Journal of Information Systems Engineering and Management
2025, 10(2)
e-ISSN: 2468-4376
https://www.jisem-journal.com/

Research Article

Enhancing Coverage and Efficiency in Wireless Sensor Networks Using Bio-Inspired Improved Moth-Flame Optimization (IMFO) Technique

[1,*]Deeptha R, [2]Ms. S. Rama, [3]Dr.G.Rajalakshmi, [4]Dr.K.P.Senthilkumar, [5]Navaneethan S, [6]Dr.N.Kumaran,
[1]Assistant Professor, Department of Information Technology, SRM Institute of Science and Technology, Ramapuram, Chennai, r.deeptha@gmail.com
[2]Assistant Professor, Department of Information Technology, SRM Institute of Science and Technology, Ramapuram, Chennai, ramas1@srmist.edu.in
[3]Assistant professor, Department of IT, SRM Institute of Science and Technology, Ramapuram, Chennai, raje.ma2007@gmail.com
[4] Associate Professor, Department of ECE, Saveetha School of Engineering, Saveetha Institute of Medical and Technical Sciences, Saveetha University, Chennai, India, skkr8587@gmail.com
[5]Assistant Professor, Dept.of ECE, Saveetha Engineering College, Chennai, Tamil Nadu. India. jssnavi.37@gmail.com
[6]Associate professor, Department of Mathematics, Vel Tech Rangarajan Dr. Sagunthala R&D Institute of Science and Technology, Avadi, Chennai - 600062
nkumaran@veltech.edu.in.

ARTICLE INFO

Received: 28 Sept 2024

Revised: 27 Nov 2024

Accepted: 05 Dec 2024

ABSTRACT

WSNs have become highly implemented in several disciplines, ranging from medicine and ecology to agriculture, among many other areas. Critical for development and advancement are innovative approaches towards boosting Wireless Sensor Networks (WSNs) where positioning and density become prime objectives with a goal towards complete sensing within a much more expansive space. This research is interested in finding coverage nodes on a target-based WSN over a wider area by applying the bio-inspired Improved Moth-Flame optimization technique, IMFO. It has been assumed that insects apply transverse orientation for finding their way in the wild. In this work, the technique IMFO targets to enhance the rate of convergence in wireless sensor networks by maximizing coverage with a minimum number of nodes needed across a broader region and with low compilation time. The whole sensing area is partitioned into an exact number, "n," of individual sensing zones. The sensors were localized in each location using IMFO, and for the next iterations, the entire localized sensors of all regions were treated as a moth. A great deal of efficacy and efficiency is shown by the results of the trial. The proposed approach entails modeling the optimization problem mathematically, taking into account variables like energy consumption, communication range, sensing coverage, and total node count. Whenever there comes an optimization issue, the toolbox in MATLAB optimizes with a powerful optimization engine and comes up with the best possible solution. Several performance measures include compilation time, F-value, packet delivery ratio, and throughput, which are significantly improved according to the proposed method.

Keywords: Moth Flame Optimization, Wireless Sensor Network, Network Simulation, Cluster, Routing

1. INTRODUCTION

A wireless sensor network (WSN) is an ad-hoc network of low-cost, battery-operated modules acting as sensors with the capability of wireless communication. A huge area of application has been found for it where the data is collected in the field and sent to a central location for analysis in fields as wide as environmental monitoring, healthcare, and agriculture [1]. However, some of the weaknesses associated with WSNs are the unpredictable network topologies, inconsistent connectivity, and power output. These dangers jeopardize the effectiveness, longevity, and usefulness of the network [2]. The main objectives of continuing research on optimizing WSN are to enhance network performance and solve these challenges. The following are some of the major network aspects WSN optimization focuses on: location of sensor nodes, number of nodes, communication range, routing approaches, and power consumption [3].

The aim of optimization is to reduce power consumption, increase network availability, prolong network life, and speed data transmission rates. Because optimization may significantly enhance WSNs' usefulness and efficiency, it is an essential part of their design and implementation [4]. Research into WSN optimization has come a long way, but there are still a number of challenges that academics might use some assistance with [5]. If you wish to have maximum coverage of the detecting region while minimizing practical power consumption, you have to determine how many sensor nodes to place where [6]. The efficiency, longevity, and value of the network depend on the placement and density of the sensor nodes.

It is important to consider the sensing zone's size and form, the number of barriers, the breadth of communication, and the energy needed to sustain the network when deciding on the best place to put the sensor nodes and how many there should be. If we want WSNs to last as long as possible and be as useful as possible, we need to optimize them [7]. Both data transfer rates and power consumption are augmented through improved communication for a node of the WSN. An appropriate placing with the right assigning of sensor units could provide entire coverage to the zone with sensing functionality and nearly inappreciable impacts on overall system performance. Better real-time monitoring along with better decision support can thus come to fruition based on enhanced versions of WSNs. To ensure optimum performance in a WSN, careful consideration has to be made regarding placement and the number of sensor sites [8]. Optimizing the sensor node design in a way that provides full coverage to the target area using the minimum possible power is considered to be achieved by an optimization approach. Several steps involved in the optimization approach are: problem formulation of optimization, choice of an optimization method, and assessing the optimized result [9]. Moths are insects belonging to the phylum arthropoda, with specific metaheuristics in navigation, and recently, researchers in optimization got much interest in these insects. The transverse orientation mechanism, along with its unique nighttime flight pattern, allows effective navigation by pointing the center of attraction toward the moon that they call flame.

In a WSN, one random sensor played the role of the flame, and the others, who played the role of the moth, discussed how to achieve minimal distance with improved packet delivery in order to get cheap outcomes. Now philosophy has spread to IMFO, where a faster compilation time is traded for a better PDR through an upgrade of the Sensor's location and an update of the global candidate solution of smaller subregions of the overall region. In order to improve sensing coverage while minimizing power consumption, the proposed study uses IMFO to estimate the ideal distribution and number of sensor nodes over a WSN. Moths, attracted to lights, are what the IMFO meta-heuristic optimization approach was inspired by [10].

In this way, individuals who just hang around can be exploited. In addition to the ability of the Moth Flame Optimization (MFO) algorithm to solve optimization problems that are neither linear nor convex, one of the numerous advantages it offers is quicker convergence and more efficient global search. The optimization technique accompanying the analysis is performed in MATLAB. Among others, some popular applications of MATLAB include the simulation and optimization of mathematical models [11, 12]. A mathematical model incorporating many parameters of the network - total size, sensing area, communication distance, and energy use - is here presented. By using the toolbox of MATLAB in optimization, a best alternative could be obtained. To determine the effectiveness of the proposed optimization technique, the current study uses a hypothetical scenario. Several performance measures are considered when running simulations.

2. METHODS/EXPERIMENTAL

This research is motivated by the desire to improve the dependability of WSNs, which is currently constrained by limited resources, unstable topologies, and intermittent connectivity. While there are several optimization approaches, enhancing the performance of WSNs requires a more effective method [13]. The research suggests that essential features of WSNs, including their design, power consumption, and throughput, may be enhanced by using MFO [14]. Fundamental goal of this study is to assess MFO's value in WSN optimization and to contrast it with other optimization methods. Matlab tool is used in optimization. Design parameters and design variables are chosen varyingly to analyze the optimization process. Experimentally, the region is divided into subregions with randomly deployed sensors, and the IMFO algorithm is carried out in each subregion to localize the flames with the objective function of either maximizing the PDR or minimizing the compilation time based on the fitness value. The proposed research will utilize simulated trials to examine the usefulness of MFO in optimizing WSNs and to assess the performance of MFO-optimized multiple metrics, including compilation time, F-value, Packet delivery ratio, and throughput, which are used to evaluate WSNs. The study focuses on compilation time with optimal Sensor and converges area while the entire sensing region is divided and conquered to locate the optimized flame location. Finally, flames in the subregion are considered as moths to localize the Sensor globally. This research will add to the body of knowledge regarding WSN optimization techniques, resulting in more efficient and effective WSNs [15].

