ISSN (Online) 2394-2320

International Journal of Engineering Research in Computer Science and Engineering (IJERCSE)

Volume 11 Issue 7 July 2024

# A Comparative Analysis of Selected Optimization Algorithms for Mobile Agents Migration Pattern

[1] Micheal Tokunbo Adenibuyan, [2] Mayowa Oyedepo Oyediran, [3] Oyinkansola Anuoluwapo Olagunju

[1] [3] Department of Computer Science & Information Technology, College of Natural and Applied Sciences, Bells University of Technology, Ota, Ogun State, Nigeria
[2] Department of Computer Science, Ajayi Crowther University, Oyo, Oyo State, Nigeria
Corresponding Author Email: [1] mtadenibuyan@bellsuniversity.edu.ng, [2] mayor_yoppy02@yahoo.com, [3] oaolagunju@bellsuniversity.edu.ng

***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 itinerary with the statistics generated on each host through a route known as migration pattern. Migration pattern therefore is the route the agents use to move from the first host to the last host within the itinerary. However, there is need for a comparison between the commonly used optimization algorithms in developing migration patterns for mobile agent in respect to some evaluation metrics. In this paper, the various techniques commonly used for developing migration pattern for mobile agent are discussed and their comparison was done based on migration time, complexity and network load as metrics.*

***Index Terms**— Mobile Agents, Migration Pattern, Optimization Algorithm.*

## I. INTRODUCTION

Computer systems have developed from a massive computing device to a highly complex client-server environment over the past few years. This new phase of evaluation gives room for total mobility of application code among supporting platforms to produce a loosely coupled distributed system [5]. 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 which acts on behalf of a possessor and possesses the ability to migrate from end-to-end of a heterogeneous network of computer. The agent of a mobile agent knows its possessor, 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 results of its work to the user through different channels of communication such as electronic mails, websites, pagers, or short messages via mobile phones.

Mobile agent 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 old. That does not always require intellectual abilities, such as the responsive, social conducts and hands-on which are also traits of current software agent knowhow. These abilities are liable to being enormous in size and handling, where individual mobile agent should not use many computational resources such as processor, memory, files, and networks, at its destinations. It is however pertinent that each mobile agent be as small as possible as the cost of migrating it over a network is dependent on its size. [7].

Migration in mobile agent is the means of conveying the mobile agent code, data and state from host-to-host. Mobile agents has a mind of its own and can at anytime migrate to any place that it wants unlike the general process migration which doesn't allow process to chose migration time and migrating target itself. Migration pattern therefore is the route taken by the agent to migrate from one host to another. 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 agent 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].

The various optimization algorithms listed below also can be used for developing mobile agent migration pattern which are discussed as follows;

1. Honeybee Optimization Algorithm: According to nature, bees exhibit numerous complex traits like copulating, breeding and foraging. All these traits have been imitated for some honeybee-based optimization algorithms. In most of the developed algorithms, it begins with a single queen without any family and later progress to the establishment of a colony with more than one queen.
2. Particle Swarm Optimization Algorithm (PSO): The swarming traits detected in schools of fish, flocks of birds, swarms of bees, and also in human social behavior are incorporated into PSO algorithm. PSO

algorithm can be seen as a population-based optimization tool, which is useful in solving diverse function optimization and complex problems.

3. Firefly Optimization Algorithm: In firefly algorithm, the behavior used is that of the fireflies which is inclined to the source of the light. Two variables are vital, 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.
4. Ant Colony Optimization Algorithm (ACO): ACO is based on the pheromone-based approach of ants foraging in nature. Finding the shortest route from their source to their nests is based on the foraging behavior of ants. During this process, the ants drop their pheromone pathways on the route when they return to their nest from the source, so that the route is located by other members of the colony through the use of the pheromone pathways and level.
5. Artificial Bee Colony Algorithm: In Artificial Bee Colony, the colony in ABC algorithm comprises of 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 watches the employed bees dance and pick food sources depending on the dances.
6. Bats Algorithm: Bats Algorithm is also an example of swarm-intelligence-based algorithm which is based on the echolocation behavior of microbats. Echolocation is an attractive sonar wave the microbat releases in order to locate preys, 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 maxima, It balances long-range jumps automatically around the universal search space. Searching in more details around known good solutions to find local maxima by controlling loudness and pulse emission rates of simulated bats in the multi-dimensional search.
7. Cockroach Swarm Optimization: Cockroach Swarm Optimization is inspired by cockroach social behavior. Cockroach belongs to Insecta Blattodea family, lives in warm, dark, and humid environment, and its habits include chasing, swarming, dispersing, being ruthless, omnivorous, and food searching. Cockroaches communicate with one another and react to their immediate environment and make decisions based on their interaction such as selecting shelter, searching for food sources and friends, dispersing when danger is noticed, and consumption of one another when there is scarcity of food. It has been used successfully to solve global optimization problems because it has an efficient and simple meta-heuristic algorithm.

