Ant Colony Optimization for Service Composition in Multi-cloud Environments

[1]Mahesh Maurya,
Department of Computer Engineering,
K.C. College of Engineering,
Mith Bunder road, Kopri, Thane East,
Maharashtra, India.
mahesh.maurya@kccemsr.edu.in

[2]Vidhya G,
Department of Computer Science and Engineering,
Sona College of Technology,
Salem, Tamilnadu, India.
gvidhyacse@gmail.com

[3]Faiz Akram,
Department of Computer Science,
Faculty of Computing and Informatics,
Jimma Institute of Technology,
Jimma University,
Jimma, Oromia Region, Ethiopia.
akram.faiz@ju.edu.et

[4]Saravanan M,
Department of ECE,
Sri Eshwar College of Engineering,
Coimbatore, Tamilnadu, India.
saranecedgl@gmail.com

[3]Venkata Ramana K,
Department of CSE,
QIS College of Engineering and Technology,
Ongole, Andrapradesh, India.
ramanakaveripakam@gmail.com

[6] Rajakumari S
Assistant Professor Department of Mathematics
R.M.D. Engineering College
rajakumari.snh@rmd.ac.in

2024 15th International Conference on Computing Communication and Networking Technologies (ICCCNT) | 979-8-3503-7024-9/24/$31.00 ©2024 IEEE | DOI: 10.1109/ICCCNT61001.2024.10725778

Abstract*— the *Ant Colony Optimization (ACO) algorithm is a famous metaheuristic technique that has been successfully applied in various optimization issues. It mimics the foraging conduct of ants to locate the shortest path among meals assets and their colony. Recently, ACO has been increasingly more applied for provider composition in multi-cloud environments. This involves the composition of offerings from different cloud carriers to satisfy the service necessities of a person. By way of using ACO, services can be correctly selected and composed from a couple of carriers, considering elements including cost, excellent, and availability. ACO has proven promising outcomes in enhancing the overall performance and reliability of carrier composition in multi-cloud environments, making it a precious technique in todays tremendously distributed computing landscape.

Keywords*— Service Composition, Multi-cloud Environments, Distributed Computing, Resource Management, Quality of Service (QoS)

I. INTRODUCTION

Ant Colony Optimization (ACO) is a metaheuristic algorithm stimulated by means of the collective foraging behavior of ants. it has been broadly used in fixing diverse optimization troubles and has gained large attention in the discipline of service composition in multi-cloud environments. on this essay, we are able to speak the importance of ACO for provider composition in multi-cloud environments [1]. carrier composition is the method of combining a couple of unbiased offerings to create a brand new composite service that meets particular person requirements. In multi-cloud environments, one of a kind cloud providers offer a variety of offerings, and the composition of these offerings can lead to new functionalities and fee optimization. however, provider composition in multi-cloud environments brings its own set of demanding situations, along with heterogeneous provider portfolios, various fine of provider (QoS) tiers, and one-of-a-kind pricing models. those challenges make it difficult for customers to select the superior composition of offerings to satisfy their necessities [2]. ACO provides a approach to this problem by way of offering an adaptive and efficient approach for carrier composition. It mimics the conduct of ants in finding the highest quality direction to food sources, wherein the person movements of the ants together lead to a global answer. Ant Colony Optimization (ACO) is a famous metaheuristic set of rules stimulated by way of the foraging behavior of ants. This modern algorithm has been extensively implemented in diverse fields, together with carrier composition in multi-cloud environments. one of the main innovations of ACO for carrier composition is its ability to correctly address the challenges posed through the distinctly dynamic and heterogeneous nature of multi-cloud environments [3]. In traditional service composition strategies, predefined workflows are used to compose services, however in multi-cloud environments, offerings are continuously changing and new services are being delivered. conventional methods warfare to evolve to those modifications, while ACO is able to dynamically modify and optimize service compositions based at the current country of the surroundings [4]. every other innovation of ACO for carrier composition is its capability to address the complexity of multi-cloud environments. With multiple provider companies supplying similar offerings, it could be hard to discover the most useful composition that fulfills the user's necessities. ACO addresses this difficulty through the use of pheromones to guide the ants toward the maximum promising services and answers. This lets in for a extra green search for the most reliable composition, even in extraordinarily complex environments [5].