2.1 Literature Survey

Although WSNs are quite new, they have already found extensive use in sectors as diverse as environmental monitoring, agriculture, and healthcare. However, because WSNs have limited resources, performance improvement is critical. In the last few years, several academics have suggested different optimization techniques to increase WSNs' data transmission speeds, longevity, and energy efficiency. The performance of several optimization schemes for WSN is tested in the present research study. Y. Yao et al. [15] presented the new approach termed as Moth Flame Search (MFS). Here, an optimized deployment of the WSN can be accelerated as per their introduced approach. Their proposed work suggested an adaptive inertia weight and variable location of spiral towards enhancing local augmentation and global searching ability of MFS. The sum of all attractive forces of the exposed grid points, all virtual forces between neighbor sensor nodes, and all repulsive forces between sensor nodes and barriers is considered to obtain the total virtual force of the system. Results from the simulations prove that the proposed method is better than the state-of-the-art methods in terms of network coverage, energy efficiency, and lifespan. The suggested approach has the ability to optimize the deployment of WSNs, which might find use in many fields like as healthcare, environmental monitoring, and surveillance. More efficient and effective methods of optimizing WSN deployment may be created with the help of this study.In order to establish a minimum distance between cluster members, J. Wen et al. [16] created a clustering method that relies on analyzing an adjacency matrix. Our approach allows nodes to be assigned to multiple cycles simultaneously as long as they do not migrate during the applicable phase. Our approach requires us to recalculate the adjacency matrix only based on local information when restoring functionality following a node failure. The establishment of wireless sensor networks in underserved regions becomes much easier with our approach. Cluster heads that depend on connectivity can steer the mobile robots and drones to centralized locations, thereby facilitating faster data gathering and command transmission. The resource block management work proposed by Mustika et al. [17] aims to reduce interference between different levels of a heterogeneous network. In this comparison between Discrete Moth Flame Optimization and its modified version, the interference management optimization is emphasized. As far as performance improvements are concerned, the modified form comes above the previous DMFO, because of higher resource block usage, lesser unused slots, and less interference. Using a cumulative distribution function, an average improvement of 2.50 dB in Signal-to-Interference Noise Ratio is presented by SINR. For analyzing data related to 5G IoT networks, HOHDST data techniques were therefore introduced by P. Kumar Sharma et al. [18]. This proposed method combines the STF and DP approaches without losing data privacy, allowing it to remain sensitive and also analytically rigorous. This suggested method to enhance the protection of privacy applies perturbation in the gradient, reduces the gradient, and adds noise to the tensor. Experimental studies with real datasets demonstrate that this proposed strategy surpasses current alternatives regarding analytical accuracy and protection of privacy. Among the many possible applications of the proposed approach are anomaly detection, intrusion detection, and network traffic analysis. The proposed approach advances techniques for safely and dependably analyzing HOHDST data sent over 5G IoT networks.

This paper explores the work of Pandey et al. [19] to determine how WSNs might be made to last longer by combining supernodes with Moth Flame Optimization (MFO). When it comes to energy-efficient routing and clustering, heterogeneous WSNs employ MFO. This research compares and contrasts the performance of MFO with that of other optimization methods, such as GA and PSO. We study the deployment of the base station and various populations of supernodes and sensor nodes in relation to the lifetime and efficiency of the network. This study provides an approach for enhancing coverage for WSNs while reducing power consumption (Yao et al., 20). It proposes a hybrid approach of a variable spiral position update and an adaptive inertia weight method to enhance the MFO algorithm's local and global search ability. The main aim of the research is to find ways in which power consumption at the time of node deployment can be reduced without decreasing coverage. In comparison with the previous approaches, in the new algorithm VF-IMFO, average node movement distance is reduced and higher coverage rates are achieved, as depicted in simulation results. To help reduce the power consumption of wireless microsensor networks, A. P. Chandrakasan et al. [21] used the low-energy adaptive clustering hierarchy approach that is known as the LEACH protocol. The design proposed here enhances the system's lifespan, latency, and perceived quality by integrating application-specific data aggregation with cluster-based media access and routing. By implementing innovative algorithms of cluster adjustment energy consumption distribution based on rotating head positions of a cluster, the applications suited data aggregation, the designed architecture allows the several nodes to be self-organized. The overall design for a wireless microsensor network is highly marked by minimal latency, reliability and energy efficiency. The suggested design outperforms its predecessors at two areas where energy efficiency, and network life time are involved. The purpose of this proposed research is to improve the performance and reduce energy consumption in wireless microsensor networks. With such a goal, the suggested research has greatly contributed to the development of efficient and effective protocols. Clustering WSNs using the particle swarm optimization technique, in order to eliminate residual nodes, was suggested by C. Vasanthanayaki et al. [22]. To reduce the formation of redundant residual nodes, the proposed method optimizes the cluster head generation mechanism by PSO. In comparison with other clustering protocols in vogue currently, the proposed method is superior in terms of energy consumption, throughput, packet delivery ratio, and network lifetime, among others. The study presented here might help the wireless sensor network advance the development of effective clustering algorithms, thus increasing performance with decreased power consumption. A complete analysis of the proposed approach, its execution, and efficacy is provided in the paper. The proposed approach can be used to prolong the lifetime and enhance the performance of WSNs for many applications. Rajan et al. [23] aim to prolong the operational lifetime of WSNs by better utilization of their battery resources through optimization of routing and clustering using Moth Flame Optimization. We compare it with a wide variety of optimization techniques. This includes Genetic Algorithm, Particle Swarm Optimization, Least Distance Clustering Algorithm, and many others. It reveals that MFO can extend the lifetime of WSN networks using the more efficient routing and clustering techniques in networks. W. Fei et al. [24] proposed a novel composite clustering technique to enhance underwater sensor network performance based on fuzzy-c-Means and the MFO. The suggested approach begins by creating power-efficient clusters using Fuzzy-c-Means. Then for every cluster, it uses MFO to choose the best cluster head. Relative to the efficiency in data transmission, network dependability, and resource consumption, the proposed clustering method surpasses competing techniques. The mentioned above approach enhances clustering methods in an efficient and effective way, which might be helpful for creating underwater sensor networks. This research will outline the plan, show you how to implement it, and then assess how well it worked. The efficiency and longevity of underwater sensor networks might be improved by this.

Ashish Pandey et al. [25] proposed an innovative approach to enhance the average lifetime of heterogeneous WSNs using energy-aware routing and energy-efficient clustering. The ratio of supernodes and sensor nodes used to compute metrics varied in the sensing area of the WSN. For the test location of the base station in the sensing area, the number of hops for the entire lifetime is also considered. We discuss how to influence the lifespan with the optimized parameter and compare performance using different genetic bio-inspired algorithms. The WSN is prone to a number of problems that are unavoidable when sensors are deployed remotely inside a limited design.

