Journal of Soft Computing Paradigm (ISSN: 2582-2640)
www.irojournals.com/jscp/

Cost Efficient Resource Provisioning using ACO

Suriya S.[1], Madhvesh V S.[2], Mrudhhula V S[3]

[1]Associate Professor, Department of Computer Science and Engineering, PSG College of Technology, Coimbatore, Tamil Nadu, India.

[2]PG Scholar, Department of Computer Science and Engineering, PSG College of Technology, Coimbatore, Tamil Nadu, India.

[3]PG Scholar, Department of Computer Science and Engineering, PSG College of Technology, Coimbatore, Tamil Nadu, India.

E-mail: [1]suriya84@gmail.com, ss.cse@psgtech.ac.in, [2]23mz38@psgtech.ac.in, [3]23mz39@psgtech.ac.in

Abstract

Cloud computing has revolutionized the way computational resources are provisioned and managed, offering scalable and flexible services to meet diverse user demands. However, cost-effective resource management is a very challenging process because of the dynamism and diversity of the aspects of cloud environments that changes in terms of load and resources. The traditional sources of resource acquisition do not have the capacity to deliver the alternatives expected on their cost without having a negative impact on the performance of the resources. This work describes the new approach of utilizing the ACO for resource management in cloud computing. The method that is proposed contains the potential to incorporate pheromone-based heuristics for controlling the process of resource allocation such that reduced operational costs are ensured as well as the performance of the process is maintained at the optimal rate. ACO explains the behaviour of the search process where the allocation of the tasks is done based on the values of the pheromone trails and the heuristic information. An ACO model that includes dynamic measurements for the diverse cloud environment and several adaptive mechanisms for creating more tasks and virtual machines (VMs) can be considered a helpful solution for actual cloud applications. The results of the experiments are high in terms of cost-effectiveness compared to other approaches and reflect the ACO's ability to function in dynamic cloud environments.

DOI: https://doi.org/10.36548/jscp.2024.4.003
Received: 27.11.2024, received in revised form: 20.12.2024, accepted: 03.01.2025, published: 11.01.2025

Keywords: Cloud Computing, Resource Provisioning, Cost Efficiency, Heuristic Information, Ant Colony Optimization (ACO).

1. Introduction

Cloud computing has transformed the landscape of computing by providing scalable and on-demand access to a wide array of computational resources. As such it is subscription-based and provided on a need basis that is provisioned allowing the consumers/customers to increase and decrease the capacity of the resources according to their needs and requirements automatically and on needs of demand basis according to workload or load. But there are some more points, in which the pay-as-you-go model of cloud computing has some merits. However, the interesting and highly flexible nature of the cloud environment means that there are certain constraints on how resources can be allocated for the cloud environment. The framework also seeks to incorporate some resource planning that is supposed to lower the operations cost towards cost or performance or reliability. This means that the computation power shall be measured to the required levels of computation and ensure optimal task storage/exploitation while not overshooting the number of tasks that exist at any given time to address the resource costs or QoS. The static allocation techniques and the simple heuristics that are rule based which is therefore considered to be rigid as non-chaotic approaches to the resource provisioning for cloud works because the former approaches cannot deal with the chaotic nature or the randomness and uncertainty associated with such cloud works. Yet, the essential problem of translating flexibility into cost-effective resource deployment in constantly changing settings remains acute. There is the problem of resource wastage or failure to meet service levels arising from the traditional approach’s inflexibility. To overcome these limitations, the study put forward an attempt to extend the execution of the present heuristic approach, namely, Ant Colony Optimization (ACO), which can be optimized in an iterative manner to handle cloud resources. ACO is chosen because of its nature-inspired algorithm which is good at the exploration-exploitation trade-off in optimization tasks. ACO’s mechanism for the adaptive pheromone works well with the need for cost-effective operation that is needed to provide for the dynamic workload requirements as well as reliability.

1.1 Ant Colony Optimization (ACO)

ACO is a Metaheuristic Algorithm that works with fluidic behaviours as an inspiration to ants. ACO approach has already proven its effectiveness in solving optimization problem.