Ant Colony Optimization (ACO) gives a dynamic and scalable approach to carrier composition in multi-cloud environments. It makes use of a population-based search method, in which multiple ants work together to solve the composition trouble. This allows for a flexible and adaptive composition procedure, making it appropriate for handling modifications inside the environment, together with the addition or removal of services. ACO can be used to optimize the choice of offerings in a multi-cloud environment. via leveraging the pheromone path and heuristic facts, ants are capable of discover the maximum suitable services to include in the composition, considering factors together with carrier first-class, availability, value, and user choices [6]. Multi-cloud environments are extraordinarily dynamic and unsure, with continuously changing service availability, performance, and pricing. ACO offers a strong technique to these challenges by using constantly updating the pheromone path primarily based on the achievement or failure of previous compositions. This permits the system to conform to modifications and enhance the satisfactory of provider compositions over time [7].

ACO faces difficulties in handling dynamic and unpredictable environments, which might be commonplace in multi-cloud situations. As services may additionally end up unavailable or showcase various stages of overall performance over time, ACO might also fail to discover most appropriate or maybe feasible solutions. that is because the pheromone trails may additionally turn out to be out of date or deceptive, leading the ants to make suboptimal decisions. One key novelty of using Ant Colony Optimization (ACO) for service composition in multi-cloud environments is its capacity to conform to changing environments. ACO is inspired by way of the foraging conduct of ants, who go away a path of pheromones to manual their peers toward the best food resources. in addition, ACO algorithms work through depositing and updating pheromone stages on paths connecting offerings, indicating their satisfactory and popularity [8]. This lets in the ACO algorithm to dynamically modify carrier composition based on the present day nation of the multi-cloud environment, making sure most desirable performance even in exceedingly dynamic and unpredictable cloud environments. This adaptive nature of ACO sets it other than different techniques, including Genetic Algorithms or Particle Swarm Optimization, which won't be capable of handle the constantly changing nature of multi-cloud environments [9].

II. Related Works

one of the fundamental challenges in service composition in multi-cloud environments is the difficulty in finding the ultimate composition of services to satisfy a specific request. that is where the concept of Ant Colony Optimization (ACO) comes into play. ACO is a swarm intelligence-based optimization set of rules stimulated by means of the foraging conduct of ants. it's been efficaciously implemented in numerous optimization issues, inclusive of carrier composition in multi-cloud environments [10] but, there are several problems that want to be addressed when the usage of ACO for carrier composition. firstly, ACO calls for a large number of iterations to converge to an most fulfilling solution, that could lead to high computational fees. that is because ACO relies on the stochastic motion of ants to explore exceptional provider compositions and converge to the most beneficial one. Secondly, a single ACO set of rules won't be appropriate for all types of provider composition problems in multi-cloud environments [11]. every environment has unique characteristics, which include various carrier services, pricing fashions, and nice of service (QoS) necessities. therefore, a customized ACO algorithm is wanted for every environment, which may be time-ingesting and resource-intensive [12]. Ant Colony Optimization (ACO) is a famous metaheuristic algorithm, inspired by way of the foraging conduct of ants that has been applied to clear up numerous combinatorial optimization troubles. In recent years, ACO has been increasingly more used for provider composition in multi-cloud environments, which involves choosing and combining more than one cloud offerings to fulfill a particular person request [13]. at the same time as ACO has shown promising consequences in carrier composition, it also faces several challenges and troubles in this context [14-24]. one of the essential problems of ACO in carrier composition is the scalability issue. because the variety of offerings and the complexity of the composition problem growth, the performance of ACO deteriorates considerably. that is because of the fact that ACO is based heavily on constructing and updating pheromone trails based totally on nearby interactions among the ants, which turns into inefficient in large and complex environments [15].

III. Proposed Model

The Ant Colony Optimization (ACO) is a metaheuristic set of rules stimulated by way of the foraging behavior of ants. it has been effectively implemented to diverse optimization problems, which includes carrier composition in multi-cloud environments [16].

A. Construction

Ant Colony Optimization (ACO) is a nature-stimulated metaheuristic algorithm that has been broadly used in solving combinatorial optimization issues in diverse fields. In latest years, ACO has also been applied in the domain of cloud computing for carrier composition in multi-cloud environments [17]. The goal of provider composition is to select the excellent aggregate of cloud offerings to meet the consumer's request even as meeting their nice of carrier (QoS) necessities [18]. the development of an ACO algorithm for service composition in multi-cloud environments includes modeling the problem as a graph where nodes represent the to be had offerings and edges constitute the viable compositions [19]. The construction of proposed model have shown in the following fig.1