In this study, three selected optimization algorithms commonly used for developing migration pattern was evaluated in MATLAB environment. The three optimization algorithms are; Particle Swarm Optimization (PSO), Firefly Algorithm (FFA) and Honeybee Optimization Algorithm, using transmission time, complexity, and network load as metrics. The purpose for 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 itinerary.

## II. REVIEW OF RELATED WORKS

This section discusses some of the reviewed related works:[7] developed a frame for the creation and operation of the 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 is carried by the lower layered components 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 each of the nodes to be visited. The second idea was to provide a mechanism that match makes between the two-layer components. [4] worked on Firefly Algorithm with Artificial Neural Network for Time Series Problems. Discovering interesting patterns concealed in the data was the primary objective. The multi-layered perceptron Artificial Neural Networks (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. Firefly Algorithm was experimented with ANN for time series classification problems and was then revealed by the experimental results that the proposed Firefly Algorithm with Artificial Neural Network can successfully solve time series classification problems.[5] developed using transmission time as 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 mobile agent to roam from the initialized host to the first host on the itinerary is reliant on the execution time that is used to load all the needed code units on that host (the agent data and state) and the transmission delay 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 with the existing pattern for circumstances when data is migrated and when data is not involved. [6] developed using path strategy of Ant Colony Optimization (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 path strategy of Ant Colony Optimization (ACO) technique produced a lower network load and transmission time compared to the existing pull-all data migration pattern. [1] studied migration schemes for mobile agents, determined core factors of migration strategy, and proposed Shark Machine Learning Algorithm (SMLA) and Direct Method of Loss Allocation algorithms (DMLA). It presents Mpid-DMLA algorithm by aiming at allowing for multiple agents' cooperation, putting ahead pid-DMLA algorithm, and revealing the characteristics of target tracking.

## III. METHODOLOGY

The objective of this research was realized using the evolutionary and unified modeling languages in software engineering. This was derived from the selected algorithms which will be used to develop migration pattern for mobile agent 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 home host to 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 migration pattern of mobile agent;

i. Transmission time
ii. Complexity
iii. Network load
iv. Throughput
v. Number of hops
vi. Network Latency

In this paper, the approach used three (3) meta-heuristic algorithms to develop migration patterns for mobile agent in distributed network, which will afterwards evaluate the developed migration patterns by the selected algorithms using transmission time, complexity and network load as metrics. The selected optimization algorithms which are: Firefly, Particle Swarm Optimization (PSO), and Honeybee algorithm, were used in developing migration patterns for mobile agent using the same set of parameters. The parameters are mentioned below;

i. Network Area: 50m x 50m
ii. Number of Bees: 50
iii. Population size: 50
iv. Number of flies: 50
v. Number of host: 20
vi. Maximum Iteration:100
vii. Traffic Type: UDP

The performance of the proposed migration pattern for each of the algorithm was evaluated using transmission time, complexity and network load as evaluation metrics. It was implemented in MATLAB.

Transmission time: The transmission time for the mobile agent to migrate from the home host to the first host on the itinerary is dependent on the execution time used to load all needed code units on that host; the agent data and state; and the transmission delay between the two hosts.

Network load: This is the size of information transmitted across a network.

Complexity: Time complexity is a measure of time it takes to simulate an algorithm.

## IV. RESULTS

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

i. The architecture diagram section which shows the different migration patterns and the set parameters that can be uttered,
ii. The result table section which shows the different result after simulation has be carried out,
iii. 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 complexity per execution separately.

The results obtained in respect of transmission time, network load, and complexity for the selected algorithms are presented in table 1, table 2, and table 3 at 10, 30, 50, 70 and 90 iterations respectively.