This works just like ants doing blazing which means that the ants mark their trail to find the shortest path to food supply and then they would go over the trail to indicate the shortest path to the food supply as per their nature. This follows the effective formulation of decisions through this group behaviour and the attainment of solutions at the right time. ACO can be used to solve the resource provisioning issue in the Cloud by taking the allocation of tasks as a path-finding problem. Key characteristics of ACO including iterative optimization and heuristic adaptability make it suitable to meet the cloud provisioning demand where optimization of cost while maintaining the optimum performance is desirable. The algorithm is one of the heuristics and employs the pheromone-based mechanism for finding the low cost solution, where the pheromone indicates the attractivity of the assignment of certain tasks to certain VMs in the past. The heuristic information, which can be acquired by cost metrics, allows the evaluation of the attractiveness of various allocation options. The ACO, where the problem of task allocation is modelled as path finding in the context of cloud computing. Pheromone levels dictate the preference for allocating tasks to certain VMs, whereas heuristic information derived from the costs and measures of available resources guide decision making. This back-and-forth process allows for finding near-optima for changed cloud environments with reasonable accuracy. The combination of balancing the exploration of new solutions and the exploitation of known good solutions allows ACO to gradually develop a better allocation solution until it approaches an optimal solution for resource allocation and achieves a minimum cost.

2. Related Work

Marco et al.[1] is a well-written introduction to ACO and its various forms and applications. The approaches and mechanisms covered in this section can be used to solve a wide range of optimization problems, including resource provisioning in cloud computing, using ACO. Oda et al. [2] deals with dynamically allocating resource provision in cloud infrastructure to ensure scientific workflow execution in varying conditions of the cloud platform and varying resource demand. The key challenges are the resource requirements shift due to high-performance uncertainty and unpredictability of the platform. It proposes a scheduling method that inculcates the principles of scientific workflows with cloud elasticity and aims to minimize the workflow execution time using a given budget. While using the dependencies approach, the algorithm prioritizes activities based on the dependencies among them. Plans the activities using an Earliest Finish Time (EFT) approach, the resources are allocated dynamically. Experiment results find that algorithm performance is not highly affected by performance variations in cloud platforms, and the time and cost of workflow process aren't seriously fluctuated. Furthermore, it also demonstrates the possibilities of upgrading parallel computation and workflow performance with budget constraint. Shen et al [3], considers a cost-optimized method of provisioning resources in cloud computing applications that address the problem of generating the best economic solution for application providers. The suggested approach exploits prediction of workloads to find a correlation between workload level and the number of virtual machines (VMs) needed. Following this, it employs pricing policies of numerous cloud providers to put together a hybrid resource allocation plan, combining instances of different prices as well as leasing periods. The prototype results displays this technique is enough to meet the cost-saving objective, and minimizes both SLA violation and resource waste, thus confirming its practicability and efficiency in the management of cloud resources. Moreover, in the document, the researches related to workload prediction, resource provisioning, and cloud pricing are also provided which foster the understanding and innovations of cost-effective cloud computing methods. Ma et al. [4], the presented framework about the Cost-Efficient Resource provisioning for dynamic requests at Cloud Assisted Mobile Edge -based (CAME) system. It has algorithms that will be used to optimize the allocation processes by using prices of computational resources at edge hosting and cloud instances and mobile requests' arrival rates. ORP algorithms capitalize on the flexible charging fares and mobile orders came in a fast rate to reach the optimal resource movement. In comparison to the approaches including local-first and cloud-first algorithms, the algorithms with ORP show better cost and performance features, among others, interesting in a situation when the nature of mobile requests varies or has tight time limit. Optimal provisioning of cloud resources, utilizes Ant Colony Optimization (ACO) as a heuristic algorithm for providing the best resource alternative given its priority for a user application. Chaisiri et al [5], proposed the OCRP (Optimal Cloud Resource Provisioning) algorithm applied in resources provisioning. It proposes an optimal resource allocation from numerous cloud providers. It takes into account multiple uncertainties like price fluctuations and demand variation to achieve the best use of resource reservation and on-demand allocation. The algorithm makes use of stochastic integer programming mixed with Benders decomposition and Sample Average Approximation (SAA) principles to solve the complicated nature of uncertainties while constantly preserving the optimal balance into on-demand and reservation resource provisioning. This optimization will in the end help cloud users revise their costs considerably without compromising on the capacities they require to run operations. Yousefyan et al. [6], put forth a cost-efficient approach for cloud computing provisioning using the Imperialist Competitive Algorithm Optimization process (CECRPICAO). It addresses choosing between both reservation and on-demand payment plans offered by cloud vendors. Our process includes a demand predictor algorithm for future demand forecasting, and the optimization of resource allocation using the imperialist competitive algorithm. The simulations determine the performance of CECRPICAO and show that it is cost saving, compare to the past approaches such as stochastic integer programming (SIP), evolutionary optimal virtual machine placement (EOVMP), and others. The simulations reveal that CECRPICAO has the best results in a shorter term time and at a lower cost via the reservation, utilization, and on-demand phases. Furthermore, the document points out the accurateness of the MLP network with error backpropagation learning in the demand prediction compared to other prediction systems like the Double Exponential Smoothing (DblSES) and Markov Chain (MC). Zuo et al. [7], proposes multi-objective optimization of ACO for task scheduling in cloud computing. It shows how ACO can deal with different and sometimes opposing requirements like cost and performance and is therefore applicable to our resource cost-efficiency problem. Sharma et al. [8], proposed ACO as a solution to resource management in IoT based Cloud computing environment which encompasses issues such as energy utilization and QoS. Evaluation outcomes also reveal that ACO is far more effective in identifying the best strategies for resource allocation, energy management and for increasing network throughput and decreasing latency. Zhan et al. [9], strives to present the evolution of resource scheduling strategies in cloud computing and their connection to the evolutionary algorithm ACO. It talks about the advantages of ACO in managing cloud environments that are dynamic in nature along with the better management of resource allocation. Buyya et al.[10], is aimed at discussing the importance of energy-efficient resource management in cost-efficient provisioning for data centres. It takes into consideration various optimization techniques and their application to energy management, which in turn is related to the cost optimization question. Xia et al[11] suggests the new approach for overcoming the drawbacks of the basic ACO algorithm for cloud computing resource scheduling regarding higher convergence efficiency and avoiding local optima traps. Analysis of performance evaluations reveal enhanced publication of load balancing, utility value, and scheduling, which creates further innovations in cloud resource management. Rajasekaran et al. [12] traces issues of resource allocation in cloud computing and presents a solution, the HMAO technique from GA and MAO for optimal resource utilization and minimized bandwidth expense. To facilitate a practical implementation it employs Decentralized MAO algorithm and uses quality assessment in terms of Taguchi method. The study also notes cost optimization as the best solution for achieving cheaper cloud resource acquisitions. Gupta et al. [13] discusses the problems of applying task scheduling in cloud computing with the help of traditional algorithms and offers a new practically-oriented ACO-based model to consider networking and execution costs. This works has revealed its effectiveness in terms of the time it takes to accomplish tasks and the cost implications highlighting its real world applicability in managing cloud resources. To reduce the energy consumption and SLA violations in cloud computing, Usha et al.[14] presents a new algorithm: the Enhanced Ant Colony Optimization (EACO). CloudSim experiments demonstrate that EACO has lower energy usage by 41-44% than ordinary ACO while providing a highly capable method.

