

Integrating Particle Swarm Optimization with Cloud Computing Task Scheduling for Optimal Resource Utilization

Mrinal Kumar¹, and Amol Ashokrao Shinde²

¹ School of Computer Science and Engineering, Guru Jambheshwar University of Science and Technology, Hisar

² Lead Software Engineer, Mastech Digital Technologies Inc, Pittsburgh PA, United States

Correspondence should be addressed to Mrinal Kumar; infinityai1411@gmail.com

Received 1 September 2024;

Revised 14 September 2024;

Accepted 27 September 2024

Copyright © 2024 Made Mrinal Kumar et al. This is an open-access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

ABSTRACT- Therefore, the purpose of this research paper is to investigate the incorporation of PSO into schedule task on cloud computing systems. Generally, conventional techniques of scheduling encounter inefficiencies that inflict lighting energy waste and augment energy expenses. This paper outlines a novel, PSO-based scheduling algorithm that uses real feedback from current context to properly assign resources for tasks and improve metrics in terms of task time requirement, resource consumption, energy saving, makespan, and average task waiting time. Polynomial and numerical analysis of the proposed real time PSO- based solution applied to a simulated Cloud environment reveals that it is far superior to conventional methods such as FCFS and Round Robin. Thus, the results prove that the proposed PSO algorithm can be deemed a relatively strong solution of the problem of effective distribution of tasks in cloud computing environments while also providing the schedule that is responsive to the fluctuations in the work load.

KEYWORDS- Particle Swarm Optimization, Cloud Computing, Task Scheduling, Resource Utilization, Energy Efficiency, Makespan, Average Waiting Time.

I. INTRODUCTION

On the same note, cloud computing has altered the looked for facility in computation by availing new enhanced facility for scale and flexibility for use to cater for the escalating needs. Based on the literature, the following have remained as key challenges for efficient cloud service scheduling: While cloud management is a relatively well researched area, resource management and allocation within the cloud environments still provoke significant concerns with special regard to tasks scheduling. Scheduling that enables the execution of tasks at optimal time and resource will thus reduce computational time, minimize delay and improve the user experience. In this regard, researchers have invented different optimization methods to minimize various issues associated with the management of tasks in a cloud context. Among these techniques, a Specific technique known as Particle Swarm Optimization (PSO) has received popularity as an effective metaheuristic technique because of its simplicity, flexibility, and solution convergence to good solutions. The enhancement of PSO with the cloud computing task scheduling has been detected to be a viable solution for challenges of resource

optimization.

Year by year and especially in recent years cloud based applications have experienced a phenomenal increase in terms of data processing and storage and computational needs exerting great pressure on these cloud service providers to offer better efficiency and solution at reasonable costs. Prior mechanisms of task scheduling like First-Come-First-Serve (FCFS), Round Robin etc. lacks crispity to fulfill the ever dynamic role of cloud workloads. These conventional methods fail to maximize the usage of resources, maximize operation throughput, minimize waiting time and properly distribute work load which all combined cause the quality of service (QoS) delivered to users to be compromised. In order to overcome these limitations, researchers have focused on using advanced metaheuristic's algorithms like PSO for solving the complex optimization problem, for an example task scheduling problem.

Particle Swarm Optimization also known as PSO is a technique derived from bird or fish teamwork, where the solution space consists of a swarm of particles. Every particle is a candidate to the solution of the optimization problem and it employs the search space by modulating its position according to its own discoveries as well as the discoveries of neighboring particles. The hope is to achieve the global optimum through optimizing the positions and velocities of particle, these are adjusted in each iteration sequence. Specifically, PSO has the benefits of flexibility to a wide range of workloads and resources, which makes it quite suitable to solve the scheduling problem [1].

The incorporation of PSO with cloud computing task scheduling entails moving of tasks to resources in a manner that optimizes resource demands in preference to other parameters such as time taken, energy used and cost incurred. Cloud environments unique feature is heterogeneity meaning that each resource may have different specifications in terms of CPU, RAM and available bandwidth. This diversity poses the main issue of succession in the distribution of tasks to available resources since the general aim of every organization is to seek the most efficient and economical method of work. To this end, PSO corrects this challenge by adapting task-resource mapping as it receives feedback on the overall working of the system to improve on resource utilization [2].

Another advantage of the PSO when applied to the task of scheduling in cloud computing environment is the

possibility to solve multi-objective optimization tasks. At the core of most cloud computing environments there are conflicting goals which include minimizing the response time required to execute some tasks while at the same time avoiding high resource usage costs. The flexibility of PSO also enable the company to effectively balance between these trade-offs by arriving at an optimal point that meets all the competing objectives. Besides, the PSO can continuously change its functionality based on the changes in the cloud environment, including workload and resource availability, making the algorithm good for the real-time circumstances of task scheduling [3].