To improve the fault detection ability of WSN with low energy detection, Hitesh et al. in [27] proposed a scheme titled partitioned energy efficient low energy adaptive clustering hierarchy (PE-LEACH) algorithm. The PE-LEACH algorithm is an extension to LEACH. Ideology would be discussed in the subsequent session; however, this scheme provides an overview of how IMFO, the partition-based MFO, would be used in order to further improve the PDR.

2.2 Wireless Sensor Network

It can configure WSN sensors to report the whereabouts of specific people or even the local weather. A "hub" or "sink" node collects data gathered by the sensors and transmits it further. When "sink node" captures information, it keeps sending. Generally, such applications as agricultural, environmental, and military surveillance would make considerable use of WSNs because such networks allow faraway monitoring of environmental data. Others apart from these include low power consumption, deployment and data collection capacity in areas that would otherwise be impassable or hazardous to human life. Healthcare providers also utilize WSNs in tracking patients' vital signs, blood sugar levels, and drug delivery. Patients may expect reduced healthcare costs, better quality of care, and shorter hospital stays with WSNs. Some tools that may be used to design the WSN effectively include routing protocols, energy-efficient communication protocols, data fusion and aggregation methods, and optimization algorithms. In fact, optimizing WSNs is an enormous area of research because doing so is essential in maintaining the efficiency, reliability, and robustness of the network. Using optimization methods to find the optimal design characteristics of a network, such as the number of nodes, location, transmission power, and routing patterns, maximizes network performance while reducing energy consumption and assuring connection and coverage. The WSN is depicted in Fig. 1.

A formal mathematical model specifies the WSN optimization problem's decision variables, objective function, and constraints. Considerations include the quantity of power a sensor node sends, the mechanism used to aggregate data in a network, and the availability of periods for the nodes. The objective is to minimize WSN energy consumption without jeopardizing vital network performance indicators.

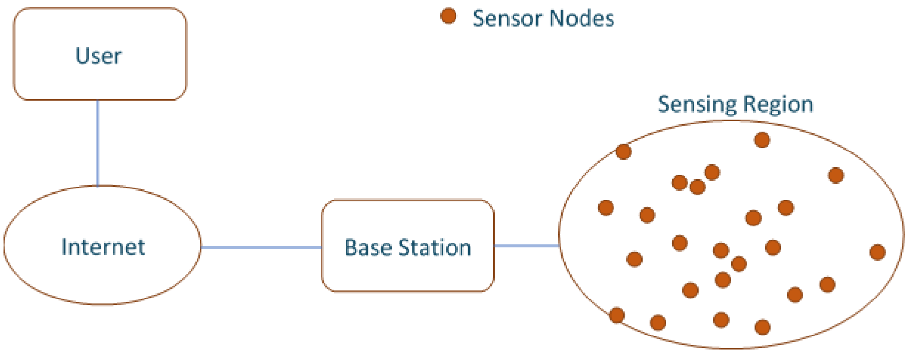

Fig. 1 Basic WSN network

Rest assured, the capacity of this network will be maintained by these constraints. Here is an expanded version of the mathematical model: With N sensor nodes in the WSN and T being the maximum data transmission rate, it can be depicted how the network works. Power of transmission for each node I, where I = 1, 2,., N, is a selection variable for this optimization problem. Data aggregation (DA) in the network can be either centralized or distributed. Assign time slots 〖T_s〗_i for the sensor nodes based on the sum of transmission periods of node I (where I = 1, 2, N) and I is an integer greater than zero. The optimization problem aims at minimizing the energy used by WSN, which is expressed as equation 1:

$$\min E = \sum_{i=0}^{N} P_i T_{s_i} \tag{1}$$

The packet delivery ratio (PDR) limitation guarantees that the base station receives a specified proportion of packets from sensor nodes. Delay constraint ensures that packet delivery delays do not exceed a specified threshold. Lifetime limitation on the network assures that the network stays functioning for a certain amount of time. The limits are mathematically represented as equation 2:

$$\begin{Bmatrix} PDR \geq PDR_{min} \\ Delay \leq Delay_{max} \\ Network\ Lifetime \geq LT_{min} \end{Bmatrix} \tag{2}$$

PDR_{min}, $Delay_{max}$, and LT_{min} represent the minimal packet delivery ratio, delay, and network lifespan. MFO solves the optimization problem by identifying the parameters that should be adjusted to maximize the WSN's efficacy while minimizing its power consumption. Metaheuristic optimization techniques such as the MFO method, which follows the trajectory of light-trapping moths have been useful in solving optimization issues across various domains. In order to solve a problem that would enable the energy consumption within a certain boundary and within more acceptable network performance standards, a mathematical model is constructed based on an optimization issue for WSN problems through a choice variable, an objective function, and constraints. Solving the model by applying the IMFO method.

2.3 Proposed Inspired Moth Flame Optimization Algorithm (IMFO)

This type of metaheuristic optimization techniques, such as the IMFO approach, follows the actual behavior of moths in the wild. To find the optimum solution, IMFO, just like any other population-based algorithms, randomly explores the space of search. As is depicted in Fig 2, the algorithm has been developed from the moths' behavior pattern. Moths use the moon as a landmark and are drawn to bright sources of light like fires. Applying the moth flame concept, IMFO can now optimize the positions of candidate solutions and search for the global optimum. This strategy, as shown in Figure 3, can solve optimization problems having nonlinear and non-convex objective functions. It also shows a much higher convergence rate compared to other metaheuristic algorithms. The areas of control engineering, image processing, and renewable energy optimization have proved the fruitful use of IMFO. In order to optimize

WSN parameters and improve WSN performance, the IMFO approach has been proved to be effective in simulation experiments.

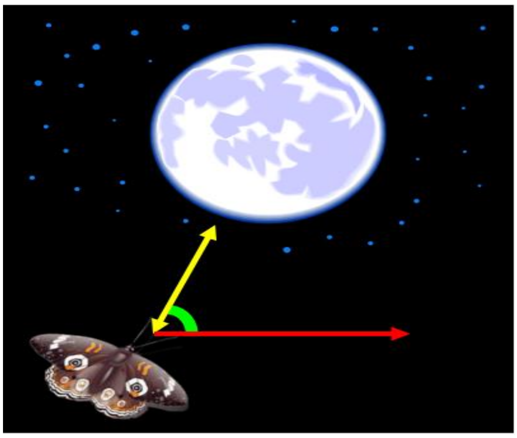

Fig 2. Illustration of IMFO

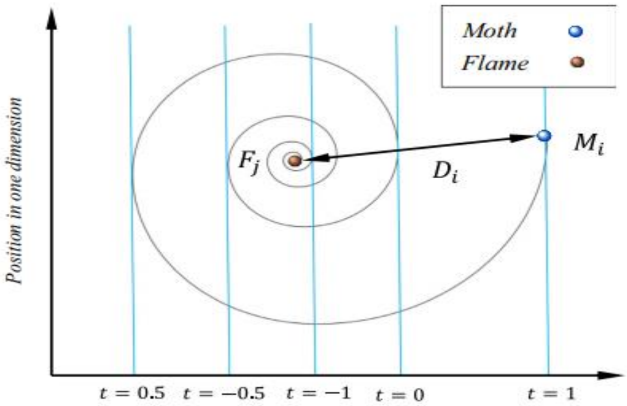

Fig 3. Graphical representation of IMFO

2.4 Optimization of IMFO in WSN

In IMFO, there is a meta-heuristic optimization technique based on the natural behavior of moths to fly towards any source of light. This activity formed the basis for the system's design. The flow diagram in Fig. 4 provides the algorithm for IMFO to maximize coverage while lowering energy usage. In order to determine where in a WSN nodes should be placed, it does a number of scans across the search space. The members of IMFO are moths from two distinct species. Here, we have moths, and more specifically, burning moths. In this metaphor, moths stand for the best local solution and flames for the best worldwide one. The placement of the fire and other moths within the search area influences the random flight of moths in every iteration. In addition, attraction of the moth towards the flames and repulsion from each other balance out to lead its flight.

