Indonesian Journal of Electrical Engineering and Computer Science
Vol. 36, No. 1, October 2024, pp. 685~693
ISSN: 2502-4752, DOI: 10.11591/ijeecs.v36.i1.pp685-693

Comparative analysis of selected optimization algorithms for mobile agents' migration pattern

Mayowa O. Oyediran[1], Sunday Adeola Ajagbe[2,3], Olufemi S. Ojo[1], Adedayo Wasiat Elegbede[4], Michael Olumuyiwa Adio[5], Abidemi Emmanuel Adeniyi[6], Isaiah O. Adebayo[2], Princewill Chima Obuzor[7], Matthew Olusegun Adigun[2]

[1]Department of Computer Science, Ajayi Crowther University, Oyo, Nigeria
[2]Department of Computer Science, University of Zululand, Kwadlangezwa, South Africa
[3]Department of Computer Engineering, First Technical University, Ibadan, Nigeria
[4]Department of Computer Engineering, Bells University of Technology, Ota Ogun State, Nigeria
[5]Department of Computer Engineering, Ajayi Crowther University, Oyo, Nigeria
[6]College of Computing and Communication Studies, Bowen University, Iwo, Nigeria
[7]School of Science, Engineering and Environment, University of Salford, Salford, United Kingdom

Article Info

Article history:

Received Feb 22, 2024
Revised Jun 8, 2024
Accepted Jun 25, 2024

Keywords:

Firefly
Honeybee
Migration pattern
Migration time
Mobile agents
Optimization algorithm
Particle swarm optimization

ABSTRACT

Mobile agents are agents that can migrate from host-to-host to work in a heterogeneous network environment. A mobile agent can migrate from host-to-host in its plan with the statistics generated on each host through a route known as migration pattern. Migration pattern therefore is the route the agents use to travel within the plan from the first host to the last host. However, there is a need for a comparison between the commonly used optimization algorithms in developing migration patterns for mobile agents with respect to some evaluation metrics. In this paper, the three techniques firefly algorithm (FFA), honeybee optimization (HBO) and particle swarm optimization (PSO) were used for developing migration patterns for mobile agents and their comparison was done based on migration time, time complexity and network load as metrics. PSO is discovered to perform better in terms of network load with an average of 242.3905 bits per second (bps), time complexity with an average of 41.2688 number of nodes (n), and migration/transmission time with an average of 4.203462 seconds (s).

Corresponding Author:

Sunday Adeola Ajagbe
Department of Computer Engineering, First Technical University
Ibadan, 200255, Nigeria
Email: saajagbe@pgschool.lautech.edu.ng

1. INTRODUCTION

Computer systems have developed from massive computing devices to highly compound client-server environments over the past few years. This new phase of evaluation gives room for total flexibility of application code amid supporting platforms to produce an insecurely combined distributed system [1]. Mobile agent technology has emerged as an attractive solution for the development of distributed computing systems [2]. Mobile agents can autonomously move from one host to another to execute their tasks, which reduces network traffic and improves system performance. However, the optimization of mobile agent migration patterns is crucial for ensuring the efficient use of network resources and minimizing system overhead [3]. Several optimization algorithms have been proposed in the literature to address this issue, including metaheuristic algorithms and machine learning-based algorithms.

Metaheuristic algorithms, such as genetic algorithm (GA), particle swarm optimization (PSO), and ant colony optimization (ACO), are commonly used to solve optimization problems [4]. These algorithms are based on natural phenomena or social behavior and can provide an optimal or near-optimal solution within a reasonable time frame. On the other hand, machine learning-based algorithms, such as artificial neural networks (ANNs) and support vector machines (SVMs), use statistical techniques to learn from data and make predictions [2]. These algorithms have shown excellent performance in various optimization tasks.

In recent years, while earlier studies have explored the impact of several studies compare different optimization algorithms for mobile agent migration pattern optimization [3]. These studies aim to determine the most effective algorithm for optimizing mobile agent migration patterns. However, there is no consensus on the best algorithm for this task, as the performance of these algorithms depends on the specific optimization problem and the algorithm parameters. Mobile agent sequence has captured the attention of researchers over a period of time because of its novel abilities and striking applications. Therefore, mobile agents are agents that can migrate from host-to-host to work in a heterogeneous network environment. It is a self-directed and practical software body that acts on behalf of a possessor and possesses the ability to migrate from end-to-end of a heterogeneous network of computers. Mobile agents are autonomous software entities that can move from one computing node to another in a network to perform tasks on behalf of their users [2]. Mobile agent migration is an important aspect of mobile agent systems, as it determines the efficiency and effectiveness of the system. Mobile agents can migrate from one node to another to take advantage of local resources or to avoid network congestion [5]. The agent of mobile agent knows its possessor, and its possessor’s partialities, and gains knowledge by interacting with its possessor. The user can allocate tasks to the agent that is able to explore the network efficiently by moving to the service or information provider. It assists roaming users since the agent can function asynchronously while the user is offline. The agent lastly presents the results of its work to the user through different channels of communication such as electronic mail, websites, pagers, or short messages via mobile phones [6].

