Journal of Information & Optimization Sciences
ISSN 0252-2667 (Print), ISSN 2169-0103 (Online)
Vol. 45 (2024), No. 2, pp. 351–360
DOI : 10.47974/JIOS-1559

Enhancing cloud service efficiency through ant colony optimization with multi-objective task scheduling

Prabh Deep Singh *
Department of Computer Science & Engineering
Graphic Era (Deemed to be University)
Dehradun
Uttrakhand
India

Kiran Deep Singh †
Department of Computer Science & Engineering
Chitkara University Institute of Engineering and Technology
Chitkara University
Punjab
India

Harsh Taneja §
Rohan Verma ‡
Department of Computer Science & Engineering
Graphic Era (Deemed to be University)
Dehradun
Uttrakhand
India

Abstract

The emergence of cloud computing technology as a prevailing paradigm for providing scalable and readily available computing resources has presented a complex challenge in terms of efficiently allocating activities to resources, taking into consideration many innovative applications. The integration of Ant Colony Optimization, with particular focus on addressing multiple objectives, presents a potentially adaptable approach to tackling the intricacies inherent in the management of cloud services. This problem is often categorized as an NP-hard problem and necessitates methods that not only cater to user

* *E-mail:* **ssingh.prabhdeep@gmail.com (Corresponding Author)**
† *E-mail:* **kdkirandeep@gmail.com**
§ *E-mail:* **harshtaneja88@gmail.com**
‡ *E-mail:* **rohanvermahnb@gmail.com**

© Taru Publications

requirements but also enhance overall system efficiency. In this paper, the application of Ant Colony Optimization (ACO) is utilized for improving cloud service efficiency. As cloud infrastructure handle diverse and dynamic workloads, the study addresses the challenges of optimizing resource utilization and minimizing cost. The proposed methodology introduces multi-objective optimization, aiming to balance conflicting goals and create Pareto-optimal solutions. The integration of ACO in cloud-based job scheduling contributes to the discourse on advancing efficiency in modern computing environments. Results show the effectiveness of the proposed approach.

Subject Classification: *62-XX, 65-XX.*

Keywords: *Ant colony optimization, Multi-objective task scheduling, Cloud/fog computing, Cloud service efficiency.*

1. Introduction

In the realm of cloud computing, where the demand for efficient resource allocation and task scheduling is paramount, the integration of sophisticated optimization techniques becomes imperative (1) (2). This research endeavors to explore the potential of Ant Colony Optimization (ACO) in enhancing cloud service efficiency through a novel approach—Multi-Objective Task Scheduling. As cloud infrastructures cater to diverse and dynamic workloads, the conventional challenges of optimizing make span, resource utilization, and energy consumption demand innovative solutions (3) (4). The utilization of ACO, inspired by the foraging behavior of ants, offers a decentralized and adaptive paradigm well-suited for the intricacies of cloud computing (5).

Cloud environments host diverse applications with varying computational requirements and priorities (6). Effective job scheduling ensures optimal allocation of resources, minimizes idle times, and maximizes the utilization of cloud infrastructure (7) (8). The dynamic nature of cloud workloads requires adaptive scheduling to meet evolving requirements and priorities. Task/job scheduling in a cloud environment is a crucial task due to its direct impact on resource utilization, performance optimization, and overall efficiency of cloud services (9) (10). It improves resource efficiency, cost-effectiveness, and cloud computing responsiveness (11).

This research explores the dynamic task allocation, workload adaptation, and goal optimization possible with ACO to shed light on the multi-objective nature of cloud-based job scheduling. The exploration of this integration is poised to contribute to the ongoing discourse on

advancing the efficiency and responsiveness of cloud services, aligning with the ever-evolving demands of modern computing environments (12).

2. Background

Ant Colony Optimization (ACO) is a metaheuristic algorithm inspired by the foraging behavior of ants. The idea is that, over time, the artificial ants converge towards better solutions by exploiting the information encoded in the pheromone trail and exploring new paths based on the heuristic information (13). It is commonly applied to solve combinatorial optimization problems, including the Traveling Salesman Problem (TSP). The TSP is a classic optimization problem where the goal is to find the shortest possible route that visits a set of cities and returns to the starting city (14).

In ACO for TSP, the algorithm simulates the foraging behavior of ants to find an optimal or near-optimal solution to the TSP. The basic idea is to model the problem in such a way that artificial ants build solutions by traversing the cities, depositing pheromones on the edges of the TSP graph, and updating the pheromone levels based on the quality of the answers (15).