To know how good a solution is, the optimization process evaluates the fitness function at each iteration. In WSN optimization, the fitness function usually considers the network's coverage, node energy consumption, and degree of connectedness. When an optimization process reaches an endpoint, such as a certain number of iterations or a certain degree of convergence, it stops. These two alternatives show possible termination circumstances. By a significant margin, IMFO outperforms competing optimization techniques. It can easily handle optimization problems that are neither linear nor convex, and its convergence rate is much faster than that of many other approaches. As it is a population-based algorithm, IMFO is able to explore the whole search space and avoid becoming stuck when it can only discover one ideal solution. The IMFO algorithm determines the appropriate places for placing WSN nodes based on a moth's behavioral trait, such as being attracted to a flame. This successful optimization technique may apply to wireless sensor networks in the aspects of coverage, energy, and connectivity. As an application in routing, IMFO is presented to locate the optical sensor located within the region of coverage but with low compilation time.

Utilizing a metaheuristic algorithm that draws inspiration from natural habitat behavior—be it hunting style or directional localization—is a noteworthy discovery for routing purposes. Figure 4 depicts the IMFO's process.

Set the initial values of the parameters, including the area covered and the number of sensors within the region. Each moth represents the position sensor of the Sensor, and their numbers are randomly distributed over the region. Find the moth in each of the n subregions that comprise the larger region. Find the moth with the highest fitness as flame by calculating its fitness value. The location of the moth is updated for any little change in the movement of the parameter beta. The procedure repeats again until the optimal moth is identified as the flame by estimating the fitness value. The iteration concludes with the selection of the optimal moth in each subregion. This is the total population of the area, and it is from here that the search for the ideal candidate solution in the area starts. This partition-based approach decreases power usage and increases, as mentioned in the results session.

2.4.1 Initialization

During the start-up step of the IMFO method, a small number of moths are made. The information from these people is then used to find the best solution to the problem. A random group of moths is made as part of the plan to improve the WSN. Each moth is a possible answer to the issue of optimization. As part of setting up for the operation, this is done. With the help of a random number generator, we will give each moth in the group a different starting value. The capacity of the sensor nodes to send data, the way the network collects data, and the time slots given to the sensor nodes all significantly impact the WSN optimization problem. The values of these selection variables are generated randomly within their respective ranges, which are determined by the optimization problem's constraints. Table 1 displays the ranges of the decision variables used during the initialization phase of the WSN optimization problem.

Table 1 Interval Decision Variables Affecting WSN Optimization Problem

Decision Variable	**Range**
Transmission Power (P_i)	$[0, P_{max}]$
Data Aggregation Method (DAM)	{Centralized, Distributed}
Allocation of time slots (Ts_i)	[0, T]

T is the no. of feasible transmission time slot, Pmax is the maximum transfer power. For the first population of moths, a fitness function is created that determines each individual's fitness relative to the solution quality it produces. In order to improve the solutions further, another generation of moths comprising those with the maximum fitness values has to go through movement and update methods.

2.4.2 Fitness Evaluation

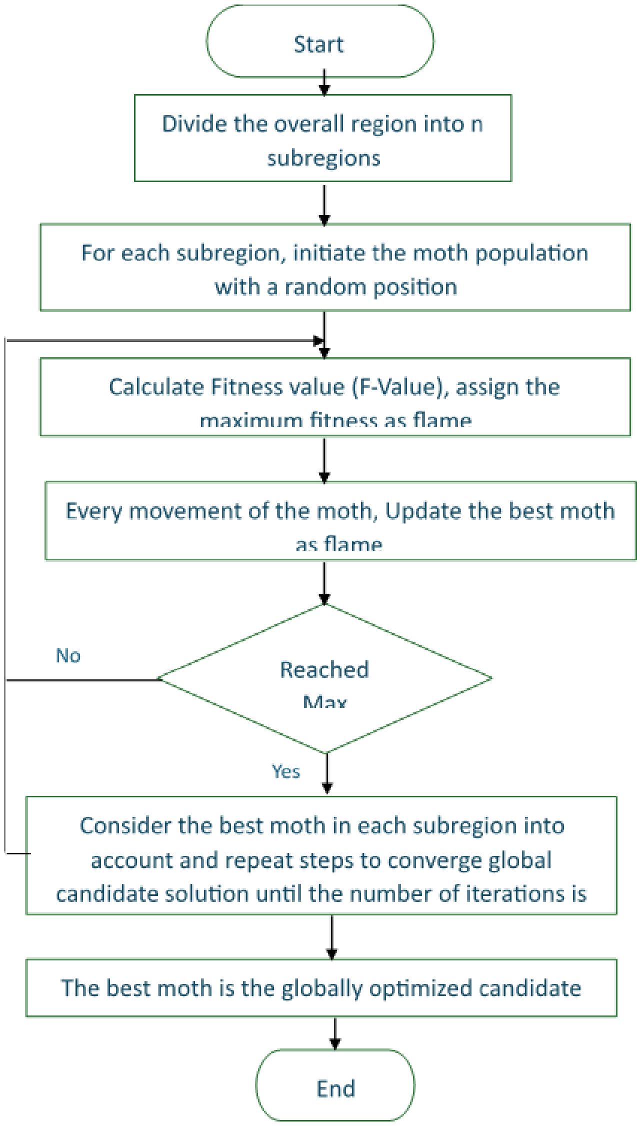

Fig. 4 Flowchart for IMFO algorithm

A fitness evaluation is performed to evaluate the success of IMFO-based optimization for WSNs. Multiple metrics are used to evaluate the network's effectiveness. These include the distance it travels, energy consumption, and the proportion of containers successfully delivered. The objective function's value is used by the IMFO method to evaluate a solution candidate's viability. When optimizing for WSN performance, the objective function represents the metrics being optimized for. Every potential candidate is tested repeatedly until the best one emerges. The IMFO algorithm finds the optimal solution by continually updating the moth's position. The population's top and bottom solution choices refine the moth's location. We utilize MATLAB to assess the fitness function quickly and accurately. These strategies use several optimization techniques to determine the optimal fitness function value. Table 2 displays the fitness function's performance metrics and their corresponding weights.

Table 2 Performance metrics and associated fitness function weights

Performance Metric	**Weight**
Network Lifetime	0.4
Network Coverage	0.2
Energy Consumption	0.3
Data Delivery Rate	0.1

The fitness function maximizes network longevity, coverage, and data transmission rate while minimizing energy usage. The values of these performance measures are derived based on the WSN configuration that each member of the population represents. Network coverage is determined by sensor nodes' sensing range and location, while energy consumption is determined by transmission power and distance between sensor nodes. After calculating the fitness value of each population member, those with the highest fitness values are selected to produce the next generation of moths. To better improve the solution, these persons are relocated and updated.