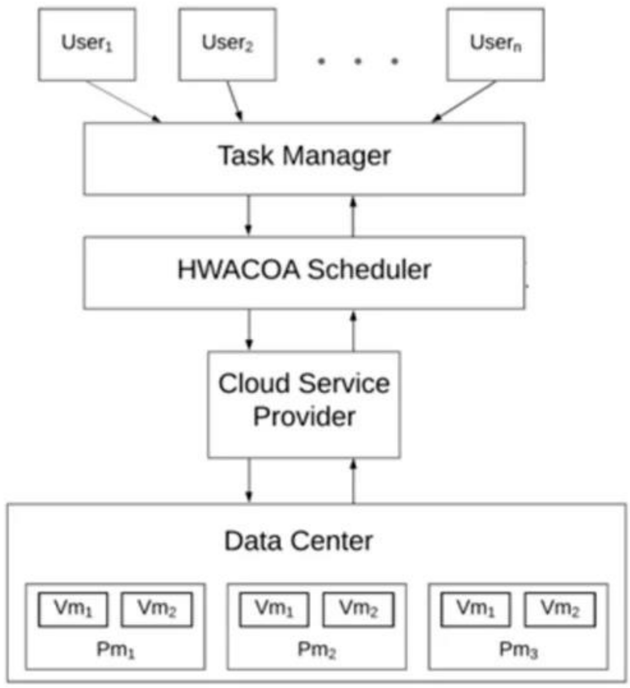

Fig.1: Construction of proposed model

A digital device (VM) is a software-based totally illustration of a bodily system, allowing multiple working systems and packages to run on a unmarried physical device [20]. The VM is accountable for handling the hardware resources of the physical gadget, together with the processor, reminiscence, and disk garage [21]. The venture supervisor is a device thing responsible for handling the processes and duties running at the VM. It guarantees that obligations and strategies are allotted the essential sources and that they're done effectively [22]. The assignment manager is likewise answerable for monitoring the performance of the VM and identifying any bottlenecks or resource constraints [23]. The HWACOA scheduler is a specialized scheduler designed in particular for virtualized environments. it is accountable for coping with the allocation and utilization of hardware sources, including processors and reminiscence, amongst more than one VMs [24]. It takes into consideration the specific hardware architecture of the gadget and the offloading capabilities of the underlying hardware. The cloud provider company is a company that provides cloud computing services, consisting of garage, computing electricity, and programs, to clients over the net [25]. The cloud service issuer is responsible for maintaining and coping with the bodily machines on which the VMs are going for walks. It additionally provides the important infrastructure and software program for the VMs to feature [26]. The interaction between those factors is vital in making sure the smooth operation of the device. The VM manages the hardware sources, the challenge manager manages the responsibilities and approaches at the VM, and the HWACOA scheduler ensures green allocation of hardware resources among a couple of VMs [27]. The cloud service provider, alternatively, presents and keeps the underlying physical infrastructure for the VMs to feature [28].

B. *Functional Block diagram*

Ant Colony Optimization (ACO) is a metaheuristic set of rules stimulated by using the foraging conduct of ant colonies. it's been widely applied to remedy complex optimization troubles, such as the service composition trouble in multi-cloud environments [29]. ACO consists of a population of synthetic ants which are programmed to look for the most excellent answer. every ant represents a potential solution and actions through a graph of service additives, comparing and updating the route based on a probabilistic selection-making mechanism [30]. in the context of service composition in multi-cloud environments, ACO is used to find the excellent aggregate of services from multiple cloud vendors to meet a given mission [31]. The problem is modeled as a graph, with offerings as nodes and their relationships as edges, in which every ant represents a potential composition solution. The functional working model have shown in the following fig.2