Mobile agents can be said to be a result of the combination of software agent technology and distributed computing technology. It is so naturally distinct from the remote procedure call and the network computing node of the old [7]. That does not always require intellectual abilities, such as responsiveness, social conduct and hands-on which are also traits of current software agent know-how. These abilities are liable to be enormous in size and handling, where individual mobile agents should not use many computational resources such as processors, memory, files, and networks, at its destinations. It is however pertinent that every mobile agent be as little as possible as the cost of migrating it over a network is dependent on its size [8]. Migration in the mobile agent is the means of conveying the mobile agent code, data and state from host-to-host. Mobile agents have a mind of its own and can at any time migrate to any place that it wants, unlike the general process migration which doesn’t allow the process to choose migration time and migrating target itself. Migration pattern therefore is the route taken by the agent to migrate from one host to another [9].

The time it takes for migration is known as transmission time which is subjective to network bandwidth as well as network latency. Migration of mobile agents can be used in place of communications amongst a server to client-side program. This helps it to build up distributed systems, such as an ever-present computing environment, without being conscious of communications and protocols [2]. Optimization tends to provide the best output with the least investments [8]. The aim of optimization techniques for test cases is to reduce the number of test cases without altering the fault coverage of the testing process. Optimization techniques are used to improve the performance of mobile agent systems by reducing the migration time, minimizing the network traffic, and improving the load balancing of the system. The comparative analysis on optimization techniques for mobile agent migration patterns aims to evaluate different optimization techniques used in mobile agent systems and compare their performance in terms of migration time, network traffic, and load balancing [9]. There is a need for a comparison between the commonly used optimization algorithms in developing migration patterns for mobile agents with respect to some evaluation metrics. This study can help researchers and practitioners in the field of mobile agent systems select the most appropriate optimization technique for their specific needs.

In this study, three selected optimization algorithms commonly used for developing migration pattern was evaluated in a MATLAB environment. The three optimization algorithms are; PSO, firefly algorithm (FFA) and honeybee optimization (HBO) algorithm, using transmission time, time complexity, and network load as metrics, the selection of the algorithms was based on their effectiveness and efficiency in migration pattern investigations. The purpose of evaluating the algorithms is to determine the shortest migration route from the home host to the destination host in distributed networks while the optimization goal is to make minimal network load and minimal migration time not only to the next location of the agent’s itinerary but also for the whole plan.

Study contributions:

a) Advancements in optimization techniques: through the evaluation of optimization algorithms, the research could contribute to advancements in optimization techniques, potentially leading to the development of new algorithms or improvements to existing ones tailored specifically for mobile agent environments.

b) Insights into mobile agent systems: understanding how optimization algorithms perform within mobile agent systems can provide insights into the behavior and dynamics of such systems. This knowledge can inform the design and optimization of mobile agent-based applications and system.
c) Real-world applications: the findings of the research could be applied to real-world scenarios where mobile agent systems are utilized, such as in distributed computing, network management, and resource optimization. By improving the efficiency and effectiveness of optimization algorithms, the research can have practical implications for these applications.

2. REVIEW OF RELATED WORKS

This section discusses some of the reviewed related works: Several optimization algorithms have been proposed for optimizing the migration patterns of mobile agents. The GA, PSO, and ACO are three widely used optimization algorithms. GA is a metaheuristic algorithm that mimics the process of natural selection. PSO is another metaheuristic algorithm that mimics the social behaviour of birds. ACO is a metaheuristic algorithm that mimics the behavior of ants in finding the shortest path to a food source [10] developed a framework for the creation and operation of reusable mobile agents which is able to independently travel among nodes on several sub-networks to execute at all nodes they visit their management tasks on distributed or networked systems. This was accomplished by using a methodology with two ideas, the first was to create from two encrusted components a mobile agent, where the upper layered components were carried by the lower layered components in-between hosts following their own itineraries optimized for their target sub-networks and the upper layer components states a set of management tasks to be executed at every of the nodes to be visited.