3. Methodology

3.1 Problem Formulation

This problem of allocating resource in cloud environment is basically the problem of allocating a set of tasks to a set of VM's so that the minimum cost of resource is met subject to some performance constraints. From the mathematical viewpoint, let the set of tasks be $T = \{t_1, t_2, \ldots, t_n\}$ and $V=\{v_1, v_2, \ldots v_m\}$ represents the set of VMs. Each task t_i has some specific requirements and time for the execution, whereas the VM v_j has some capacity and cost per unit of resource usage. The objective is to find a mapping $\pi: T \rightarrow V$ that minimizes total cost.

$$\min \sum_{i=1}^{n} \sum_{j=1}^{m} c_{ij} . x_{ij} \quad (1)$$

3.2 Optimizing ACO

ACO promotes the process of specifying an objective function that determines the degree of cost efficiency that might include variables such as the utilization of the resources, the use of energy, the fulfilment of SLAs, and the cost of the resources. In the case of resource allocation, this can be a mapping of tasks/jobs to VMs, which shows which VM is responsible for which task/job. Ants directly mark the path depending on the quality of a solution they were able to create. The cost efficiency of the path is higher, the more pheromone will be deposited onto it. Ants synthesise solutions and analyse their cost of utility in terms of the objective function. Cost efficiency is used to eliminate solutions that do not meet the criteria. Pheromone

ISSN: 2582-2640

trails on the ACO system get reduced as generations increase probably to simulate natural pheromone decay. This helps in carrying out suitable variation of solutions over time.