The Traveling Salesman Problem (TSP) has various applications in the field of cloud computing, where optimization of resource allocation, data transfer, and overall system efficiency are crucial. In cloud computing, businesses can improve general performance, lower operational costs, and give end users better services by using TSP and other optimization methods. Following are some critical applications of TSP in cloud computing (2) (5) (16):

1. *Job Scheduling*: TSP is able to optimize the scheduling of jobs or tasks across several computer nodes in an environment that is characterized by cloud computing. A greater utilization of available resources and a shorter amount of time needed to finish work can be achieved via effective scheduling of work (17).
2. *Energy-Efficient Task Assignment:* Cloud data centers are responsible for a substantial amount of wasted energy. TSP is able to assist in optimizing the distribution of work among servers in a manner that results in the lowest possible amount of energy consumption (18). It is achieved by taking into factors such as the power consumption of the servers and the cooling needs.

3. *Data Migration and Replication:* In cloud computing, it may be necessary to duplicate data over numerous servers for the purposes of maintaining data integrity and distributing the processing load (19). TSP may be used to determine the most effective path for data replication or migration, hence reducing the total cost of the project as well as the amount of time that will be necessary.
4. *Network Design and Optimization:* TSP may be utilized to improve the physical architecture of network connections that are established between the various cloud infrastructure components (20). It might result in decreased latency, more incredible network speed, and overall improved communication across the many parts of the cloud ecosystem.
5. *Edge Computing Resource Allocation:* TSP may be utilized to optimize the distribution of computing resources at the edge, which is becoming increasingly crucial as edge computing becomes more prevalent (21). Edge computing refers to the practice of doing computations at locations that are physically closer to the sources of the data. This is of utmost significance for applications that require low-latency processing.
6. *Virtual Machine Placement:* TSP can be used to optimize the placement of virtual machines across physical servers in a cloud environment (22). Efficient order can lead to better resource utilization, reduced energy consumption, and improved overall system performance.

Adapting Ant Colony Optimization (ACO) to job scheduling in cloud computing involves a meticulous mapping onto the traditional Traveling Salesman Problem (TSP) framework. Initially, tasks or jobs are equated to cities, while edges signify feasible task sequences. The process unfolds with the placement of artificial ants on various towns and the initialization of pheromone levels on edges to express task sequence desirability.

Following is the mathematical model for Ant Colony Optimization (ACO) in the context of the Traveling Salesman Problem (TSP) (15) (14):

Decision Variables:

Let x_{ij} be a binary decision variable representing whether there is a connection (edge) between cities *i*and *j*.

Here, $x_{ij} = 1${if there is a connection between cities *i*and *j*; otherwise, 0.

Objective Function:

Minimize the total distance traveled (or minimize the sum of edge weights in the TSP).

$$\text{Minimize} \sum_i \sum_j c_{ij} x_{ij}$$

Constraints:

Each city must be visited exactly once:

$$\Sigma_j \, x_{ij} = 1 \text{ for all } i$$

Each city must be departed from exactly once:

$$\Sigma_i \, x_{ij} = 1 \text{ for all } j$$

Eliminate sub tours to ensure a connected tour:

$$u_i - u_j + n \cdot x_{(ij\leq)} n - 1, \quad \forall I \neq j, I, j \neq 0$$

Where u_i is a nonnegative continuous variable associated with each city.

Pheromone Update:

Update pheromone levels on edges:

$$\tau_{ij} \leftarrow (1 - \rho) \cdot \tau_{ij} + \Delta \tau_{ij}$$

Where τ is the pheromone evaporation rate, and $\Delta \tau_{ij}$ is the amount of pheromone deposited by ants.

This mathematical model captures the essence of ACO in solving the Traveling Salesman Problem, where ants construct solutions by selecting paths between cities, and pheromone levels on edges guide the exploration toward shorter paths over time.

3. Proposed Methodology

The proposed solution addresses the critical challenges of efficient resource allocation and task scheduling in cloud computing environments. It minimizes the overall execution time, maximizes resource utilization, and reduces costs. The proposed solution shown in Figure 1 investigates the intricate interplay of tasks, resources, and algorithmic parameters to

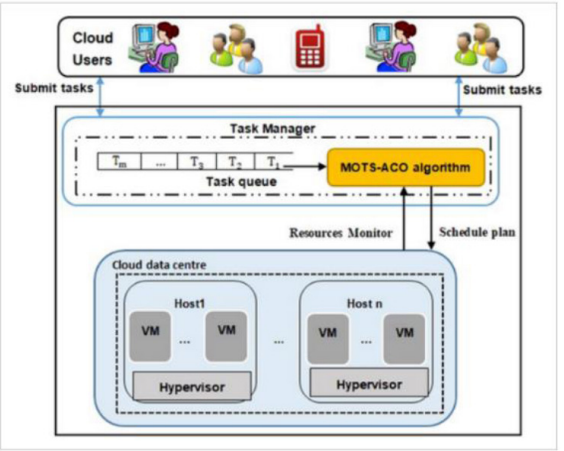


Figure 1

Proposed Solution

enhance the performance of cloud-based systems. Task and Resource Management Module represents a comprehensive task list with specific requirements. It encompasses the cloud infrastructure's virtual machines and their availability metrics. Also, VMs are mapped for optimal task distribution across available virtual machines.