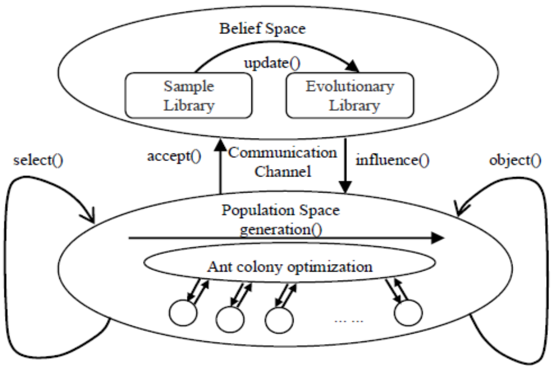

Fig.2: Functional working model

This method involves the usage of the natural foraging conduct of ants as a version for solving optimization issues. step one on this procedure is to create a sample library, that is a collection of ability answers to the optimization hassle [32]. This library is created through randomly producing a large number of capability solutions within the given populace space. those solutions may be seen as meals assets that the ants will forage for all through their look for the top-quality answer. next, the assessment library is generated via the use of a fitness feature to assess the effectiveness of every capability answer within the sample library [33]. The fitness characteristic determines how well each solution performs in achieving the preferred final results. The evaluated answers are then saved inside the evaluation library. The populace space technology is a vital step in the system as it involves growing a map of the surroundings and defining the parameters that impact the ants' foraging behavior. those parameters encompass the pheromone trail, which is the chemical trail left by the ants, and the visibility of each meals source. The ant colony optimization algorithm then starts off evolved, with the ants starting their look for the greatest solution from the sample library. They explore distinct paths and go away pheromone trails based totally on the fine of the answers they locate [34]. This pheromone trail serves as a communication mechanism among the ants, guiding them closer to the maximum promising answers located inside the evaluation library. as they seek progresses, the pheromone path intensifies across the solutions which are again and again visited by using the ants, which leads to a convergence in the direction of the top of the line answer. This system keeps till the ants locate the best solution inside the given population space.

The set of rules consists of two key mechanisms: pheromone trails and neighborhood heuristic facts. Pheromone trails act as a form of conversation among ants, wherein an ant deposits pheromone on every carrier it visits based totally on the fine of the service. This pheromone courses other ants towards probably better solutions. As ants move through the graph, they use nearby heuristic statistics to make probabilistic selections on which service to pick subsequent.

C. *Operating Principle*

Ant Colony Optimization (ACO) is a meta-heuristic set of rules stimulated by the foraging behavior of ants to resolve complicated optimization problems. it has been effectively implemented to an expansion of issues, including provider composition in multi-cloud environments. The ACO set of rules is based totally at the precept of neighborhood seek, wherein man or woman ants move through the problem space, looking for properly fine solutions. within the context of carrier composition in multi-cloud environments, ACO works with the aid of thinking about each cloud provider as a capacity solution or course towards the desired average provider composition. The Operating principle have shown in the following fig.3

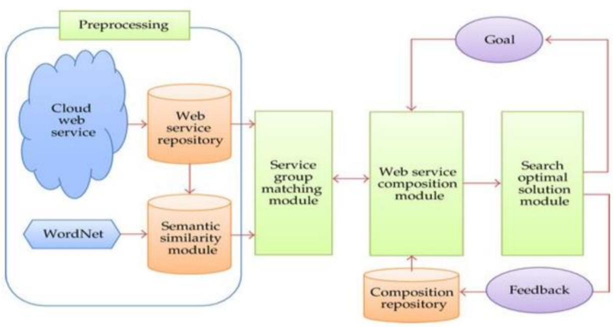

Fig.3: Operating principle

the primary module is the cloud net service, which refers back to the various packages and offerings that are hosted on a cloud platform. these services are accessed thru the net and might include things like internet website hosting, garage, and on-line packages. the subsequent module is the carrier organization matching module, that is responsible for organizing the diverse cloud internet services into one of a kind businesses based totally on their capability and compatibility. This module facilitates to streamline the search manner for finding the optimum aggregate of services for a particular undertaking. The composition module is in which the actual optimization takes place. It takes into account the user's needs and the provider organization matching outcomes to generate a composition of cloud web offerings which could great fulfill the user's requirements. This module additionally guarantees that all the individual offerings paintings together seamlessly to provide a continuing consumer revel in. The very last module is the hunt optimum module, which acts as a sort of high-quality manage for the composition module. It evaluates the composition and makes certain that it's far the most green and cost-effective answer for the consumer. This module takes into account elements consisting of fee, velocity, and reliability to determine the best composition for the consumer's needs.

The ants then deposit pheromone trails at the visited offerings, representing the elegance of a selected provider. while different ants come across these trails, they may be more likely to follow the identical route, hence reinforcing a popular answer. This procedure is analogous to ants following the heady scent of meals to their colony. The nearby search nature of ACO guarantees that it unearths precise quality answers quick, without getting caught in neighborhood optima. moreover, the adaptive nature of ACO allows it to dynamically adapt to modifications within the environment, such as service failures or addition of new offerings. as the algorithm progresses, the pheromone trails are updated, leading to a continuous development inside the high-quality of solutions.

IV. Results and Discussion

The operations of proposed model have compared with the existing ACSCE (Ant Colony System for Cloud Service Composition and Execution), ACOSCO (Ant Colony Optimization for Service Composition in Cloud Computing) and MC-ACO (Multi-cloud Ant Colony Optimization)

A. Computation of Accuracy

Ant Colony Optimization (ACO) is a metaheuristic set of rules inspired with the aid of the foraging conduct of ants. It's been widely utilized in diverse fields together with optimization, routing, scheduling, and more these days in carrier composition in multi-cloud environments. The purpose of the use of ACO for carrier composition is to correctly pick out the maximum appropriate services from a pool of to be had offerings in more than one clouds to form a composite service that meets the person's necessities. The overall performance evaluation of ACO for provider composition in multi-cloud environments has been the concern of several studies. This is especially useful in multi-cloud environments where the availability and overall performance of services may additionally range across distinctive clouds. Table.2 shows the comparison of accuracy.