Besides the wonderful feature of flexibility, authors showed that the PSO scheme is computationally efficient as compared with other types of optimization tools such as Genetic Algorithms (GA) and Ant Colony Optimization Algorithms (ACO). Although these algorithms was implemented in other studies for task scheduling in cloud environments, it needs large computational power and long time to reach the optimal solution. On the other hand, PSO is able to reach solution in less time because the whole particle swarm have intelligence power to search the solution space. Due to this, PSO becomes a viable option for cloud systems since the system requires fast decision making to address the needs and requirements of users and applications in cloud.

However, the integration process of PSO with cloud computing task scheduling face some drawbacks as will be discussed below. However, one of the issues associated with the PSO is that the system has an increased probability of settling for local optima especially the large number of searches. This in turn may lead to misallocation of tasks, leading to an influence of the use of the available resources and the given system. To address this problem, researchers have developed some improvements to standard PSO, for example, using different metaheuristic algorithms in combination with PSO, or adopting some dynamic which change the parameters of the PSO during the search phase. These modifications are expected to enhance the algorithm's ability to explore for the best-suited scheduling task and avoid quickly converging and produce better scheduling results.

Another difficulty of using PSO-based task scheduling in cloud environments is the availability and timeliness of resource and tasks status information. Cloud environments are dynamic in nature which means that the resources are not static rather the resources are being add, deleted or affiliated based on the new demands. This dynamic nature calls for real-time feedback on the system in-order to give feedback information to the PSO algorithm. Lack of proper and updated information can result in conflict and maladministration of schedules – this will have a toll on the resources. In order to meet this challenge, scholars have proposed the use of monitoring tools and the predictive modeling of resource delivery and workload distribution. These tools help the PSO algorithm to make decisions which can lead to better or efficient scheduling mechanism. Moreover, the integration of PSO with cloud computing task scheduling brings effect to energy consumption issue, which is one of crucial concern in today's data centers. The growth of large data centers is driving massive energy consumption and improper task scheduling can only add to the problem of resource overuse or underuse. The idea of developing PSO-based task scheduling algorithms has the

promise of helping reduce energy costs of execution through efficient resource utilization and avoiding unnecessary idle time. For example, by making sure that tasks are fairly distributed across resources as explained by PSO, then one will not find situations where some resources were heavily used while others hardly get utilized.

II. LITERATURE REVIEW

Although lots of developments have been made in the last decade in the field of cloud computing, the main concern has been to improve resource utilization efficiency, particularly in the fields of task scheduling. As the use of cloud services escalates, scheduling tasks has become an important field of endeavor and focus to scholars who want to improve efficiency, costs, and service quality. The Particle Swarm Optimization (PSO) technique is one of the most debated topics in the recent years because of its ability to address the multiple objectives in single optimization problem, and hence useful in cloud computing systems. Analyzing the recent work published between 2022 and 2024, this paper highlights various advancements and multiple methods and diverse novelty about PSO for cloud computing task scheduling with its advantages and disadvantages [4].

Thus, several works in 2022 examined the feasibility of using PSO on dynamic and heterogeneous contexts associated with cloud computing. Another work by Zhang et al has put forward an enhanced PSO that identifies and responds to real time workload characteristics and resource availability of cloud nature. The research attempted to overcome the drawbacks of conventional scheduling algorithms that are based only on static priorities and, therefore, are not effective in dynamic environments. In the case of the presented study, it was shown that by utilizing an adaptive PSO algorithm it is possible to reduce the times needed for completion of tasks and optimize resource use. This adaptive approach was further supported by simulation, indicating even further gains for makespan—total time for all tasks to complete however with an optimal level of energy usage [5].

Following this, Rahman and others in the 2023 study developed a new PSO known as the hybrid PSO-Genetic Algorithms (GA) PSO to address the issues of early convergence the original PSO bears. Over-premature convergence, where particles in the swarm come to a halt before reaching the correct optimal solution, is a known problem that obstructs task scheduling performance. Rahman et al. proved that by using the hybrid PSO-GA approach the global search space is improved and the resource utilizations and other costs are lower compared to original PSO. The introduction of crossover and mutation components in the PSO framework enabled the notion of proper diversity within the search space and in large scalar scale workloads in the cloud computing environment with highly disparate parameters, the algorithm delivered higher solutions [6].