2.4.3 Movement and Updating

The Movement stage of the Moth Flame Optimization (IMFO) method entails moving the moths toward the flames, which indicates viable solutions in the search space. The moth's location governs the moth's movement, the flame's position, and the algorithm's current iteration. The movement equation 3 used by the suggested approach is:

$$x_{i(t)} = x_{i(t-1)} + beta\_0 * e^{-gamma * t} * (flame\_position - x_{i(t)}) \tag{3}$$

where e is the Euler's number, flame_position is the position of the flame, beta_0 and gamma are movement parameters, and x_(i(t)) is the position of the ith moth at a time, which is t. In each iteration, the location is slightly changed according to the position of the moth (tracking sensor) and the position of the flame (fixed sensor), so the importance of Euler's number in Equation 3. Reduced now is the moth's location compared to the flame. The exponential growth cares more for the local minimum value rather than the linear increase. Convergence occurs during the movement stage of IMFO as the moth ideally converges to the flame. In the subregion, it creates feasible solutions optimized locally. More generally, a global optimum is reached where the sensors should position themselves. The movement in the sensing region is hyper-elliptical. The positions of the moths and the flames determine the way the random sensors move to reach the optimal sensor inside the WSN sensing region.

The Updating involves updating the position of each moth based on the new movement calculated in the Movement step. The updating equation 4 used in the proposed method is given by:

$$x_{i(t+1)} = x_{i(t)} + lambda * (x_{best(t)} - x_{random(t)}) \tag{4}$$

where x_(i(t+1)) represents the current location of the ith moth at the time t+1, best (t) is the position of the best moth in the population at the time t, random (t) is the location of a randomly chosen moth at the time t, and lambda is the updating parameter. At every iteration, the location of the moth is updated considering the movement. In this case, the best moth at the current iteration and the position of a random moth is used as parameters to update the position of the moth. Beta_0 and gamma values of the movement parameters as well as updating parameter lambda will be found by performing experimentations and tuning for an optimal result. The rationale of the movement updation equation lies in the lambda parameter. The optimal value is chosen through experimentation with the sensing area in order to cover the overall sensing region with minimal movement and upgradation. It contributes to optimizing every minimal change of the best position concerning the random position. Moth movement and update depend on the parameter's beta and gamma. In WSN, node coverage optimization evaluates the fitness value for every small differential change in the area, and hence, it is chosen appropriately to optimize the node coverage. The table 3 indicates the movement and updating parameter values of the method. After every iteration, just one moth's position is used best for optimization of the location of sensor location which has to be selected as 〖_(best (t))〗. Now, for every randomly localized Sensor in the sensing region, its position is updated concerning the best. In IMFO, the movement and the update of moth depend on the central focus, which is called the flame. Among all the available population, it chooses the best fitness as a flame for checking the other's fitness for every small movement by the parameter gamma and updates the position. The process iterated for maximum numbers to attain the best candidate solution. The optimal value of the parameter for the proposed method is very essential. A decrease in beta increases the compilation time. Increasing the beta might reduce any global optimum point with the best fitness value. For other applications, the value might be different, depending on how smaller differential movements will be considered each month to cover the overall area.

Table 3 Modification and revision of the parameters of the proposed method

Parameter	Value
Beta_0	0.5

Gamma	1.0
Lambda	0.5

The population of moths is subjected to repeated cycles of migration and updating until a termination threshold is fulfilled. A preset number of 100 iterations causes the algorithm to end in the suggested approach.

2.4.4 Termination

The termination criteria is a crucial component of any optimization algorithm since it specifies when it should cease searching for better solutions. The termination criteria in the proposed approach for improving the WSN utilizing IMFO sets several repetitions. Specifically, the algorithm is finished with the proposed method after a certain number of iterations, in this case, 100. This means that the algorithm will continue trying to find the best solution for the moth population until it has done 100 iterations of applying the movement and update phases. The choice of appropriate termination criteria can be crucial for how effective the optimization process will be. If it stops too early, it might take longer for the algorithm to find better answers to explore the search space. But if it stops too early, it might become stuck in local optima and miss the global optimum solution. The compromise between the computational expense of the method and the need to get great answers is used to set the termination criterion. We experimentally obtained the optimal number of iterations through trial and error. A termination criterion of a specified number of criteria, a maximum time constraint, threshold on growth in the fitness function, or a mix of these might be used for termination of iterations. Here is the pseudo-code of the IMFO algorithm with enhanced coverage:.

1. Initialize parameters, including the number of moths, maximum iterations, and search range.

2. Divide the sensing region into 'n' number of regions

3. For each sensing region n,

   a. Generate the initial moth's population with positions in a random place and fitness values.
   b. Evaluate the each moths fitness in the population using the performance metrics.
   c. Rank moths based on their fitness and assign flame to the top moth.
   d. Update the position of each moth using the formula:
      i. new_position = current_position + step_size * distance_to_flame * (position_of_flame - current_position) + beta * (rand() - 0.5)
      ii. step_size is the step size, distance_to_flame is the moth and the flame distance, beta is a constant, and rand() returns a cluster number from 0 to 1.
   e. Evaluate the fitness of each moth after the position update.
   f. If a moth has a better fitness than the current flame, replace the current flame with that moth.
   g. Decrease the step size and repeat steps 5-7 until reaching the highest number of iterations.
   h. Return the position of the flame as the locally optimized solution.

4. Initialize the local optimized solution as the moth's with the flame's position and corresponding fitness values

5. Repeat steps a-h until reaching the highest number of iterations.

6. Return the position of flame as the global optimized solution.

The entire sensing area is divided into n regions by the proposed methods. The number of subregions that are produced depends on the total size of the sensing region. The more the area, the more the regions. The optimum subregion area restriction determines the number of subregions that are produced. To find where the flame should be located within the restricted area, the IMFO algorithm considers all the constraints about the flames that are associated with each subregion. For a subregion, the location of the flame would be decided with respect to its fitness value; hence, a relation can also be seen about the movement of moths along with updates that result in its convergence within a confined subregion area. Under the optimization constraint in the general coverage area, the subregions flame locations are initialized as moth and undergo global convergence to increase the overall coverage area of the sensor nodes. Rather than doing it once for the whole area, IMFO keeps on updating and moving n subregions. There is a compromise in compilation time, but it enhances the PDR. According to the entire sensing area, n subregions can be subdivided. The proposed algorithm hence surpasses Moth Flame Optimization and all present techniques based on PDR and throughput.

3. RESULTS AND DISCUSSION

A certain number of sensor nodes is pre-planned, 50, 75, or 100, in a certain region of installation with a certain communication range. The step two consists of the settlement on a fitness function that is representing the optimization goal, be it an increase of the network coverage area or decrease of the energy usage. It uses all subsequent start, movement, and update phases of the IMFO algorithm. After performing a certain number of iterations, the algorithm stores the optimal solution. Various metrics can be used to measure the quality of the designed strategy. The following points should be considered while evaluating any optimization technique: Significant experimentation is needed to solve the WSN problem as there are many possible network configurations, deployment scenarios, and objectives for optimization. For this purpose, we have attempted all of the proposed methods with MatlabR2022, Table 4 summarizes the results of simulations performed on a network with different settings and 50, 75, and 100 sensor nodes dispersed across a 100 m × 100 m region.