Table.2: Computation of Accuracy (in %)

Inputs	ACSCE	ACOSCO	MC-ACO	Proposed
100	56.26	69.51	61.98	66.38
200	54.63	67.77	60.40	64.96
300	54.15	65.43	58.20	63.70
400	52.86	64.62	56.57	61.71
500	50.75	62.33	55.43	69.24

In a computation cycle, the existing ACSCE obtained 50.75 %, ACOSCO obtained 62.33 %, MC-ACO obtained 55.43 % accuracy. The proposed obtained 69.24 % accuracy. one of the key blessings of the use of ACO for provider composition is its ability to address huge seek spaces and discover right solutions fast. ACO works through simulating the conduct of ants in locating the shortest direction among their nest and a meals supply. This parallel and dispensed approach of looking lets in ACO to effectively explore the quest space and find most advantageous solutions. furthermore, ACO has proven to perform well in dynamic environments in which the supply and performance of services might also trade over the years. It uses nearby statistics to make selections, which makes it adaptable to modifications in the surroundings.

B. Computation of Sensitivity

Ant Colony Optimization (ACO) is a popular metaheuristic algorithm inspired via the behavior of ants. It has been extensively used for fixing optimization issues together with the service composition problem in multi-cloud environments. Provider composition is the technique of selecting and composing offerings from one-of-a-kind cloud providers to meet the wishes of a consumer. ACO is specifically appropriate for this hassle as it is able to take care of the dynamic and unpredictable nature of cloud environments. There are several approaches in which ACO can be optimized for better overall performance in provider composition for multi-cloud environments. One technique is to use adaptive ACO algorithms, which alter the parameters of the set of rules primarily based at the characteristics of the surroundings. Table.4 shows the comparison of Sensitivity.

Table.4: Computation of Sensitivity (in %)

Inputs	ACSCE	ACOSCO	MC-ACO	Proposed
100	59.26	71.51	64.98	68.38
200	57.63	69.77	63.40	66.96
300	57.15	67.43	61.20	65.70
400	55.86	66.62	59.57	63.71
500	53.75	60.33	58.43	67.24

In a computation cycle, the existing ACSCE obtained 53.75%, ACOSCO obtained 60.33%, MC-ACO obtained 58.43% precision. The proposed obtained 67.24 % Sensitivity .

This allows the set of rules to conform to changes within the cloud surroundings, which includes fluctuations in service availability or overall performance. another approach is to use parallelization strategies, consisting of dispensing the ACO set of rules across multiple processors or cloud times. this may lessen the overall execution time and improve the scalability of the algorithm. furthermore, incorporating local search heuristics, inclusive of hill-hiking, can assist enhance the exploration and exploitation skills of the algorithm. This lets in ACO to locate promising solutions more efficaciously.

C. Computation of Precision

Ant Colony Optimization (ACO) is a nature-inspired metaheuristic algorithm that has been broadly used to resolve optimization troubles in various fields. In recent years, ACO has received attention inside the field of carrier composition in multi-cloud environments because of its efficiency and potential to locate near-top-rated solutions in complicated and dynamic environments. The manner of service composition in multi-cloud environments includes selecting suitable offerings from more than one clouds to meet a given assignment or request. This poses several demanding situations, consisting of locating the maximum appropriate service combination, making sure quality of provider, and handling the dynamic adjustments inside the cloud surroundings. These demanding situations make the use of conventional optimization techniques difficult. Table.3 shows the comparison of precision.

Table.3: Computation of Precision (in %)

Inputs	ACSCE	ACOSCO	MC-ACO	Proposed
100	61.26	74.51	66.98	71.38
200	59.63	72.77	65.40	69.96
300	59.15	70.43	63.20	68.70
400	57.86	66.62	61.57	66.71
500	55.75	67.33	60.43	74.24

In a computation cycle, the existing ACSCE obtained 55.75 %, ACOSCO obtained 67.33 %, MC-ACO obtained 60.43 % precision. The proposed obtained 74.24% precision. ACO, however, has proven promising effects in tackling those challenges and demonstrating its effectiveness in carrier composition in multi-cloud environments. The comparative evaluation of ACO for carrier composition in multi-cloud environments involves comparing its overall performance in comparison to different optimization techniques. several studies had been carried out to compare ACO with other famous algorithms inclusive of Simulated Annealing, Genetic set of rules, Particle Swarm Optimization, and Tabu seek. those studies have proven that ACO outperforms other algorithms in phrases of performance, convergence, and answer best.