4. Simulation and Experimental Setup

The experimental setup involves implementing an Ant Colony Optimization (ACO) algorithm for task scheduling in a cloud computing environment using MATLAB. The ACO algorithm is encapsulated in the `CloudSchedulingACO` function, which initializes parameters such as the maximum number of iterations, population size, and pheromone constant.

Figure 2 shows a cloud model created with random virtual machine/task coordinates, and the ACO algorithm is initialized with heuristic information and initial pheromone levels. The main loop iteratively moves ants, updates pheromones, and evaluates solutions based on the calculated cost, which represents the sum of distances between tasks. The algorithm aims to minimize this cost by refining task assignments over iterations. Results, including the best selection and its cost, are displayed at the end of the process. The experimental framework provides insights into how ACO can be applied to cloud task scheduling and its effectiveness in optimizing resource utilization and overall completion time.

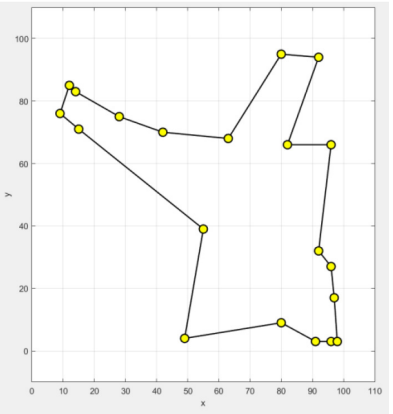


Figure 2

Mapping of Resources as X and Y vectors

5. Results and Discussion

The obtained results from the Ant Colony Optimization (ACO) for the Cloud Scheduling algorithm exhibit a consistent best cost of 351.8557 over the course of 550 iterations, as shown in Figure 3.

This stability in the best cost suggests that the algorithm converged to a solution early in the optimization process and maintained this optimal assignment throughout subsequent iterations. While the fixed best cost may indicate the algorithm's efficiency in finding a solution, it prompts further investigation into the nature of the assignment and the corresponding cloud scheduling configuration. The ACO algorithm's

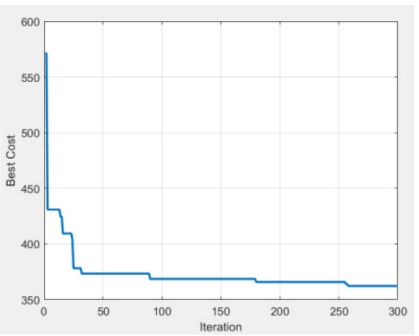


Figure 3

Best cost found with iterations

ability to converge quickly and consistently to a solution raises questions about the sensitivity of the algorithm to its parameter settings and the impact of the initial pheromone levels. Future research may explore the algorithm's behavior under varying parameters to uncover the trade-offs between exploration and exploitation, shedding light on the robustness and adaptability of ACO for cloud scheduling problems. Additionally, a detailed examination of the final task assignment and its associated cloud resource configuration would contribute to a deeper understanding of the algorithm's effectiveness in optimizing cloud service efficiency.

6. Conclusion

The incorporation of advanced optimization techniques becomes crucial in the cloud computing domain, where effective resource allocation and work scheduling are essential. The effectiveness of ant colony optimization converges to an optimal solution, emphasizing its efficiency in the distribution of tasks. This paper shows the comprehension of ACO's effectiveness in maximizing cloud service efficiency. Future research should explore parameter variations to understand trade-offs and enhance adaptability. However, the fixed best cost prompts further investigation into the algorithm's sensitivity to parameters and initial pheromone levels.

References

[1] Reddy GN, Kumar SP. MACO-MOTS: modified ant colony optimization for multi objective task scheduling in Cloud environment. *Int J Intell Syst Appl*. 11(1) : 73 (2019).

[2] Amer DA, Attiya G, Ziedan I. An efficient multi-objective scheduling algorithm based on spider monkey and ant colony optimization in cloud computing. Cluster Comput. 1–21 (2023).

[4] Singh KD, Singh P, Tripathi V, Khullar V. A Novel and Secure Framework to Detect Unauthorized Access to an Optical Fog-Cloud Computing Network. In: 2022 Seventh International Conference on Parallel, Distributed and Grid Computing (PDGC). p. 618–22 (2022).