Table 4. Configuration Parameter

Simulation Parameter	**Value**
Population size	50,75,100 nodes
Deployment area	100 x100 m^2
Subregion area	25 x25 m^2
Initial energy for each node	0.5 J
Maximum iterations	400

Important for the evaluation of the reliability and effectiveness of an approach is a finding from research. From the results, we get the relative effectiveness of the proposed strategy and the ones currently in place in the sense of optimizing a wireless sensor network using Moth Flame Optimization. But if we aim to get interesting conclusions and insights from these numbers, we should evaluate and know them properly. The results might be used for better and effective optimization of a wireless sensor network. In the Table 5 and Fig. 5, a comparison between suggested and current techniques is shown at the nodes of 50. Comparison between proposed and current techniques in the case of 75 nodes is shown by Table 6 and Fig. 6. In the last phase of cycle, F-values of all the sensor nodes are compared with each other. Table 7 and Fig. 7 gives the result of 100 nodes. Whereas Equation 5 provide the PDR formula, equation 6 illustrate the Fitness value formula and equation 7 provide the throughput formula.

$$PDR(Packet\ Delivery\ Rate) = \left(\frac{Number\ of\ Received\ Packets}{Number\ of\ Sent\ Packets}\right) * 100\% \tag{5}$$

$$Fitness\ Value = \left(\frac{Ratio\ of\ sensor\ node\ in\ final\ stage}{comparison}\right) \tag{6}$$

$$Throughput = \left(\frac{Total\ Amount\ of\ Data\ Transmitted}{Total\ Time\ taken\ to\ transmit\ the\ data}\right) \tag{7}$$

Table 5. Table comparison for existing and proposed method for 50 nodes

	Compilation Time (s)	**F-value**	**PDR**	**Throughput**
Proposed Method [IMFO]	7.24	1.73	97.81%	23.45
PSO [8]	6.61	1.42	88.5%	4.3
MFO-SSA[12]	7.01	1.11	88.3%	17.20

EESCA [14]	6.93	1.23	92.5%	4.2
Supernode-MFO [25]	7.3	1.43	93.64%	12.2
DBCOA [26]	6.15	1.6	82%	9.12

MFO-SSA - Moth–Flame Optimization (MFO) algorithm and the Salp Swarm Algorithm

EESCA - Efficient Electro Static Discharge Algorithm

PSO –Particle Swarm Optimization

SUPERNODE-MFO–Supernode Moth Flame Optimization

DBCOA - Directed Bee Colony Optimization

Figures 5a, 5b, 5c, and 5d show the performance of Compilation Time (s), F-value, PDR, and throughput for 50 nodes. Here, the performance of the proposed design is analyzed with the IMFO algorithm. For comparison time, the proposed design attains 7.24s; for F-value analysis, the proposed method attains 1.73; for PDR analysis, the proposed design attains 97.81%; for throughput analysis, the proposed design attains 23.45, which is better than the existing design such as PSO, MFO-SSA, EESCA, SUPERNODE-MFO, DBCOArespectively.

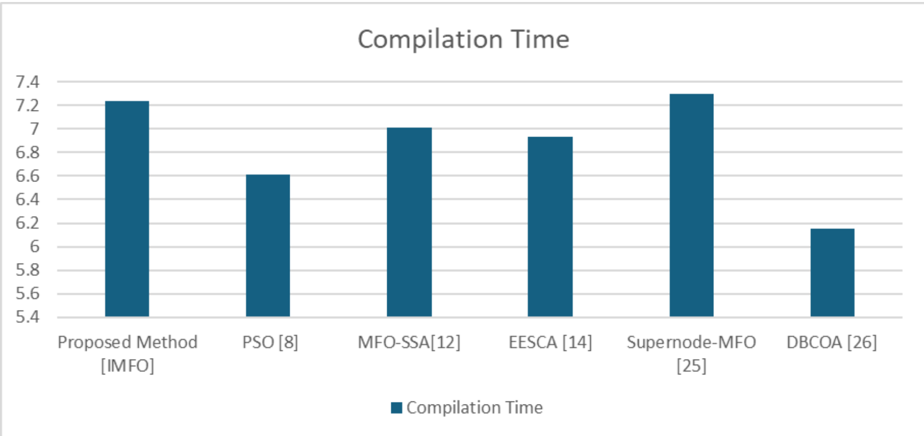

Fig. 5a) Comparison of Compilation time for IMFO, PSO, MFO-SSA, EESCA, SUPERNODE-MFO, DBCOA at 50 nodes

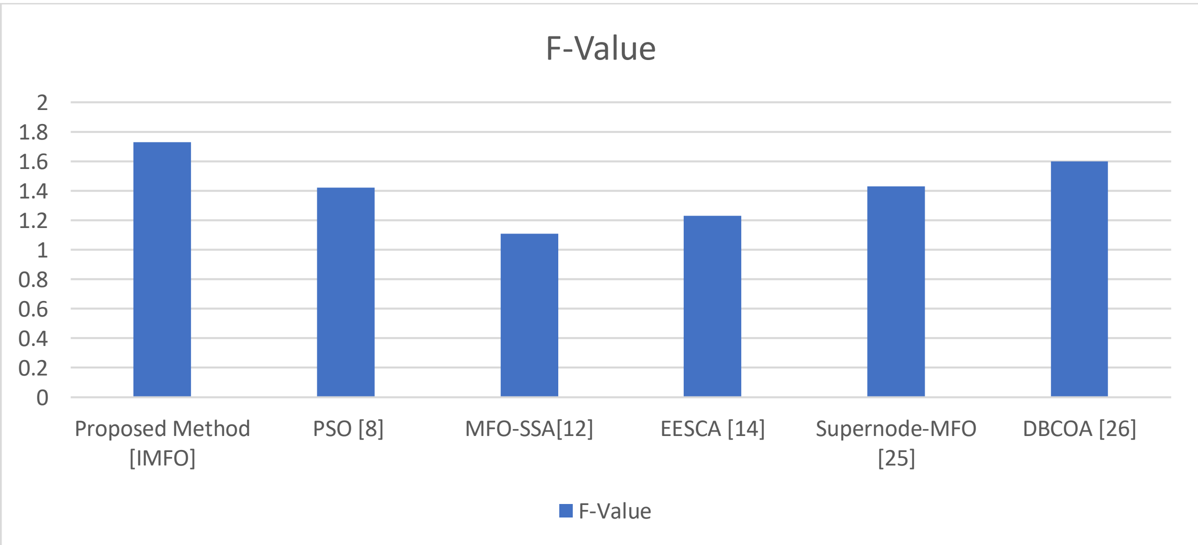

Fig. 5b) Comparison of F- value for IMFO, PSO, MFO-SSA, EESCA, SUPERNODE-MFO, and DBCOA at 50 nodes