D. Computation of Recall

Ant Colony Optimization (ACO) is a metaheuristic algorithm inspired by using the foraging conduct of ants. It's been extensively used in fixing numerous optimization issues which include carrier composition in multi-cloud environments. Provider composition involves choosing and mixing multiple cloud offerings to meet a consumer's request. But, in a multi-cloud environment, the hassle turns into more complex because of the heterogeneity of cloud offerings and the varying network situations among distinct clouds. ACO has shown promising effects in fixing service composition issues in multi-cloud environments as it may correctly cope with the dynamic and heterogeneous nature of these environments. Its performance can further be more desirable by using incorporating a few extra techniques.Table.4 shows the comparison of Recall.

Table.4: Computation of Recall (in %)

Inputs	ACSCE	ACOSCO	MC-ACO	Proposed
100	64.26	76.51	69.98	73.38
200	62.63	74.77	68.40	71.96
300	62.15	72.43	66.20	70.70
400	60.86	71.62	64.57	68.71
500	58.75	69.33	63.43	76.24

In a computation cycle, the existing ACSCE obtained 58.75%, ACOSCO obtained 69.33%, MC-ACO obtained 63.43% recall. The proposed obtained 76.24 % recall. One such approach is the usage of a pheromone updating mechanism. Pheromones are the conversation indicators used by ants to find the shortest paths to food assets. In addition, in ACO, pheromones assist in guiding the search towards better carrier compositions by way of representing the first-class of the answers observed. The updating mechanism includes increasing the pheromone cost of the paths that lead to better answers and reducing the pheromone fee of the paths that cause terrible solutions. This way, the algorithm can speedy converge closer to a globally ideal solution, enhancing its overall performance.

V. Conclusion

Ant Colony Optimization (ACO) is a meta-heuristic algorithm stimulated by the foraging conduct of ants. It has gained reputation in recent years as an effective approach for fixing optimization troubles, especially in the field of provider composition in multi-cloud environments. provider composition involves the aggregation of character offerings to shape a more complicated and entire device to cater to the users' desires. In multi-cloud environments, there are multiple carrier vendors, every presenting numerous services with distinct first-class and cost metrics. ACO's herbal behavior of foraging for meals in an most beneficial manner makes it an efficient technique for locating the nice combination of services from extraordinary carriers, thinking about the users' requirements and preferences. The ACO set of rules works via simulating the conduct of real ants searching out meals in a colony. every ant plays a random walk, depositing pheromone trails on the direction it takes. those pheromones function oblique communication among the ants, guiding them to the meals assets. further, in ACO for service composition, the pheromone trails represent the quality and fee metrics of offerings.

References

[1] Bei, L., Wenlin, L., Xin, S., & Xibin, X. (2024). An improved ACO based service composition algorithm in multi-cloud networks. *Journal of Cloud Computing*, *13*(1), 17.

[2] Jayaudhaya, J., Jayaraj, R., & Ramash Kumar, K. (2024). A New Integrated Approach for Cloud Service Composition and Sharing Using a Hybrid Algorithm. *Mathematical Problems in Engineering*, *2024*.

[3] S. Modi, Y. K. Mali, V. Borate, A. Khadke, S. Mane and G. Patil, "Skin Impedance Technique to Detect Hand-Glove Rupture," 2023 OITS International Conference on Information Technology (OCIT),

Raipur, India, 2023, pp. 309-313, doi: 10.1109/OCIT59427.2023.10430992.

[4] Y. Mali, B. Vyas, V. K. Borate, P. Sutar, M. Jagtap and J. Palkar, "Role of Block-Chain in Health-Care Application," 2023 IEEE International Conference on Blockchain and Distributed Systems Security (ICBDS), New Raipur, India, 2023, pp. 1-6, doi: 10.1109/ICBDS58040.2023.10346537.

[5] Y. Mali, M. E. Pawar, A. More, S. Shinde, V. Borate and R. Shirbhate, "Improved Pin Entry Method to Prevent Shoulder Surfing Attacks," 2023 14th International Conference on Computing Communication and Networking Technologies (ICCCNT), Delhi, India, 2023, pp. 1-6, doi: 10.1109/ICCCNT56998.2023.10306875

[6] A. Chaudhari et al., "Cyber Security Challenges in Social Meta-verse and Mitigation Techniques," 2024 MIT Art, Design and Technology School of Computing International Conference (MITADTSoCiCon), Pune, India, 2024, pp. 1-7, doi: 10.1109/MITADTSoCiCon60330.2024.10575295.