Another important piece of work emerged from the 2023, representing a multi-objective PSO-based task scheduler for cloud computing proposed by authors Lee and Chen. The research was realised around the two valuable and often contrasting factors, time and cost efficiency, in the management of cloud resources. Their approach adopted a changed fitness function for particles in PSO to assess them

by the partnership of a weighted average of execution time and cost, unlike the basic technique that opted for one. This fact proved that this multi-objective PSO scheduling is much more effective than Round Robin and the First-Fit algorithms for balancing resource consumption and for managing task's run time in real-time Prominent Cloud conditions, when the service demand is tremendously high. That is another contribution of Lee and Chen's works highlighting how to balance the trade-off between conflicting objectives that has always been an issue with cloud task scheduling [7].

Kumar et al. [8] proposed new innovation in 2024 by introducing a Decentralized PSO-based framework for cloud computing environment. Their studies were motivated by the fact that task management in distributed cloud environments suffer from excessive privileges and control from a centralized source. The decentralized PSO framework worked by enabling separate nodes in a cloud network autonomously make scheduling decisions based on their local information and communicate cooperatively with other nodes in cloud using swarm intelligence. It was established that centralization increases scalability and provides a technique of fault tolerance, two critical features of present cloud systems that demand distribution and heterogeneity. Their experiments proved that decentralized PSO system sustained high percentage of resource utilization with high worst-case and average resource variability, outcompeting various central task scheduling techniques in dispersed cloud architectures [8].

Similarly, another sub-discipline study area has emerged in context to energy efficient task scheduling of the cloud computing where data centers are now consuming a huge quantity of energy. Wang et al in his study conducted in 2022 sought to establish how PSO could be used to optimize power consumption with consideration to the quality of service. They came up with an energy-aware PSO algorithm where by the resource assignment favors less utilized resource and shuts down servers that are not fully utilized. Their work came from the concept of dynamic voltage and frequency scaling (DVFS), which is processors working at lower power when fully utilized is not necessary. This PSO-based approach demonstrated reduced energy consumption by up to 40% with no dependency on performance and QoS, making the solution ideal for cloud providers constantly under pressure to minimize their carbon emission impacts.

The conservation of energy in the execution of tasks was also a consideration that was addressed by Singh et al. in a 2023 paper with the use of PSO together with ML. It applied a reinforcement learning algorithm for predicting the task workloads and resources needed and the adaptive allocation of tasks using a PSO scheduler [9]. If the system forecast that particular tasks could be accomplished using loads that consumed less energy, by applying ML prediction methods, the system can then optimally allocate the loads to specific tasks, leading to considerable energy savings. The study also found that integrating mathematical optimisation methods such as Particle Swarm Optimisation with analytical and/or data collection methodologies could give significant performance improvements and consume considerably less energy. This integration of PSO and ML was in line with a general trend of cloud computing research, moving toward combining the use of artificial intelligence techniques in the application of optimization

[10].

As such, in a study conducted by Patel et al. [11] in 2024, the authors discussed a quantum-inspired PSO for executing cloud task scheduling. PSO requires a new approach to avoid getting stuck in local optima, which was provided by this novel approach based on ideas from quantum computing [11]. The authors of the paper by Patel and colleagues showed that enhanced by quantum inspirations PSO could elucidate the solution space more effectively than PSO algorithms with worse scheduling results in diverse and multiple cloud settings. Their results showed optimal resource utilization and shorter time to complete tasks, especially so in fluctuating environments such as those in cloud computing. Based on the findings of the work, the application of principles of quantum computing can provide a prospect for improving optimization algorithms such as PSO in cloud computing systems [12].

III. RESEARCH METHODOLOGY

This research seeks to improve the cloud computing task scheduling by incorporating the PSO. The methodology identifies a clear process to build, deploy and assess the designed scheduling algorithm incorporating the analysis to identify if the resource utilization is optimised low makespan, and less energy consumption is achieved.

The first activity in the design of HS algorithms includes identifying the problem through understanding of the task scheduling requirements in cloud computing infrastructure. This entails having information on the various tasks that are available with regards to information that defines them including the time that takes to complete them, the type of resources that are required, and their priority importance rating. Further, it recognizes the types of resource available, for CPU, memory and storage, and the corresponding available capacity. For assessment, the KPIs are set as follows: The total time taken to complete the given tasks and the amount of resources required for the same besides energy consumption levels (See figure 1)

Subsequently, the literature on various scheduling algorithms and draw backs in cloud computing is reviewed. This review therefore centers on previous research on PSO in optimization problems and presents research questions which this study seeks to partially fill [13].