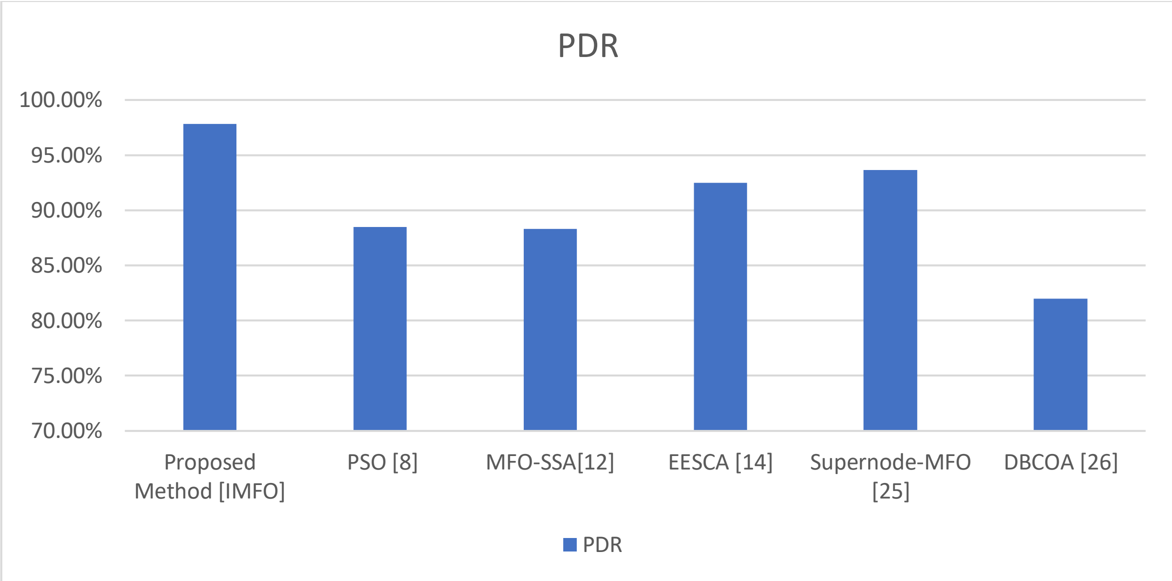

Fig. 5c) Comparison of PDR for IMFO, PSO, MFO-SSA, EESCA, SUPERNODE-MFO, DBCOA at 50 nodes

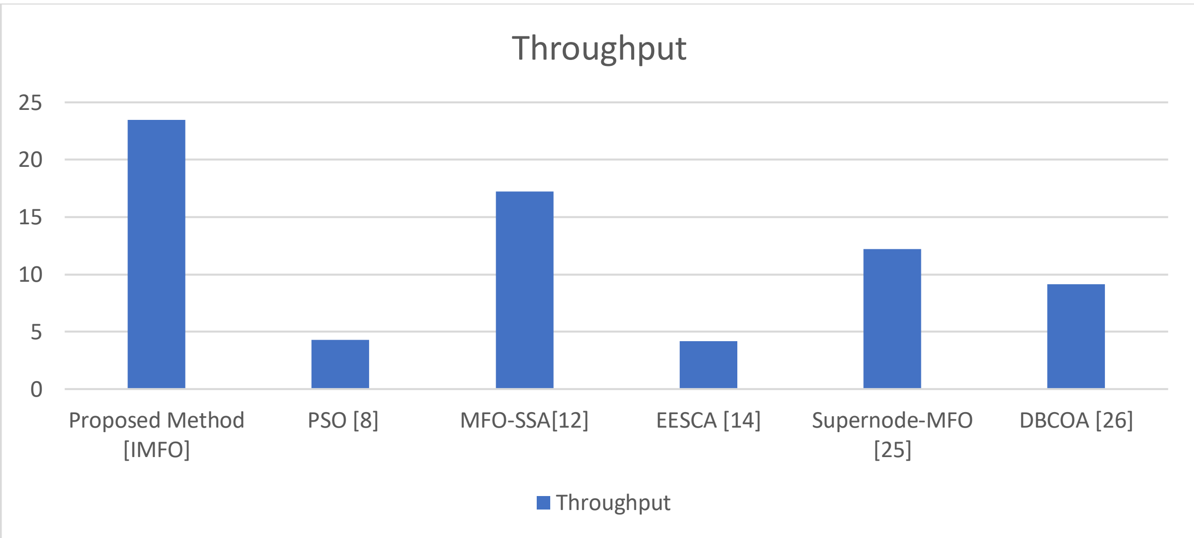

Fig. 5d) Comparison of Throughput for IMFO, PSO, MFO-SSA, EESCA, SUPERNODE-MFO, DBCOA at 50 nodes

Table 6 Table comparison of existing and proposed methods for 75 nodes

	Compilation Time (s)	**F-value**	**PDR**	**Throughput**
Proposed Method [IMFO]	8.71	1.83	96.71	22.55
PSO [8]	7.17	1.51	84.2%	3.3
MFO-SSA[12]	8.10	1.31	84.6%	15.71
EESCA [14]	7.35	1.43	91.16%	3.9
Supernode-MFO [25]	9.2	1.35	94.1%	13.5
DBCOA [26]	7.2	1.69	80%	8.3

The results of the 75 nodes' Compilation Time (s), F-value, PDR, and throughput are shown in Figures 6a, 6b, 6c, and 6d. In this case, we use the IMFO method to examine how well the suggested design works. The proposed design outperforms the current designs, including PSO, MFO-SSA, EESCA, SUPERNODE-MFO, and DBCOA, in terms of comparison time (8.71 s), F-value analysis (1.83), PDR analysis (96.71%), and throughput analysis (22.55 bps).

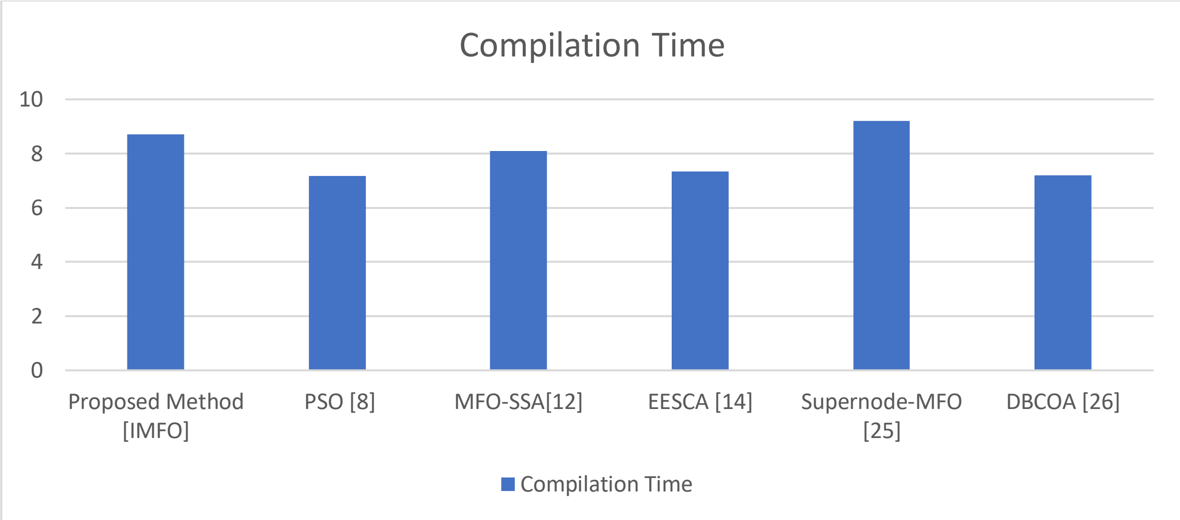

Fig. 6a) Comparison of Compilation Time for IMFO, PSO, MFO-SSA, EESCA, SUPERNODE-MFO, DBCOA at 75 nodes