[7] Y. K. Mali and A. Mohanpurkar, "Advanced pin entry method by resisting shoulder surfing attacks," 2015 International Conference on Information Processing (ICIP), Pune, India, 2015, pp. 37-42, doi: 10.1109/INFOP.2015.7489347 ‘.

[8] Y. K. Mali, S. A. Darekar, S. Sopal, M. Kale, V. Kshatriya and A. Palaskar, "Fault Detection of Underwater Cables by Using Robotic Operating System," 2023 IEEE International Carnahan Conference on Security Technology (ICCST), Pune, India, 2023, pp. 1-6, doi: 10.1109/ICCST59048.2023.10474270.

[9] A. Chaudhari, S. Dargad, Y. K. Mali, P. S. Dhend, V. A. Hande and S. S. Bhilare, "A Technique for Maintaining Attribute-based Privacy Implementing Blockchain and Machine Learning," 2023 IEEE International Carnahan Conference on Security Technology (ICCST), Pune, India, 2023, pp. 1-4, doi: 10.1109/ICCST59048.2023.10530511.

[10] Y. K. Mali, S. Dargad, A. Dixit, N. Tiwari, S. Narkhede and A. Chaudhari, "The Utilization of Block-chain Innovation to Confirm KYC Records," 2023 IEEE International Carnahan Conference on Security Technology (ICCST), Pune, India, 2023, pp. 1-5, doi: 10.1109/ICCST59048.2023.10530513.

[11] V. Borate, Y. Mali, V. Suryawanshi, S. Singh, V. Dhoke and A. Kulkarni, "IoT Based Self Alert Generating Coal Miner Safety Helmets," 2023 International Conference on Computational Intelligence, Networks and Security (ICCINS), Mylavaram, India, 2023, pp. 01-04, doi: 10.1109/ICCINS58907.2023.10450044.

[12] M. Dangore, A. S. R, A. GhanashyamChendke, R. Shirbhate, Y. K. Mali and V. Kisan Borate, "Multi-class Investigation of Acute Lymphoblastic Leukemia using Optimized Deep Convolutional Neural Network on Blood Smear Images," 2024 MIT Art, Design and Technology School of Computing International Conference (MITADTSoCiCon), Pune, India, 2024, pp. 1-6, doi: 10.1109/MITADTSoCiCon60330.2024.10575245.

[13] N. P. Mankar, P. E. Sakunde, S. Zurange, A. Date, V. Borate and Y. K. Mali, "Comparative Evaluation of Machine Learning Models for Malicious URL Detection," 2024 MIT Art, Design and Technology School of Computing International Conference (MITADTSoCiCon), Pune, India, 2024, pp. 1-7, doi: 10.1109/MITADTSoCiCon60330.2024.10575452

[14] Yadav, S. P., Zaidi, S., Nascimento, C. D. S., de Albuquerque, V. H. C., & Chauhan, S. S. (2023, January). Analysis and Design of automatically generating for GPS Based Moving Object Tracking System. In 2023 International Conference on Artificial Intelligence and Smart Communication (AISC) (pp. 1-5). IEEE.

[15] Dodda, S., Narne, S., Chintala, S., Kanungo, S., Adedoja, T., & Sharma, D. (2024). Exploring AI-driven Innovations in Image Communication Systems for Enhanced Medical Imaging Applications. Journal of Electrical Systems, vol20(3),pp. 949-959 Mar 24.doi.org/10.52783/jes.1409

[16] S. N. S. D. T. A. Madan Mohan Tito Ayyalasomayajula, SathishkumarChintala, “AI-Driven Decision Support Systems in Management: Enhancing Strategic Planning and Execution”, IJRITCC, vol. 12, no. 1, pp. 268–276, Mar. 2024.

[17] SanjuPurohit, Demographic Transition Model and Population Growth of India -Implications and Assessments”, vol 7(4) 176-184, 2023, doi: 10.26502/jesph.96120198.

[18] Sanju Purohit, SMART SOLUTIONS FOR ENVIRONMENTAL SUSTAINABILITY AND CLIMATE CHANGES”,vol 10(4), doi: 10.46587/JGR.2024.v10i01.016.

[19] X. Zheng et al., "Coupling Remote Sensing Insights With Vegetation Dynamics and to Analyze NO2Concentrations: A Google Earth Engine-Driven Investigation," in IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing, vol. 17, pp. 9858-9875, 2024, doi:10.1109/JSTARS.2024.3397496.