The second idea was to provide a mechanism that matches in-between the two-layer components [4] worked on FFA with ANN for time series problems. Discovering interesting patterns concealed in the data was the primary objective. The multi-layered perceptron ANN was used just for the purpose of solving time series classification problems. The weights in the ANN are modified to provide the output values of the network, which are a lot closer to the values of the intended output. FFA was experimented with ANN for time series classification problems and was then revealed by the experimental results that the proposed FFA with ANN can successfully solve time series classification problems [11] developed using transmission time as an evaluation metric and programmed with Java a mathematical model for two mobile agent data migration pattern process using, a new migration pattern is developed and compared with the existing pattern for situations when data is migrated and not migrated. The transmission time it takes for the mobile agent to roam from the initialized host to the first host on the plan is reliant on the time taken to load all the needed code units on that host and the transmission delay in-between the two hosts. The transmission time result shows that it performs better when an agent migrates with data for the developed data migration pattern. The model which was developed was likened to the existing pattern for circumstances when data is migrated and when data is not involved [6] developed using the path strategy of ACO technique for the pull-all data strategy in distributed networks an optimized mobile agent migration pattern. The results obtained showed that the optimized pull-all data migration pattern developed using the path strategy of the ACO technique produced a lower network load and transmission time compared to the existing pull-all data migration pattern [12]. Studied migration schemes for mobile agents, determined core factors of a migration strategy, and proposed shark machine learning algorithm (SMLA) and direct method of loss allocation algorithms (DMLA). It presents the Mpid-DMLA algorithm by aiming at allowing for multiple agents' cooperation, putting ahead pid-DMLA algorithm, and revealing the characteristics of target tracking.

Several researches have been done on the FFA and other optimization algorithms. FFA was originated by Xin-She [13]. It is a meta-heuristic algorithm based on fireflies' propensity for luminosity. FFA has been used successfully in networking applications. A distance vector routing algorithm has been optimized using FFA to perform the localization of sensor nodes in the article [11]. The location estimation of sensor nodes has been done more accurately with FA than with the conventional algorithm. A survey of applications of FFA and its variants tested on continuous optimization has been presented in the article [12]. PSO is a population-based algorithm based on the cooperation of each particle; PSO is another metaheuristic algorithm mimicking how a swarm moves by following each other [14]. The convergence ability of PSO is faster than that of other optimization techniques [9]. It requires fewer parameters for the calculation of optimizing value. A number of particles may be decreased to increase the performance [15] proposed a novel optimization algorithm for mobile agent migration patterns based on an improved PSO algorithm. The proposed algorithm improved the convergence rate and solution quality of the PSO algorithm by modifying the velocity update formula and introducing a new mutation operator [16] proposed a hybrid optimization algorithm based on PSO and ACO for mobile agent migration patterns. The proposed algorithm combined the strengths of both algorithms to improve the convergence rate and solution quality of the optimization process [17] and proposed a modified GA for mobile agent migration pattern optimization. The proposed algorithm introduced a new crossover operator and a new selection operator to improve the performance of the GA [18] and proposed a novel mobile agent migration algorithm based on multi-objective optimization. The proposed algorithm optimized the migration patterns of mobile agents by considering multiple objectives, including the makespan, migration cost, and network load balance [19] proposed an optimization algorithm based on ANN for mobile agent migration patterns. The proposed algorithm used a feedforward neural network to predict the optimal migration path for mobile agents. The various optimization algorithms listed below also can be used for developing mobile agent migration patterns which are discussed as follows;

