, Sharma SC (2018) Elastic and flexible deadline constraint load balanc- ing algorithm for cloud computing. Proced Comp Sci 125:717–724.

[7]. Annie Poornima Princess, G., Radhamani, A.S. A Hybrid Meta-Heuristic for Optimal Load Balancing in Cloud Computing. J Grid Computing 19, 21 (2021). https://doi.org/10.1007/s10723-021-09560-4.

[8]. M. Junaid, A. Sohail, A. Ahmed, A. Baz, I. A. Khan and H. Alhakami, "A Hybrid Model for Load Balancing in Cloud Using File Type Formatting," in IEEE Access, vol. 8, pp. 118135-118155, 2020, doi: 10.1109/ACCESS.2020.3003825.

[9]. Shafiq, Dalia Abdulkareem, et al. "A Load Balancing Algorithm for the Data Centres to Optimize Cloud Computing Applications." IEEE Access 9 (2021): 41731-41744.

[10]. D. A. Shafiq, N. Z. Jhanjhi, A. Abdullah and M. A. Alzain, "A Load Balancing Algorithm for the Data Centres to Optimize Cloud Computing Applications," in IEEE Access, vol. 9, pp. 41731-41744, 2021, doi: 10.1109/ACCESS.2021.3065308.

[11]. L. -H. Hung, C. -H. Wu, C. -H. Tsai and H. -C. Huang, "Migration-Based Load Balance of Virtual Machine Servers in Cloud Computing by Load Prediction Using Genetic-Based Methods," in IEEE Access, vol. 9, pp. 49760-49773, 2021, doi: 10.1109/ACCESS.2021.3065170.

[12]. Abiodun Kazeem Moses, Awotunde Joseph Bamidele, Ogundokun Roseline Oluwaseun, Sanjay Misra & Adeniyi Abidemi Emmanuel (2021) Applicability of MMRR load

balanc- ing algorithm in cloud computing, International Journal of Computer Mathematics: Com- puter Systems Theory, 6:1, 7-20, DOI: 10.1080/23799927.2020.1854864.

[13]. Miao, Zhang, et al. "A discrete PSO-based static load balancing algorithm for distributed simulations in a cloud environment." Future Generation Computer Systems 115 (2021): 497-516.

[14]. Yadav, Mala, and Sachin Gupta. "Hybrid meta-heuristic VM load balancing optimization approach." Journal of Information and Optimization Sciences 41.2 (2020): 577-586.

[15]. Mishra, Kaushik, and Santosh Kumar Majhi. "A binary Bird Swarm Optimization based load balancing algorithm for cloud computing environment." Open Computer Science 11.1 (2021): 146-160.

[16]. Kakkottakath Valappil Thekkepuryil, Jabir, David Peter Suseelan, and Preetha Mathew Keerikkattil. "An effective meta-heuristic based multi-objective hybrid optimization method for workflow scheduling in cloud computing environment." Cluster Computing 24.3 (2021): 2367-2384.