[5] Singh KD. Particle Swarm Optimization assisted Support Vector Machine based Diagnostic System for Dengue prediction at the early stage. In: Proceedings - 2021 3rd International Conference on Advances in Computing, Communication Control and Networking, ICAC3N 2021. p. 844–8 (2021).

[6] Senthil Kumar AM, Venkatesan M. Multi-objective task scheduling using hybrid genetic-ant colony optimization algorithm in cloud environment. *Wirel Pers Commun.* 107 : 1835–48 (2019).

[7] Singh KD, Singh P, Kang SS. Ensembled-based Credit Card Fraud Detection in Online Transactions. In: AIP Conference Proceedings. p. 50009 (2022).

[8] Rathore PS, Chatterjee JM, Kumar A, Sujatha R. Energy-efficient cluster head selection through relay approach for WSN. *J Supercomput*. 77 : 7649–75 (2021).

[9] Matta P, Pant B, Tiwari UK. DDITA: A naive security model for IoT resource security. *Adv Intell Syst Comput.* 670 : 199–209 (2019).

[10] Islam MSU, Kumar A, Hu YC. Context-aware scheduling in Fog computing: A survey, taxonomy, challenges and future directions. *J Netw Comput Appl*. 180 : 103008 (2021).

[11] Tiwari P, Pant B, Elarabawy MM, Abd-Elnaby M, Mohd N, Dhiman G, et al. CNN Based Multiclass Brain Tumor Detection Using Medical Imaging. Comput Intell Neurosci [Internet]. 2022 [cited 2022 Dec 9];2022:1830010. Available from: https://www.hindawi.com/journals/cin/2022/1830010/.

[12] Singh KD, Singh P, Chhabra R, Kaur G, Bansal A, Tripathi V. Cyber-Physical Systems for Smart City Applications: A Comparative Study. In: 2023 International Conference on Advancement in Computation \& Computer Technologies (InCACCT). p. 871–6 (2023).

[13] Taneja H, Kaur S. Fake feedback detection to enhance trust in cloud using supervised machine learning techniques. In: Proceedings of Data Analytics and Management: ICDAM 2021, Volume 2. p. 789–96 (2022).

[14] Taneja H, Kaur S. Reputation based novel trust management framework with enhanced availability for cloud. *J Parallel Distrib Comput.* 178 : 43–55 (2023).

[15] Kruekaew B, Kimpan W. Multi-objective task scheduling optimization for load balancing in cloud computing environment using hybrid artificial bee colony algorithm with reinforcement learning. IEEE Access. 10 : 17803–18 (2022).

[16] Elsedimy E, Algarni F. MOTS-ACO: An improved ant colony optimiser for multi-objective task scheduling optimisation problem in cloud data centres. IET Networks. 11(2) : 43–57 (2022).

[17] Dhiman P, Kukreja V, Manoharan P, Kaur A, Kamruzzaman MM, Dhaou I Ben, et al. A novel deep learning model for detection of severity level of the disease in citrus fruits. *Electronics.* 11(3) : 495 (2022).

[18] Singh PD, Singh KD. Security and Privacy in Fog/Cloud-based IoT Systems for AI and Robotics. EAI Endorsed Trans AI Robot. 2 (2023).

[19] Singh P, Singh KD. Fog-Centric Intelligent Surveillance System: A Novel Approach for Effective and Efficient Surveillance. In: 2023 International Conference on Advancement in Computation \& Computer Technologies (InCACCT). p. 762–6 (2023).

[20] Singh KD. Securing of Cloud Infrastructure using Enterprise Honeypot. In: Proceedings - 2021 3rd International Conference on Advances in Computing, Communication Control and Networking, ICAC3N 2021. p. 1388–93 (2021).

[21] Singh P, Singh KD, Tripathi V, Chaudhari V. Use of Ensemble Based Approach to Predict Health Insurance Premium at Early Stage. Proc Int Conf Comput Intell Sustain Eng Solut CISES 2022 [Internet]. 2022 [cited 2022 Dec 2];566–9. Available from: https://ieeexplore.ieee.org/abstract/document/9844398/.

[22] Singh KD, Singh PD, Bansal A, Kaur G, Khullar V, Tripathi V. Exploratory Data Analysis and Customer Churn Prediction for the Telecommunication Industry. In: 2023 3rd International Conference on Advances in Computing, Communication, Embedded and Secure Systems (ACCESS). p. 197–201 (2023).

[23] S. Tiwari, S. Kumar, and K. Guleria, "Outbreak Trends of Coronavirus Disease-2019 in India: A Prediction," Disaster Med. Public Health Prep., vol. 14, no. 5, pp. e33–e38 (2020), doi: 10.1017/dmp.2020.115.