a) HBO algorithm: nature has endowed honey bees with complex behaviors such as foraging, breeding, and mating. These behaviors have served as a basis to develop several honey bee-inspired optimization algorithms. One such algorithm, inspired by the famous mating and breeding behavior of honey bees, is the marriage in honey-bees optimization (MBO). The MBO algorithm commences with a single queen without a family and progresses toward the development of a colony with one or more queens. In the literature, several versions of MBO have been proposed, including honey-bees mating optimization (HBMO) [2], fast marriage in honey-bees optimization (FMHBO) [19] and the HBO [20].
b) PSO algorithm: the swarming traits detected in schools of fish, flocks of birds, swarms of bees, and also in human social behavior are incorporated into the PSO algorithm. PSO algorithm can be seen as a population-based optimization tool, which is useful in solving diverse function optimization and complex problems.
c) Firefly optimization algorithm: in the FFA, the behavior used is that of the fireflies which are inclined to the source of the light. Two variables are vital, the intensity of light and its attractiveness. Firefly is drawn towards another fly that has a brighter flash than itself (communicates through the light emitted via them). The light intensity attractiveness is inversely proportional to the distance from the light source.
d) ACO draws its inspiration from the pheromone-based foraging strategy of ants in their natural habitat. The foraging behavior of ants is rooted in their ability to identify the shortest path between their nests and food sources [15]. During the foraging process, ants deposit pheromone trails on the path while returning to their nest from the source. Other members of the colony use these pheromone trails and their levels to locate the path [21]. When the selected path is the shortest, the pheromone level gets reinforced; otherwise, it evaporates over time. This foraging behavior of ants has served as a basis to tackle one of the most challenging optimization problems known as combinatorial optimization [17].
e) Artificial bee colony algorithm: in artificial bee colony, the colony in ABC algorithm comprises the active bees who go to their source of food, go back to the hive and dance on that area, the employed bee whose food has been uninhibited becomes a scout and begins to look for a new source of food, and the viewers who watch the employed bees dance and pick food sources depending on the dances.
f) Bats algorithm: bats algorithm is also an example of a swarm-intelligence-based algorithm that is based on the echolocation behavior of microbats. Echolocation is an attractive sonar wave the microbat releases in order to locate prey, and also in separating the various kinds of obstacles or dangers on the way to the prey in absolute darkness. To prevent getting stuck along one local maximum, it balances long-range jumps automatically around the universal search space. Searching in more detail around known good solutions to find local maxima by controlling loudness and pulse emission rates of simulated bats in the multi-dimensional search.

Cockroach swarm optimization: cockroach swarm optimization draws inspiration from the social behavior of cockroaches. Cockroaches, belonging to the Insecta Blattodea family, thrive in warm, dark, and humid environments and exhibit behaviors such as chasing, swarming, dispersing, omnivory, and food searching. Through communication and interaction with their immediate surroundings, cockroaches make decisions such as selecting shelter, searching for food and friends, dispersing when sensing danger, and resorting to cannibalism during food shortages. This efficient and straightforward meta-heuristic algorithm has been successfully employed to solve global optimization problems [21]-[26].

3. METHOD

The objective of this research was realized using the evolutionary and unified modelling languages in software engineering. This was derived from the selected algorithms which will be used to develop migration patterns for mobile agents so as to determine the shortest migration route from the home host to the destination host in distributed networks. The developed algorithm contains starting instructions that decide sub-optimal routes from the home host to the destination host, analyze path distances for each connected host and calculate the likely total distance for each sub-optimal route.

The following evaluation metrics can be used for the migration pattern of mobile agents;
- Transmission time
- Complexity

– Network load
– Throughput
– Number of hops
– Network latency

In this paper, the approach used (3) meta-heuristic algorithms to develop migration patterns for mobile agents in distributed networks, which will afterwards evaluate the developed migration patterns by the selected algorithms using transmission time, time complexity and network load as metrics. The selected optimization algorithms which are: firefly, PSO, and honeybee algorithms, were used in developing migration patterns for mobile agents using the same set of parameters and the flow diagram of the design is shown in Figure 1. In 2008, the FFA was unveiled [13]. It's a metaheuristic optimization algorithm with inspiration from nature [25]. PSO and FFA algorithms have been found to be highly helpful in obtaining significant features from mobile agents by solving pattern mitigation in those agents, these and other significant characteristics informed the choice of these optimization algorithms in this study. The parameters are mentioned below;

– Network area: 50m×50m
– Number of bees: 50
– Population size: 50
– Number of flies: 50
– Number of hosts: 20
– Maximum iteration:100
– Traffic type: UDP

The performance of the proposed migration pattern for each of the algorithms was evaluated using transmission time, time complexity, and network load as evaluation metrics. It was implemented in MATLAB. The real-valued function whose value is to be either minimized or maximized over the set of possible alternatives is the objective function in a mathematical optimization problem.

– Transmission time: the duration that a mobile agent takes to travel from its home host to the initial host on the designated path is influenced by various factors, which include the length of time required to execute the loading of all essential code units on the target host, the status and information of the agent, as well as the transmission lag between the two hosts.
– Network load: this is the size of information transmitted across a network.
– Complexity: time complexity is a measure of times it takes to simulate an algorithm.