3.3 Heuristic Information

Adaptive information is important in ensuring that the ants arrive at the best and efficient solutions. Different values of heuristic information η_{ij} can be obtained from the cost of assigning task t_i to VM v_j within ACO. To reinforce the drive toward choosing a lower cost for the task assignment, the algorithm switches to $\eta_{ij} = 1/c_{ij}$ instead of $\eta_{ij} =1$.

3.4 Simulation Details

The effectiveness of the proposed algorithm is demonstrated in experimental evaluation where the simulated cloud platforms had different levels of workloads and availability of resources. The simulation was carried out using CloudSim toolkit, which is one of the most established simulation platforms for cloud computing environment. The parameters are

Pheromone importance (α): Weight of pheromone influence.

Heuristic importance (β): Impact of resource cost on decisions.

Evaporation rate (ρ): Evaporation rate of pheromone trails.

Pheromone quantity (Q): Pheromone deposited based on solution quantity.

The diverse conditions like Different levels of workload intensities, from low to high, were tested, Varying numbers of VMs, each with unique capacities and cost structures, were utilized and Dynamic changes in resource availability and task arrival rates were introduced to simulate real-world cloud environments.

The experiments compare the performance of ACO with Greedy and Random search algorithms in terms of total cost, execution time, and resource organizing.

3.5 Algorithm Design

Algorithm: Optimized ACO algorithm.

Output: Tasks allotted with low cost

1: Initialize pheromone levels τ_{ij}

2: Assign α, β, ρ, and Q

3: Set Heuristic information $\eta_{ij} = 1/c_{ij}$

4: Loop for k-ants

5: Ant constructs a solution by probabilistically assigning tasks to VMs.

$$p_{ij} = \frac{[\tau_{ij}]^{\alpha}[\eta_{ij}]^{\beta}}{\sum_{k \in v} [\tau_{ik}]^{\alpha}[\eta_{ik}]^{\beta}} \quad (2)$$

6: The total cost of the solution constructed by each ants are calculated iteratively

7: Reduce pheromone levels for evaporation rate $\tau_{ij} = (1-\rho)\tau_{ij}$

8: Increase pheromone levels based on quality of solutions $\Delta\tau_{ij}$

9: Best Solutions are found.

3.6 Flow Process

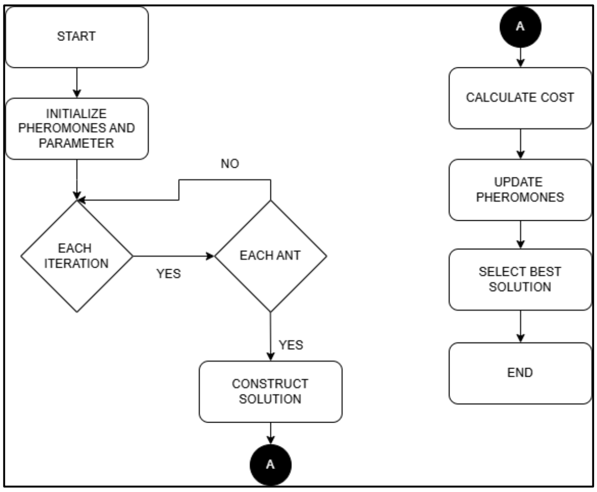

Figure 1. Flow Process of optimized ACO

The complete procedure (Figure 1) starts with the initialization of the ACO algorithm

1) Initialize Pheromones and parameters for Simulated Annealing and Ant Colony Optimization.

a) Pheromone Initialization: set the initial pheromone levels τ_{ij} for all possible assignments of tasks to VMs.

b) Parameter Setting: Select the ACO parameters including α, β, ρ, and Q.

2) The algorithm repeats until the specified number of steps or until a particular level of accuracy is achieved.

3) In each iteration, multiple ants (agents) construct solutions simultaneously.

4) Each ant constructs a complete solution by assigning tasks to VMs based on the probability p_{ij}, which is influenced by the pheromone levels τ_{ij} and heuristic information η_{ij} (2).