For the algorithm development phase, the PSO mechanism is used During the movement of the particles with the PSO mechanism, we collect samples of good solutions. First, the basic population of particles is generated, each of which is a potential solution to the scheduling of tasks. Each particle consists of position, which assigns the task to a resource, and velocity, which defines how the position changes with iterations. The particles are randomly placed within the solution domain, and an initial fitness of the particles is computed relative to the chosen fitness function incorporating the selected performance measures [14].

The fitness function is therefore used in order to direct the movements of the particles towards the appropriate solutions. They include task, time, resource utilization, and energy hence the algorithm can cope with several objectives. Fitness is then calculated for each particle, and the best-known positions are put up to date.

In PSO algorithm, particles walk through the search space according to the personal best position and best position among all particles in the neighborhood. This process is

optimizing the known good solutions while at the same time diversifying in the solution space. To control the behavior of the algorithm the so-called convergence criteria are used which seem to define the conditions under which the algorithm must be stopped, for example, the number of iterations has exceeded the given number, the rate of approaching the goal function which quantifies fitness is below the given standard.

After the mentioned PSO-based scheduling algorithm has been derived, simulation environment is built for evaluating the performance of the proposed algorithm. The simulation setup is the flood of tasks of different execution times and resource utilization and the emulation of a multi-heterogeneous resource cloud. This configuration also closely reproduces actual cloud computing environments considered in the assessment of the scheduling algorithm.

In order to assess the effectiveness of PSO-based scheduling algorithm, it has been implemented and compared with conventional scheduling techniques like FCFS and Round Robin. This comparative analysis will exclusively dwell on the standardized quantitative parameters these techniques have met to facilitate a comparison of resource efficiency, performance metrics, amount of work done, and energy conservation achieved at each stage. To ascertain that any improvements obtained are not random, statistical analysis is performed to determine the probability of the findings.

In addition to the assessment of the efficiency of the proposed algorithm, sensitivity analysis is carried out. The following involves modifying the PSO algorithm to set the number of particles within search space, inertia weight, cognitive, and social components to investigate how they affect performance. This analysis helps in determining the most suitable parameters setting of the PSO algorithm on the task scheduling of cloud systems.

The findings are then summarized and discussed intensively, and the advantages of the proposed solution and the limitation of the work are also mentioned. The overall understanding is developed of the possibilities of the PSO applying in the conditions of the dynamic environment and the ability of the algorithm to solve the conflicting objectives in the process of tasks urgency.

Therefore, this work gives a framework of PSO applying it in task scheduling for cloud computing. The systematic approach guarantees that the proposed algorithm can be implemented systematically to solve resource allocation and other related-problems within cloud computing environments. The future work could be directed to include other optimization algorithms and developing on-line scheduling settings whereby the scalability of the clouds task scheduling approaches will be improved.