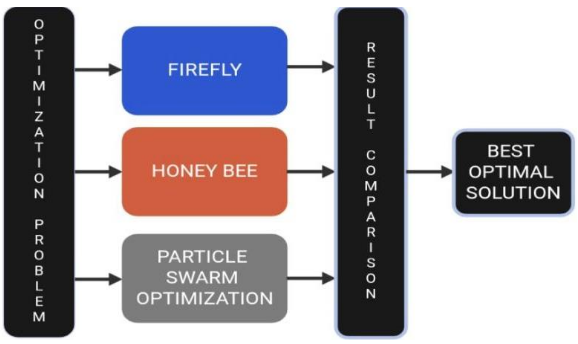

Figure 1. Flow diagram of the developed migration pattern for mobile agent

4. RESULTS AND DISCUSSION

The interface is a graphical user interface (GUI) shown in Figure 1 with three sections:

– The architecture diagram section which shows the different migration patterns and the set parameters that can be uttered,
– The result table section which shows the different results after the simulation has been carried out,
– The control panel section which shows the number of agents to be used, the selected algorithms (PSO, firefly, and honeybee) to be executed, the average migration time, network load, and time complexity per execution separately.

The results obtained with respect to transmission time (in seconds), network load measured in bits per second (bps), and time complexity (s) for the selected algorithms which were calculated using time and space ratio are presented in Tables 1-3 at 10, 30, 50, 70, and 90 iterations respectively.

Table 1. Results obtained for transmission time (seconds)

Iteration	Firefly	Honeybee	PSO
10	3.0473962	835.79216	2.9900361
30	4.551817	5.439883	3.82113182
50	9.132788	10.1061	4.2034617
70	14.22484	15.871448	4.1333098
90	19.18191	26.513358	5.459924

Table 2. Results obtained for network load (bits per second)

Iteration	Firefly	Honeybee	PSO
10	312.5787	299.9277	272.6928
30	272.1365	294.4998	248.5768
50	265.1749	277.0774	242.0905
70	260.1437	275.9076	232.0771
90	254.7464	265.6437	232.86

Table 3. Results obtained for complexity (s)

Iteration	Firefly	Honeybee	PSO
10	28.31362	44.66864	30.935135
30	41.8117	82.9685	39.22786
50	83.03399	152.97551	43.017805
70	128.76609	239.50152	50.49072
90	173.47246	399.10987	55.63134

4.1. Discussion

Table 4 shows the average of the results for each metric; network load, migration time, and time complexity, for each optimization algorithm used; PSO, honeybee optimization, and firefly optimization algorithms. The results obtained with respect to transmission time (seconds), network load measured in bits per second (bps), and time complexity (s) for the selected algorithms which was calculated using time and space ratio are for FFA, HBO, and PSO respectively.

Table 4. Summary of the obtained results

Algorithm	Transmission time (s)	Network load (bps)	Time complexity (s)
Firefly	9.132788	265.1749	83.03399
Honeybee	10.1061	277.0774	152.9755
PSO	4.203462	242.3905	41.2688

Studying optimization techniques for the migratory patterns of mobile agents is an intriguing topic, especially when considering domains such as network optimization, swarm robotics, and even urban planning. These algorithms look for the best paths and tactics to help mobile agents-like cars, robots, or drones-navigate across a given area and complete particular tasks. Finding a balance between using known information to effectively accomplish goals and exploring the surroundings to obtain information is one of the major issues in this field. In order to achieve this equilibrium, optimization algorithms are essential because they design techniques that allow mobile agents to dynamically adjust their migratory patterns in response to task requirements and environmental changes. Mobile agent migration patterns can be solved using a variety of optimization algorithms, including more modern advancements like machine learning-based methods like reinforcement learning and deep learning, as well as more traditional methods like GA, PSO, and ACO. It was observed according to the results shown in Table 4 and Figure 2 that PSO has the lowest network load, the lowest migration/transmission time, and the lowest complexity.