[20] Sanju Purohit, “Role of Industrialization and Urbanization in Regional Sustainable Development –Reflections from Tier-II Cities in India”,vol 12(10), pp. 13484-13493 ,2023, doi:10.48047/ecb/2023.12.10. 9442023.02/09/2023.

[21] SunitaSatapathy ,SanjuPurohit, “POND DEGRADATION AND WILDLIFE PRESERVATION: A GEOGRAPHICAL ANALYSIS”,vol 6(2),pp.74-85,2024, doi: 10.33472/AFJBS.6.2.2024.74-85.

[22] Mali, Y., &Chapte, V. (2014). Grid based authentication system, International Journal of Advance Research in Computer Science and Management Studies, Volume 2, Issue 10, October 2014 pg. 93-99, 2(10).

[23] Yogesh Mali, NilaySawant, "Smart Helmet for Coal Mining”,International Journal of Advanced Research in Science, Communication and Technology (IJARSCT)Volume 3, Issue 1, February 2023,DOI: 10.48175/IJARSCT-8064

[24] Trushank Mhatre , Yogesh Mali , Sairaj Chaudhari , Mohit Ganorkar, Pravin Dahalke, 2020, Design of Shoes Against Landmines, INTERNATIONAL JOURNAL OF ENGINEERING RESEARCH & TECHNOLOGY (IJERT) Volume 09, Issue 09 (September 2020).

[25] PranavLonari, SudarshanJagdale, ShraddhaKhandre, PiyushTakale, Prof Yogesh Mali, "Crime Awareness and Registration System ", International Journal of Scientific Research in Computer Science, Engineering and Information Technology (IJSRCSEIT), ISSN : 2456-3307, Volume 8, Issue 3, pp.287-298, May-June-2021.

[26] Jyoti Pathak, Neha Sakore, Rakesh Kapare , Amey Kulkarni, Prof. Yogesh Mali, "Mobile Rescue Robot", International Journal of Scientific Research in Computer Science, Engineering and Information Technology (IJSRCSEIT), ISSN : 2456-3307, Volume 4, Issue 8, pp.10-12, September-October-2019.

[27] Devansh Dhote , PiyushRai , Sunil Deshmukh, AdarshJaiswal, Prof. Yogesh Mali, "A Survey : Analysis and Estimation of Share Market Scenario ", International Journal of Scientific Research in Computer Science, Engineering and Information Technology (IJSRCSEIT), ISSN : 2456-3307, Volume 4, Issue 8, pp.77-80, September-October-2019.

[28] Rajat Asreddy, AvinashShingade, NirajVyavhare, ArjunRokde, Yogesh Mali, "A Survey on Secured Data Transmission Using RSA Algorithm and Steganography", International Journal of Scientific Research in Computer Science, Engineering and Information Technology (IJSRCSEIT), ISSN : 2456-3307, Volume 4, Issue 8, pp.159-162, September-October-2019

[29] Shivani Chougule, Shubham Bhosale, Vrushali Borle, Vaishnavi Chaugule, Prof. Yogesh Mali, “Emotion Recognition Based Personal Entertainment Robot Using ML & IP", International Journal of Scientific Research in Science and Technology(IJSRST), Print ISSN : 2395-6011, Online ISSN : 2395-602X, Volume 5, Issue 8, pp.73-75, November-December-2020.

[30] Amit Lokre, Sangram Thorat, Pranali Patil, Chetan Gadekar, Yogesh Mali, " Fake Image and Document Detection using Machine Learning", International Journal of Scientific Research in Science and Technology(IJSRST), Print ISSN : 2395-6011, Online ISSN : 2395-602X, Volume 5, Issue 8, pp.104-109, November-December-2020.

[31] Ritesh Hajare, Rohit Hodage, Om Wangwad, Yogesh Mali, FarazBagwan, "Data Security in Cloud", International Journal of Scientific Research in Computer Science, Engineering and Information Technology (IJSRCSEIT), ISSN : 2456-3307, Volume 8, Issue 3, pp.240-245, May-June-2021

[32] Yogesh Mali and Tejal Upadhyay, “Fraud Detection in Online Content Mining Relies on the Random Forest Algorithm”, SWB, vol. 1, no. 3, pp. 13–20, Jul. 2023, doi: 10.61925/SWB.2023.1302.

[33] Rathod, V., Mali, Y., Sable, N., Ajalkar, D., Bharathi, M., &Padmaja, N. (2023). A Network-Centred Optimization Technique for Operative Target Selection. Journal of Electrical Systems, 19(2).

[34] Deshmukh, J. Y., Rathod, V. U., Mali, Y. K., & Sable, R. (2024). and Classification. Data-Centric Artificial Intelligence for Multidisciplinary Applications, 114.