5) The total cost of the solution will be deducted from the solution after designing a solution, which encompasses all the cost of all the optimal task-VM assignments performed by the ant.

6) Evaporation: Use a negative sum of pheromone on all paths to simulate evaporation to ensure that the algorithm does not find too quickly, that is sub-optimal solution.

Deposition: When an ant finds a solution of high quality (low cost) on some of the paths of the multi-dimensional labyrinth, this path is reinforced by re-increasing the pheromone level $\Delta\tau_{ij.}$

7) Whenever the ants are finished with the given number of iterations, the algorithm picks up the best solution among the ants. This setting is the most appropriate and cost effective configuration of the workload tasks across the allocated VMs for resource management.

The process ends with the minimum resource allocation possibility calculated by the ACO algorithm.

4. Experimental Results and Discussion

The experiments were performed to test ACO based algorithm for resource provisioning using a simulated cloud environment. The cloud environment included several tasks (100) with different levels of resource needs (low, medium , and high) and several VMs (10 -20) with multiple sizes and prices. The cost matrix Cij, representing the cost of assigning task t to VM vj, was generated to simulate various cost structures and workload scenarios. The ACO parameters, including pheromone importance (α), heuristic importance (β), pheromone evaporation rate (ρ), and the quantity of pheromone deposited (Q), were tuned through preliminary experiments to achieve optimal performance.

The performance of the ACO algorithm was compared with the conventional resource provisioning methods such as Greedy [3] and Random search [5]. The primary metric for evaluation was the total cost, which referred to the cost of the entire task-VM assignment process. Other important measurements like execution time and resource usage were also incorporated to assess the efficiency of the algorithm.

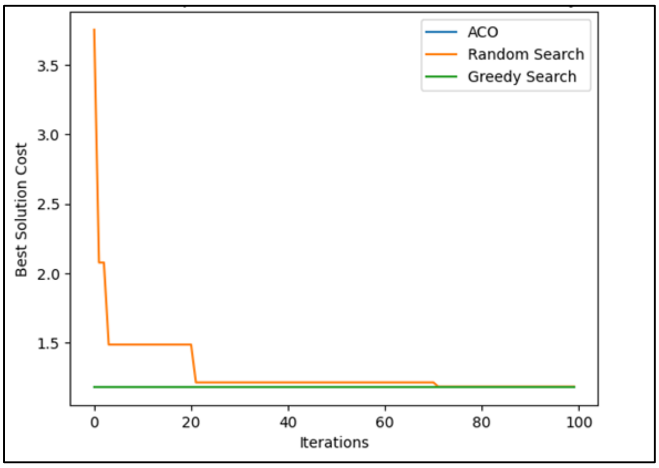

Figure 2. Performance Comparison

```
ACO Best Solution: [0, 2, 1, 2, 2]
ACO Best Solution Cost: 1.180233100039001
Random Search Best Solution Cost: 1.1825683451765254
Greedy Search Best Solution Cost: 1.180233100039001
```

Figure 3. Best Low Cost Solution using Optimized ACO

The results in Figures 2 and 3 shows the superiority of the ACO-based method in its cost efficiency compared to the conventional methods. As simulation results reveal ACO's ability to work under various conditions of simulated cloud environments, the proposed method could be considered ready to address real-world conditions.

The comparison metrics include:

Total Cost: This is a general expense of the resources that has been allocated in the organization.

Execution Time: Duration of task completion.

Resource Utilization: Capacity to use resources effectively.

The ACO algorithm reported a cost saving when compared to Random search. Due to the optimization of solutions in each iteration, ACO cut out the unnecessary expenses while achieving the best results concerning the resources usage. This increase in cost efficiency can be explained in terms of the fact that the algorithm proposes solutions that are in a much larger search space than the traditional algorithms and also finds near-optimal solutions through the iterative process of pheromone updating and heuristic guidance with the cost associated with the shortest path. Flexible management of pheromone concentration enabled to control the algorithm performance depending on the profile of the current workload and resources for the cloud environment.

5. Conclusion

The present research proves that the use of ACO could be beneficial in rational optimization of resource provisioning within cloud computing environments to minimize costs, while reducing expenditure and optimizing performance. The simulation experiments show that ACO is quite effective in changing workloads, which confirms the effectiveness of this solution in inefficient cost-oriented resource management. Nevertheless, future studies must invest in ACO parameters' optimization, combining ACO with other optimization approaches, as well as scalability for Big Data in cloud environments. Therefore, the testing of the algorithm on real cloud infrastructures is suggested to reveal its benefits and logical weaknesses.