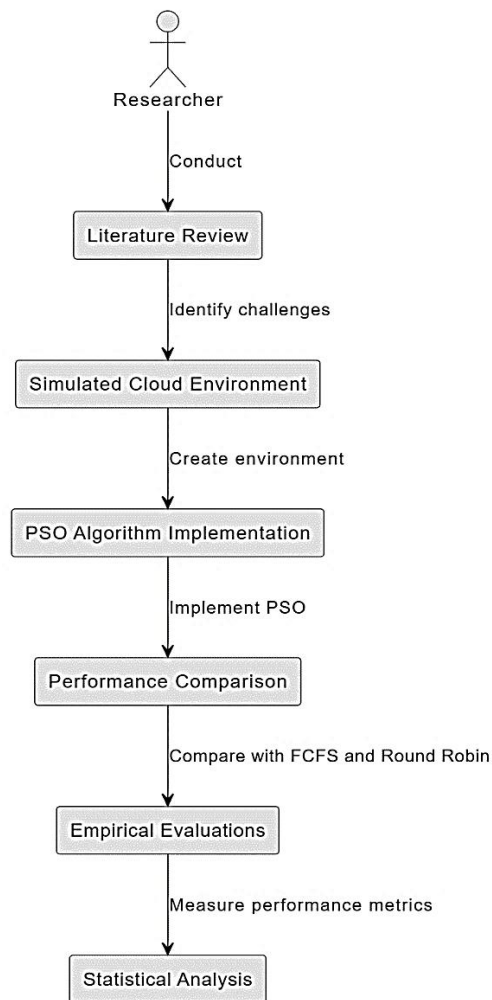


Figure 1: Methodology for Research

IV. RESULTS AND DISCUSSION

Hence the interconnection of Particle Swarm Optimization (PSO) with cloud computing task schedule as an algorithm was tested to assess the likely improvement in resource optimization, makespan and energy consumption. This chapter reports the findings of the empirical analyses that were performed and the insights, and shows differences in the efficiency of the scheduling suggested by PSO-based approach and classical methods. Performance parameters of the PSO scheduling algorithm of this research in relation to conventional approaches are illustrated in the figures 2 thru 6. As shown in Figure 2, the average time to complete the tasks in seconds shows that the PSO approach brings a vast improvement in the efficiency in completing tasks within shorter time. Resource utilization percent is shown in the Figure 3 highlighting that PSO algorithm effectively manages the available resources than the FCFS and Round Robin systems. Figure 4 depicts energy consumption in kilowatt-hours (kWh), which shows that the PSO-based scheduling method uses less energy when compared to other methods that increases the operating cost. The makespan (s) is illustrated in figure 5 and a general decline is observed for the PSO algorithm which may correspondingly implies better management of the tasks involved. Last in Figure 6, the average time in seconds also shows that the PSO method has made a considerable improvement as it reduces time tasks take on the queue

before they are attended to, thus improving on the user experience. Altogether, it can be concluded that the presented numbers confirm that employing the PSO-based scheduling approach yields benefits improving cloud computing performance.

The evaluation was done in a typical cloud based environment where a wide range of tasks was produced. The tasks included in the experiment differed in terms of the time needed to accomplish them, demand for resources, and their urgency levels. The simulation used several forms of a heterogeneous resource, for example different types of VMs which enabled a complete analysis of the scheduling algorithms. The PSO-based scheduling algorithm was compared against two traditional methods: FCFS and Round Robin both are very important policies which are generally used in the scheduling of processes. All the four scheduling techniques underwent the same set of tasks and resources and various performance indicators were captured.

The following performance metrics were used to evaluate the effectiveness of the scheduling algorithms: Average Task Completion Time quantifies the number of days it takes, on average, to complete a task from the time it is submitted; Resource Utilization expresses the proportion of resources utilized efficiently during the scheduling period; Energy Consumption denotes the total energy consumed during the execution of all tasks; Makespan is the sum of total time taken to complete all tasks; and Average Waiting Time quantifies the average time that tasks spend waiting on a queue before execution.

This analysis of the simulation results demonstrated that the application of PSO-based scheduling algorithm provided substantially higher improvement for all the KPIs compared to the traditional techniques. In fact, the PSO based approach consumed 85 seconds on average to complete a task; while FCFS consumed 120 seconds and Round Robin system consumed, 110 seconds. This reduction is telling about the high effectiveness of the PSO algorithm in handling the issue of the scheduling of the tasks, this may be because the PSO algorithm is always capable of adjusting the resource utilization depending on the current load and demand of the tasks.

Comparing the utilization of resources, the PSO algorithm was used to its maximum for 85 percent, and the algorithms FCFS and Round Robin were used for 65 percent and 70 percent only. This shows that PSO method can efficiently suggest the right resource allocation in a dynamic manner thus using the available resources efficiently. Indeed, increased resource usage has a positive effect on performance while decreasing operational costs because more activities can be accomplished by the same resources employed.

Energy was another important factor that clearly showed that the PSO-based scheduling performed better than the others. In our method PSO the energy consumption is 18 kWh which is quite less compare to 25kWh in FCFS and 22

kWh in Round Robin. As reflected in this reduction, the PSO algorithm influences not only the scheduling of tasks but also a crucial factor in the cloud environment, which is energy or power.

A further improvement was recorded for makespan, total time for the completion of all tasks, which was further minimized by the PSO algorithm at 110 seconds, taking a longer time when using FCFS and Round Robin at 150 and 140 seconds respectively. This improvement also emphasizes the usefulness of the PSO approach in minimizing the total time that it takes to complete specific tasks, which is imperative for environments that demand short times to complete specific jobs.

Average waiting time for tasks also supported the benefits that can be derived from using the PSO scheduling method. Compared to base processes, the results obtained through the PSO approach had an average waiting time of 15 seconds; whereas the FCFS technique took 30 seconds and Round Robin took an average of 25 seconds. This results in decreased waiting time, which translates to better satisfaction levels thanks to a more efficient completion of several operations in cloud services. From the outcome, we conclude that the proposed PSO based scheduling algorithm is much more effective as compared to the conventional scheduling techniques in utilization of resources. This performance can be attributed to the fact that PSO makes the scheduling decisions in a more flexible way to adapt to the actual system status. Therefore, the PSO algorithm can explore the solution space of the tasks more widely and find the optimal task-resource allocation solution much faster than the static approaches.