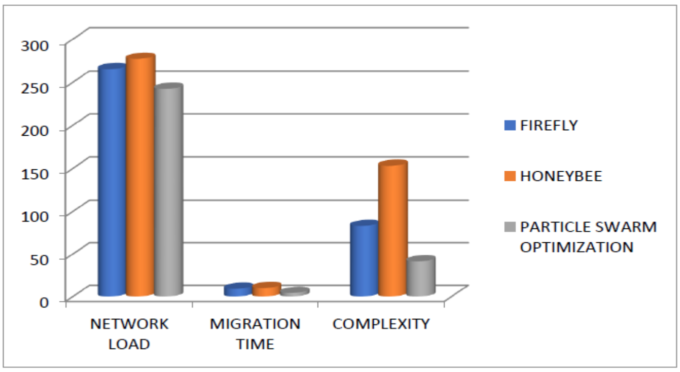

Figure 2. Summary of the obtained results

5. CONCLUSION

The purpose of evaluating the algorithms is to determine the shortest migration route from the home host to the destination host in distributed networks while the optimization goal is to make minimal network load and minimal migration time not only to the next location of the agent's itinerary but also for the whole plan. The objective of this work is to evaluate the performance of some selected optimization algorithms for the migration pattern of mobile agents using three evaluation metrics (network load, transmission time and time complexity of the algorithms). PSO is discovered to perform better in terms of network load with an average of 242.3905 bps, complexity with an average of 41.2688 n, and migration/transmission time with an average of 4.203462 seconds. Further studies may consider. Examining how machine learning techniques might be integrated to forecast mobile agents' ideal migration patterns based on past data and network circumstances. Examine the performance of the chosen optimization methods as well as any new ones when the network's size or the number of mobile agents increases. Examine how scalable they are and look for any possible bottlenecks.

ACKNOWLEDGEMENTS

The authors acknowledge the support of the University of Zululand, South Africa for footing the publication fee.

REFERENCES

[1] M. O. Oyediran, T. M. Fagbola, S. O. Olabiyisi, E. O. Omidiora, and A. O. Fawole, "A survey on migration process of mobile agent," *Proceedings of the World Congress on Engineering and Computer Science*, 2016.

[2] S. Juniastuti, M. Fachri, S. M. S. Nugroho, and M. Hariadi, "Inert and mobile agents navigation interaction using reciprocal velocity obstacles for collisions avoidance," *Indonesian Journal of Electrical Engineering and Computer Science*, vol. 26, no. 2, pp. 1116–1124, May 2022, doi: 10.11591/ijeecs.v26.i2.pp1116-1124.

[3] S. Mirjalili, "Dragonfly algorithm: a new meta-heuristic optimization technique for solving single-objective, discrete, and multi-objective problems," *Neural Computing and Applications*, vol. 27, no. 4, pp. 1053–1073, May 2016, doi: 10.1007/s00521-015-1920-1.

[4] C. Cheng, Y. Liu, C. Liu, and M. Chen, "An optimization algorithm based on artificial neural networks for mobile agent migration pattern," *Journal of Ambient Intelligence and Humanized Computing*, vol. 9, no. 5, pp. 1569–1579, 2018.

[5] P. J. García-Nieto, E. García-Gonzalo, J. A. Vilán Vilán, and A. Segade Robleda, "A new predictive model based on the PSO-optimized support vector machine approach for predicting the milling tool wear from milling runs experimental data," *International Journal of Advanced Manufacturing Technology*, vol. 86, no. 1–4, pp. 769–780, Dec. 2016, doi: 10.1007/s00170-015-8148-1.

[6] D. Ting, H. Haiping, L. Yang, W. Ruchuan, and P. Xinxing, "Research on migration strategy of mobile agent in wireless sensor networks," *International Journal of Distributed Sensor Networks*, vol. 2013, no. 11, p. 642986, Nov. 2013, doi: 10.1155/2013/642986.

[7] M. Higashino, K. Takahashi, T. Kawamura, and K. Sugahara, "Mobile agent migration based on code caching," in *Proceedings - 26th IEEE International Conference on Advanced Information Networking and Applications Workshops, WAINA 2012*, Mar. 2012, pp. 651–656, doi: 10.1109/WAINA.2012.127.

[8] M. Alweshah, "Firefly algorithm with artificial neural network for time series problems," *Research Journal of Applied Sciences, Engineering and Technology*, vol. 7, no. 19, pp. 3978–3982, May 2014, doi: 10.19026/rjaset.7.757.

[9] S. Osunade and F. A. Atanda, "Analysis of two mobile agent migration patterns," *Journal of Mobile communication*, vol. 2, no. 2, pp. 64–72, 2008.

[10] F. T. Johora, I. Ahmed, M. A. I. Shajal, and R. Chowdhory, "A load balancing strategy for reducing data loss risk on cloud using remodified throttled algorithm," *International Journal of Electrical and Computer Engineering*, vol. 12, no. 3, pp. 3217–3225, Jun. 2022, doi: 10.11591/ijece.v12i3.pp3217-3225.

[11] S. A. Ajagbe, M. O. Oyediran, A. Nayyar, J. A. Awokola, and J. F. Al-Amri, "P-acohoneybee: a novel load balancer for cloud computing using mathematical approach," *Computers, Materials and Continua*, vol. 73, no. 1, pp. 1943–1959, 2022, doi: 10.32604/cmc.2022.028331.