References

[1] Dorigo, Marco. "Ant colony optimization." *Scholarpedia* 2, no. 3 (2007): 1461.

[2] Oda, Ricardo, Daniel Cordeiro, and Kelly Rosa Braghetto. "Dynamic resource provisioning for scientific workflow executions in clouds." In 2018 IEEE International Conference on Services Computing (SCC), pp. 291-294. IEEE, 2018.

[3] Shen, Yuxi, Haopeng Chen, Lingxuan Shen, Cheng Mei, and Xing Pu. "Cost-optimized resource provision for cloud applications." In 2014 IEEE Intl Conf on High Performance Computing and Communications, 2014 IEEE 6th Intl Symp on Cyberspace Safety and Security, 2014 IEEE 11th Intl Conf on Embedded Software and Syst (HPCC, CSS, ICESS),Paris, France pp. 1060-1067. IEEE, 2014.

[4] Ma, Xiao, Shangguang Wang, Shan Zhang, Peng Yang, Chuang Lin, and Xuemin Shen. "Cost-efficient resource provisioning for dynamic requests in cloud assisted mobile edge computing." IEEE Transactions on Cloud Computing 9, no. 3 (2019): 968-980.

[5] Chaisiri, Sivadon, Bu-Sung Lee, and Dusit Niyato. "Optimization of resource provisioning cost in cloud computing." IEEE transactions on services Computing 5, no. 2 (2011): 164-177.

[6] Yousefyan, Shahla, Amir Vahid Dastjerdi, and Mohamad Reza Salehnamadi. "Cost effective cloud resource provisioning with imperialist competitive algorithm optimization." In The 5th Conference on Information and Knowledge Technology, Shiraz, Iran pp. 55-60. IEEE, 2013.

[7] Zuo, Liyun, Lei Shu, Shoubin Dong, Chunsheng Zhu, and Takahiro Hara. "A multi-objective optimization scheduling method based on the ant colony algorithm in cloud computing." Ieee Access 3 (2015): 2687-2699.

[8] Sharma, Disha, and Arjun Singh Parihar. "Ant Colony Optimization-Based Energy Efficiency Optimization in IoT-Enabled Cloud Computing Resource Allocation." In 2024 International Conference on Advances in Computing Research on Science Engineering and Technology (ACROSET), Madhya Pradesh, India pp. 1-8. IEEE, 2024.

[9] Zhan, Zhi-Hui, Xiao-Fang Liu, Yue-Jiao Gong, Jun Zhang, Henry Shu-Hung Chung, and Yun Li. "Cloud computing resource scheduling and a survey of its evolutionary approaches." ACM Computing Surveys (CSUR) 47, no. 4 (2015): 1-33.

[10] Buyya, Rajkumar, Anton Beloglazov, and Jemal Abawajy. "Energy-efficient management of data center resources for cloud computing: a vision, architectural elements, and open challenges." arXiv preprint arXiv:1006.0308 (2010).

[11] Xia, Chenyue, Rui Wang, Zhuofu Deng, and Yingnan Zheng. "Research on cloud computing resource scheduling based on improved ant colony optimization algorithm." In 2022 4th International Conference on Frontiers Technology of Information and Computer (ICFTIC),China pp. 295-298. IEEE, 2022.

[12] Rajasekaran, P., M. Pradeeshwar, and Puneeth Kumar RC. "Resources Provisioning Cost Optimization in a Decentralized Cloud Firewall Framework." In 2022 1st

International Conference on Computational Science and Technology (ICCST),CHENNAI, India pp. 311-315. IEEE, 2022.

[13] Gupta, Punit, Ujjwal Goyal, and Vaishali Verma. "Cost-aware ant colony optimization for resource allocation in cloud infrastructure." Recent Advances in Computer Science and Communications (Formerly: Recent Patents on Computer Science) 13, no. 3 (2020): 326-335.

[14] SP, Usha Kirana. "Enhanced Ant Colony Based VM Selection and Consolidation for Energy Conservation." International Journal of Innovative Technology and Exploring Engineering 13, no. 11.