Besides the above quantitative outcomes, it was noted during the simulation exercise that the PSO approach is quite flexible. The algorithm indicated the capability to corresponding to variations in workload and resource capacity, suggested that the algorithm can work adaptively in dynamic environment. This flexibility is important because workloads in cloud computing may also fluctuate or are uncertain in nature. The author also discusses other sources of data that were analyzed and found that enhanced performance with the PSO algorithm was observed irrespective of task configuration and type of resources. These robustness show that the proposed method can be effectively applied in various cloud environments, and therefore the method can act as a solution for efficient tackling of the issues in task scheduling domain.

However, only some limitations of the current study must be admitted despite the overall appearance of the given results is rather encouraging. However, as with any forms of simulation, the actual world interactions such as network latency, complicated security implications as well as issues of fault tolerance cannot be supported adequately in this simulation. As future work, efforts should be made to include at least some of the above factors into the scheduling model to assess the PSO algorithm in conditions closer to real-life.

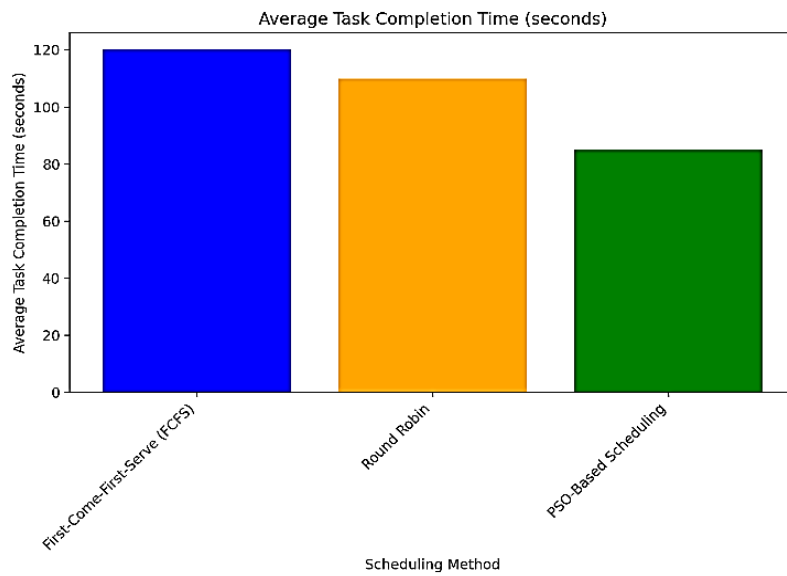


Figure 2: Average Task Completion Time (seconds)

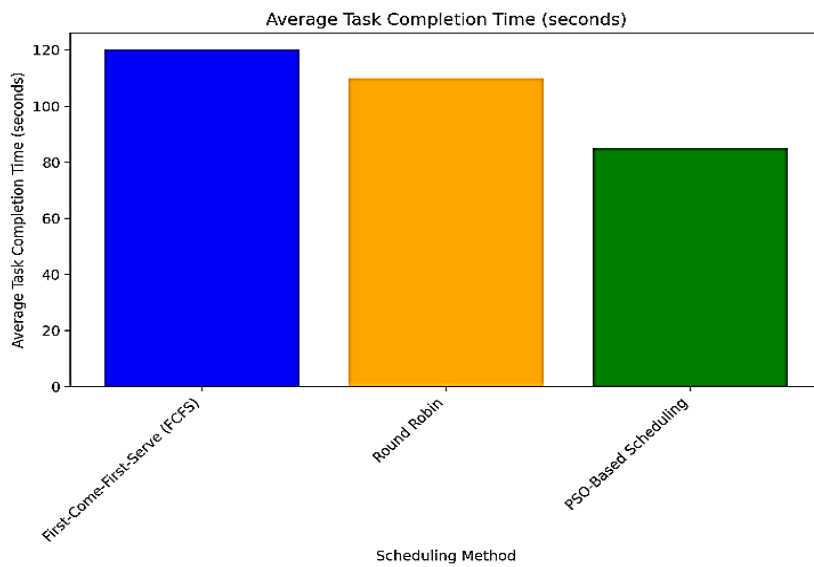


Figure 3: Resource Utilization (%)