[12] C. Yinka-Banjo, O. Daniel, S. Misra, O. Jonathan, and H. Florez, "Comparative analysis of three obstacle detection and avoidance algorithms for a compact differential drive robot I N V-Rep," in *Communications in Computer and Information Science*, vol. 1051 CCIS, Springer International Publishing, 2019, pp. 357–369.

[13] B. Pei, H. Zhang, T. Pei, and H. Wang, "Firefly algorithm optimization based WSN localization algorithm," in *IET Conference Publications*, 2015, vol. 2015, no. CP664, doi: 10.1049/cp.2015.0216.

[14] S. L. Tilahun and J. M. T. Ngnotchouye, "Firefly algorithm for discrete optimization problems: A survey," *KSCE Journal of Civil Engineering*, vol. 21, no. 2, pp. 535–545, Jan. 2017, doi: 10.1007/s12205-017-1501-1.

[15] A. Slowik, "Particle swarm optimization," 2011, doi: 10.1007/978-3-319-46173-1_2.

[16] D. Sanchez and H. Florez, "Model driven engineering approach to manage peripherals in mobile devices," in *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, vol. 10963 LNCS, Springer International Publishing, 2018, pp. 353–364.

[17] J. Hernandez and H. Florez, "An experimental comparison of algorithms for nodes clustering in a neural network of caenorhabditis elegans," in *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, vol. 12957 LNCS, Springer International Publishing, 2021, pp. 327–339.

[18] Y. Zhang, J. Liu, and N. Xiong, "A novel optimization algorithm for mobile agent migration pattern based on improved particle swarm optimization," *Journal of Ambient Intelligence and Humanized Computing*, vol. 12, no. 8, pp. 9199–9211, 2021.

[19] Y. Zhao, X. Li, L. Zhang, and S. Li, "A hybrid optimization algorithm based on PSO and ACO for mobile agent migration pattern," *Journal of Ambient Intelligence and Humanized Computing*, vol. 11, no. 11, pp. 4753–4763, 2020.

[20] J. Yang, H. Chen, and Y. Hu, "A modified genetic algorithm for mobile agent migration pattern optimization," *Journal of Intelligent and Fuzzy Systems*, vol. 39, no. 6, pp. 8243–8250, 2020.

[21] L. Zeng, J. Zhang, and C. Hu, "A novel mobile agent migration algorithm based on multi-objective optimization," *IEEE Access*, vol. 7, pp. 51807–51817, 2019.

[22] J. Wang, Y. Zhang, and X. Zhu, "A survey of mobile agent technology in distributed computing systems," *Journal of Distributed Sensor Networks*, vol. 15, no. 3, 2019.

[23] R. Rawat, O. A. Oki, R. K. Chakrawarti, T. S. Adekunle, J. M. Lukose, and S. A. Ajagbe, "Autonomous artificial intelligence systems for fraud detection and forensics in dark web environments," *Informatica (Slovenia)*, vol. 47, no. 9, pp. 51–62, Oct. 2023, doi: 10.31449/INF.V46I9.4538.

[24] H. Florez and A. Cárdenas-Avendaño, "A computer-based approach to study the gaussian moat problem," in *Communications in Computer and Information Science*, vol. 1277 CCIS, Springer International Publishing, 2020, pp. 481–492.

[25] F. W. Ipeayeda, M. O. Oyediran, S. A. Ajagbe, J. O. Jooda, and M. O. Adigun, "Optimized gravitational search algorithm for feature fusion in a multimodal biometric system," *Results in Engineering*, vol. 20, p. 101572, Dec. 2023, doi: 10.1016/j.rineng.2023.101572.

[26] A. O. David. and A. Osofisan, "Route optimization in MIPv6 experimental test bed for network mobility: trade off analysis and evaluation," *International Journal of Computer Science and Information Security IJCSIS*, vol. 18, no. 5, pp. 19–28, 2020.

BIOGRAPHIES OF AUTHORS

Mayowa O. Oyediran is a senior lecturer in the department of Computer Engineering in Ajayi Crowther University, Oyo town in Nigeria, he has a Ph.D. in Computer Engineering from the Ladoke Akintola University of Technology, Ogbomoso. Nigeria. He is specialized in Distributed Systems and Applications with focused on cloud computing, fog computing, MANETs and the likes. He can be contacted at email: mo.oyediran@acu.edu.ng.