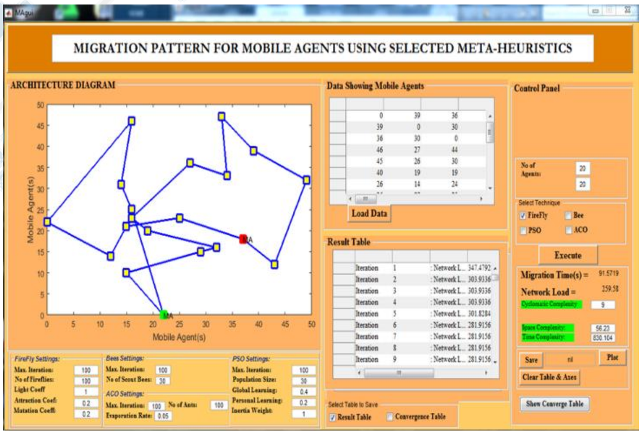


**Fig. 1.** The Graphical User Interface

**Table I:** Results obtained for Transmission time

| 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 II:** Results obtained for Network load

| 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 III:** Results obtained for Complexity

| Iteration | Firefly | Honeybee | PSO |
|---|---|---|---|
| 10 | 28.31362 | 44.66864 | 30.935135 |
| 50 | 83.03399 | 152.97551 | 43.017805 |
| 70 | 128.76609 | 239.50152 | 50.49072 |
| 90 | 173.47246 | 399.10987 | 55.63134 |

## V. DISCUSSION

Table 4 shows the average of the results for each metrics; network load, migration time, and complexity, for each optimization algorithm used; Particle Swarm Optimization, Honeybee Optimization, and Firefly Optimization algorithms.

**Table IV:** Summary of the obtained results

| ALGORITHM | NETWORK LOAD | MIGRATION TIME | COMPLEXITY |
|---|---|---|---|
| Firefly | 265.1749 | 9.132788 | 83.03399 |
| Honeybee | 277.0774 | 10.1061 | 152.9755 |
| PSO | 242.3905 | 4.203462 | 41.2688 |

It was observed according to the results shown in table 4 and figure 2 that Particle swarm optimization algorithm (PSO) has the lowest network load, the lowest migration/ transmission time, and the lowest complexity.

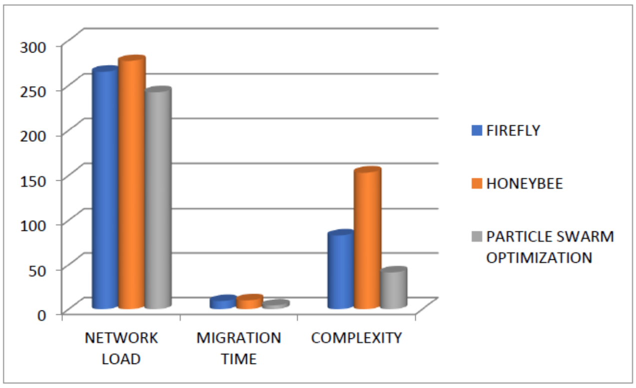


**Fig. 2.** Summary of the obtained results

## VI. CONCLUSION

The objective of this work is to evaluate the performance of some selected optimization algorithms for migration pattern of mobile agents using three evaluation metrics (network load, transmission time and time complexity of the algorithms). Particle Swarm Optimization is discovered to perform better in terms of network load with an average of 242.3905, complexity with an average of 41.2688, and migration/transmission time with an average of 4.203462 seconds.

## REFERENCES

[1] DaiTing, HuangHaiping LuYang, WangRuchuan and PanXinxing (2013): "Research on Migration Strategy of Mobile Agent in Wireless Sensor Networks" International Journal of Distributed Sensor Networks 2013(10):1-13

[2] Higashino M., Takahashi K., Kawamura T., and Sugahara K (2012): "Mobile agent migration based on code caching", 26th International Conference on Advanced Information Networking and application workshop, 651 – 656.

[3] Ichiro Satoh (2006): "Building and Selecting Mobile Agents for Network Management"Journal of Network and Systems Management 14(1):147-169

[4] Mohammed Alweshah (2014): "Firefly Algorithm with Artificial Neural Network for Time Series Problems" Research Journal of Applied Sciences, Engineering and Technology, Vol 7, Issue 19, 3978-3982

[5] Osunade S. and Atanda F.A (2008): "Analysis of two mobile agent migration patterns". Journal of mobile communication, 2(2), 65-66.

[6] Oyediran M. O, Fagbola T. M, Olabiyisi S. O, Omidiora E. O, Fawole A. O (2016): "A Survey on Migration Process of Mobile Agent", Proceedings of the World Congress on Engineering and Computer Science 2016 Vol I WCECS 2016.

[7] Satoh, I. (2002): "A Framework for Building Reusable Mobile Agents for Network Management", IEEE Communication Society, 51-64.