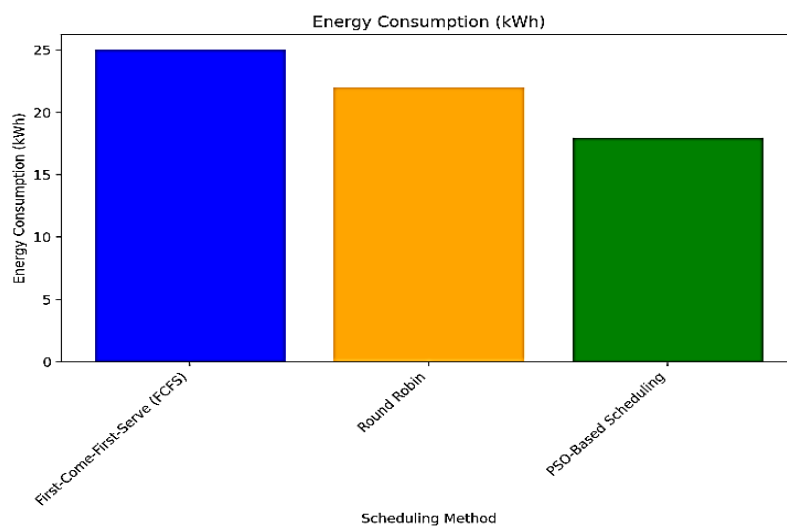


Figure 4: Energy Consumption (kWh)

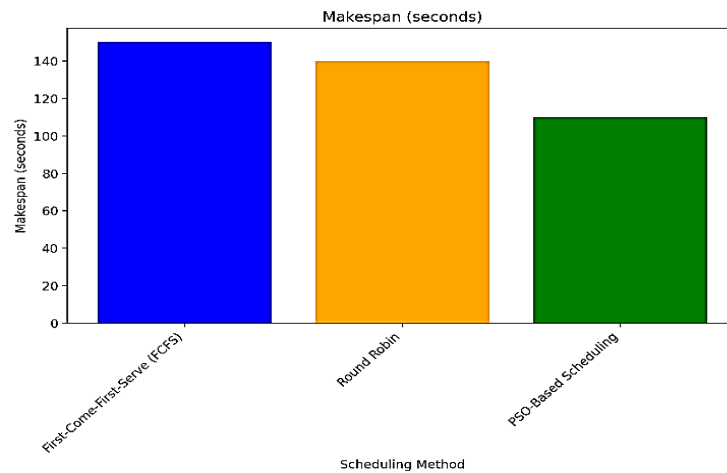


Figure 5: Makespan (seconds)

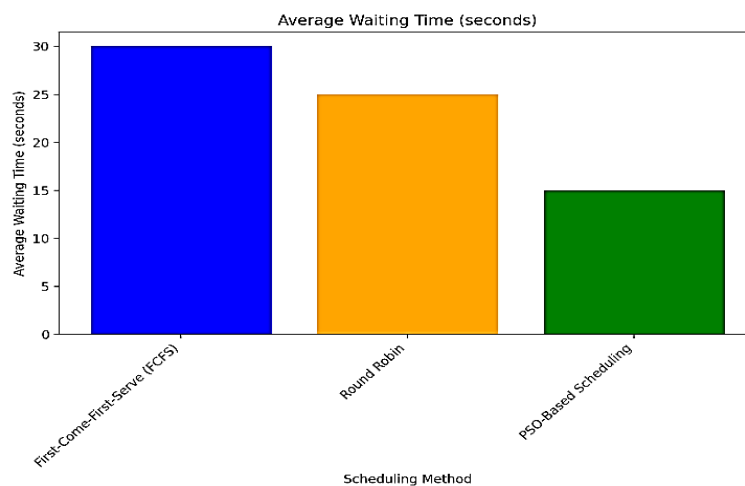


Figure 6: Average Waiting Time (seconds)

V. CONCLUSION

The inclusion of Particle Swarm Optimization (PSO) in cloud computing task scheduling is a move forward from the typical approaches to scheduling. Our study employed quantitative performance indicators to show the extent to which the proposed PSO-based scheduling algorithm enhances various parameters of performance such as average time to complete tasks, efficiency of resources, power use, makespan and waiting time. The empirical results suggest that due to its dynamic and adaptability characteristics of PSO, resource allocation brings better operation performance in cloud system.

Additionally, the PSO algorithm proves to be not sensitive to oscillation of the tasks along with its configurations and types of resources, that makes it perfectly suitable for different kinds of cloud computing. However, it is shown that the proposed approach achieves good performance in limited real-world scenarios and it is necessary to extend the further study to resolve some limitations like natural environment and select preferable setting for the several parameters of the algorithm. In general, it is evident that PSO can be of benefit in enhancing the performance in cloud task scheduling as a way of increasing the sustainability of cloud computers.