Sunday Adeola Ajagbe is a Ph.D. candidate at the Department of Computer Science, University of Zululand, South Africa and a Lecturer, a First Technical University, Ibadan, Nigeria. He obtained MPhil Computer Engineering at Ladoke Akintola University of Technology (LAUTECH), Ogbomoso, M.Sc. and B.Sc. in Information Technology and Communication Technology respectively at the National Open University of Nigeria (NOUN), and his Postgraduate Diploma in Electronics and Electrical Engineering at LAUTECH. His specialization includes artificial intelligence (AI), natural language processing, information security, data science, and the internet of things (IoT). He is also licensed by The Council Regulating Engineering in Nigeria as a professional Electrical Engineer, and International Association of Engineers (IAENG). He has many publications to his credit in reputable academic databases. He can be contacted at email: saajagbe@pgschool.lautech.edu.ng.

Olufemi S. Ojo is currently a lecturer at the Department of Computer Sciences, Ajayi Crowther University Oyo town in Nigeria, He has a Ph.D. in Computer Science from Ladoke Akintola University of Technology, Ogbomoso. Nigeria. His research focuses on Pattern Recognition, Optimization Techniques and E-learning. He is a full member of the International Association of Engineers (IAEng) and Computer Professionals of Nigeria (CPN). He can be contacted at email: os.ojo@acu.edu.ng.

Adedayo Wasiat Elegbede had her Master's degree in Electronics and Computer Engineering, at the department of Electrical/Electronics and Computer Engineering, Bells University of Technology from 2017-2018. She is attached to the Department of Computer Engineering, Bells University of Technology. She can be contacted at email: awelegbede@bellsuniversity.edu.ng.

Michael Olumuyiwa Adio had his first degree in Computer Engineering (2008) and Master's degree with research (Pattern Recognition) in Computer Science at Ladoke Akintola University of Technology, Ogbomoso (LAUTECH) in 2019, Oyo state. He taught at Osun state Polytechnic, Iree as lecturer where taught various computer engineering courses between the year 2011-2023. He got another golden opportunity to further enhance his teaching skills at Ajayi Crowther University, Oyo, Oyo state Nigeria, where his currently lecturing in the department of Computer Engineering. He is currently running his Ph.D. in Computer Engineering at LAUTECH, Ogbomoso, Oyo State. He can be contacted at email: mo.adio@acu.edu.ng.

Abidemi Emmanuel Adeniyi is a researcher in the Department of Computer Science at Bowen University, Iwo, Nigeria. He is currently a Ph.D. student at the University of Ilorin, Ilorin, Nigeria. He has authored a quite number of publications in reputable journals. His area of research interest includes; machine learning, internet of things security, information security, and intrusion detection systems. He can be contacted at email: abidemi.adeniyi@bowen.edu.ng.

Isaiah O. Adebayo is a Ph.D. candidate with the department of computer science, University of Zululand, South Africa. He had a M.Sc. by coursework at the Department of Computer Systems and networking, London South Bank University, from 2012-2013. With a decade of teaching experience in higher education and as a Ph.D. scholar, my research interests' centres around optimisation, software defined networking, and machine learning. He can be contacted at email: adebum@yahoo.com.

Dr. Princewill Chima Obuzor is a Ph.D. Candidate in Data Mining at the University of Salford in Manchester United Kingdom. He obtained a Masters in Databases and Web Based systems at the University of Salford and a B.Sc. with Distinction at V. Dahl East Ukrainian National University in Luhansk, Ukraine. His research interests are in machine learning and artificial intelligence, data science, process mining and blockchain technologies. He is a member of the British computer society (BCS) in the UK. He can be contacted at email: p.obuzor@edu.salford.ac.uk.

Matthew Olusegun Adigun retired in 2020 as a Senior Professor of Computer Science at the University of Zululand. He obtained his doctorate degree in 1989 from Obafemi Awolowo University, Nigeria; having previously received both Masters in Computer Science (1984) and a Combined Honours degree in Computer Science and Economics (1979) from the same University (when it was known as University of Ife, Nigeria). He has received both research and teaching recognitions for raising the flag of Excellence in Historically Disadvantaged South African Universities as well as being awarded a 2020 SAICSIT Pioneer of the Year in the Computing Discipline. Currently, he works as a Temporary Senior Professor at the Department of Computer Science, University of Zululand, Kwadlangezwa, South Africa to pursue his recent interest in AI-enabled Pandemic Response and Preparedness. He can be contacted at email: adigunm@unizulu.ac.za.