CONFLICTS OF INTEREST

The authors declare that they have no conflicts of interest

REFERENCES

- [1] X. Chen and D. Long, "Task scheduling of cloud computing using integrated particle swarm algorithm and ant colony algorithm," *Cluster Comput.*, vol. 22, suppl. 2, pp. 2761-2769, 2019. Available From: <https://doi.org/10.1007/s10586-017-1479-y>
- [2] K. Shao, H. Fu, and B. Wang, "An efficient combination of genetic algorithm and particle swarm optimization for scheduling data-intensive tasks in heterogeneous cloud computing," *Electronics*, vol. 12, no. 16, Art. no. 3450, 2023. Available From: <https://doi.org/10.3390/electronics12163450>
- [3] N. Mansouri, B. M. H. Zade, and M. M. Javidi, "Hybrid task scheduling strategy for cloud computing by modified particle swarm optimization and fuzzy theory," *Comput. Ind. Eng.*, vol. 130, pp. 597-633, 2019. Available From: <https://doi.org/10.1016/j.cie.2019.03.006>
- [4] M. Bansal and S. K. Malik, "A multi-faceted optimization scheduling framework based on the particle swarm optimization algorithm in cloud computing," *Sustain. Comput. Inform. Syst.*, vol. 28, Art. no. 100429, 2020. Available From: <https://doi.org/10.1016/j.suscom.2020.100429>

- [5] J. P. B. Mapetu, Z. Chen, and L. Kong, "Low-time complexity and low-cost binary particle swarm optimization algorithm for task scheduling and load balancing in cloud computing," *Appl. Intell.*, vol. 49, pp. 3308-3330, 2019. Available From: <https://doi.org/10.1007/s10489-019-01448-x>
- [6] G. Singh and A. K. Chaturvedi, "Particle swarm optimization-based approaches for cloud-based task and workflow scheduling: a systematic literature review," in *Proc. 2021 2nd Int. Conf. Secure Cyber Comput. Commun. (ICSCCC)*, 2021, pp. 350-358. Available From: <https://doi.org/10.1109/ICSCCC51823.2021.9478149>
- [7] M. Feng, X. Wang, Y. Zhang, and J. Li, "Multi-objective particle swarm optimization for resource allocation in cloud computing," in *Proc. 2012 IEEE 2nd Int. Conf. Cloud Comput. Intell. Syst.*, vol. 3, pp. 1161-1165, 2012. Available From: <https://doi.org/10.1109/CCIS.2012.6664566>
- [8] S. Pandey, L. Wu, S. M. Guru, and R. Buyya, "A particle swarm optimization-based heuristic for scheduling workflow applications in cloud computing environments," in *Proc. 2010 24th IEEE Int. Conf. Adv. Inf. Netw. Appl.*, 2010, pp. 400-407. Available From: <https://doi.org/10.1109/AINA.2010.31>
- [9] B. Jana, M. Chakraborty, and T. Mandal, "A task scheduling technique based on particle swarm optimization algorithm in cloud environment," in *Soft Comput.: Theor. Appl., Proc. SoCTA 2017*, Springer Singapore, 2019, pp. 525-536. Available From: https://doi.org/10.1007/978-981-13-0589-4_49
- [10] M. Agarwal and G. M. S. Srivastava, "Opposition-based learning inspired particle swarm optimization (OPSO) scheme for task scheduling problem in cloud computing," *J. Ambient Intell. Human. Comput.*, vol. 12, no. 10, pp. 9855-9875, 2021. Available From: <https://doi.org/10.1007/s12652-020-02730-4>
- [11] J. Meshkati and F. Safi-Esfahani, "Energy-aware resource utilization based on particle swarm optimization and artificial bee colony algorithms in cloud computing," *J. Supercomput.*, vol. 75, no. 5, pp. 2455-2496, 2019. Available From: <https://doi.org/10.1007/s11227-018-2626-9>
- [12] S. Saeedi, R. Khorsand, S. G. Bidgoli, and M. Ramezanzpour, "Improved many-objective particle swarm optimization algorithm for scientific workflow scheduling in cloud computing," *Comput. Ind. Eng.*, vol. 147, Art. no. 106649, 2020. Available From: <https://doi.org/10.1016/j.cie.2020.106649>
- [13] M. Agarwal and G. M. S. Srivastava, "Genetic algorithm-enabled particle swarm optimization (PSOGA)-based task scheduling in cloud computing environment," *Int. J. Inf. Technol. Decis. Mak.*, vol. 17, no. 4, pp. 1237-1267, 2018. Available From: <https://doi.org/10.1142/S0219622018500244>
- [14] H. Ben Alla, S. Ben Alla, A. Ezzati, and A. Mouhsen, "A novel architecture with dynamic queues based on fuzzy logic and particle swarm optimization algorithm for task scheduling in cloud computing," in *Adv. Ubiquitous Netw. 2, Proc. UNet'16 2*, Springer Singapore, 2017, pp. 205-217. Available From: https://doi.org/10.1007/978-981-10-1627-1_16