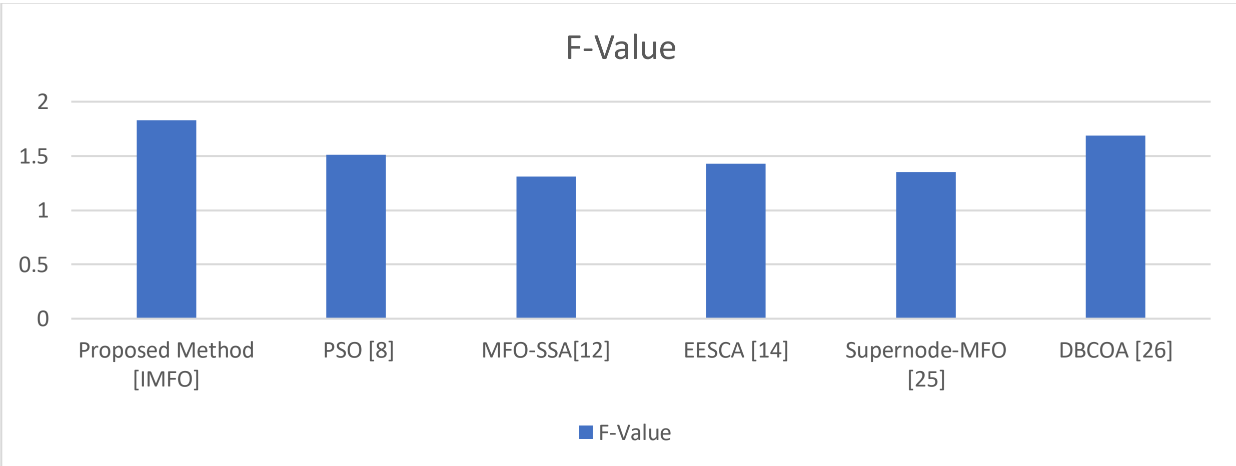

Fig. 6b) Comparison of F-Value for IMFO, PSO, MFO-SSA, EESCA, SUPERNODE-MFO, DBCOA at 75 nodes

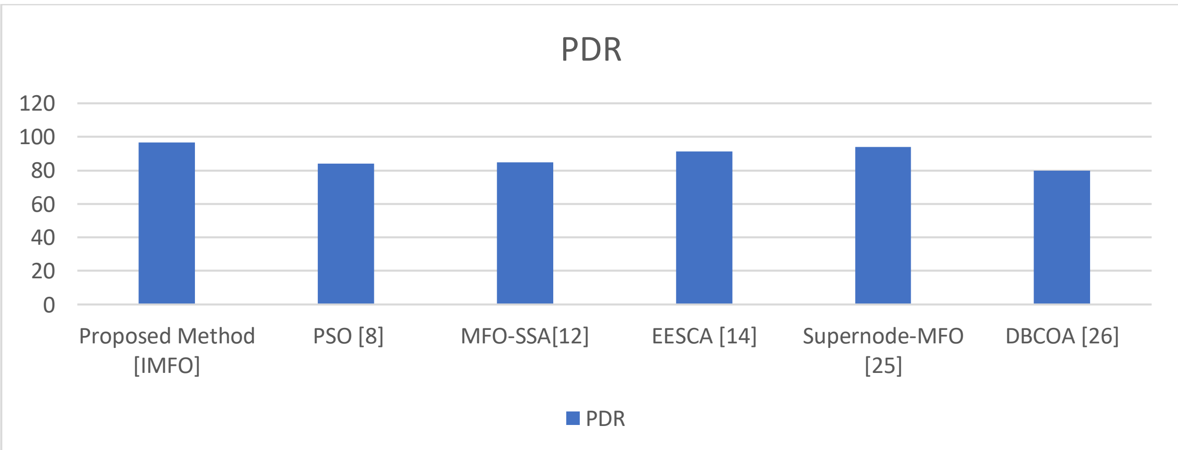

Fig. 6c) Comparison of PDR for IMFO, PSO, MFO-SSA, EESCA, SUPERNODE-MFO, DBCOA at 75 nodes

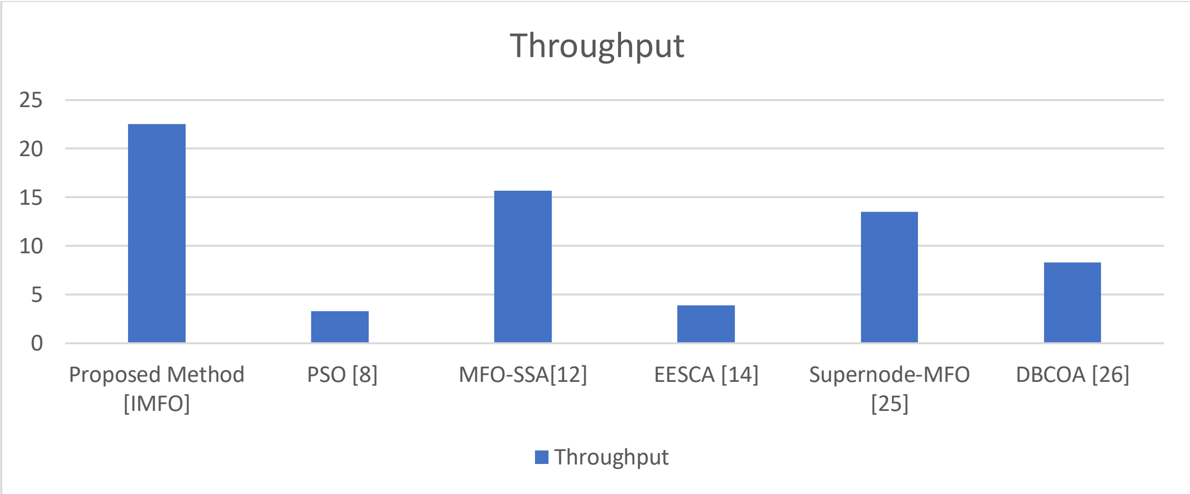

Fig. 6d) Comparison of Throughput for IMFO, PSO, MFO-SSA, EESCA, SUPERNODE-MFO, DBCOA at 75 nodes

Table 7 Table comparison of existing and proposed methods for 100 nodes

	Compilation Time (s)	**F-value**	**PDR**	**Throughput**
Proposed Method [IMFO]	8.24	1.97	96%	21.98
PSO [8]	8.5	1.62	81%	2.7
MFO-SSA[12]	7.97	1.57	86%	14.5
EESCA [14]	12.54	1.84	89.5%	3.1
Supernode-MFO [25]	10.78	1.78	93.5%	12.56
DBCOA [26]	8.21	1.8	76..78%	7.5

Figures 7a, 7b, 7c, and 7d depict performance metrics for 100 nodes, which include Compilation Time (s), F-value, PDR, and throughput. In this case, an algorithm is used to check how well the proposed design works. The proposed design has outperformed the other designs, including PSO, MFO-SSA, EESCA, SUPERNODE-MFO, and DBCOA, in comparison time, which is 8.24 s, F-value analysis, which is 1.97 s, PDR analysis, which was 96%, and throughput analysis, which was 21.98 s. The number of subregions enhanced the convergence speed by a factor of 'n' compared with the MFO method. Since WSNs are housed in a hostile environment, it is taking more time to find the sensor in MFO. Divide difficult areas into smaller ones in IMFO. From that point onwards, the speed of convergence was multiplied by 'n.' To obtain the global sensors, the time of convergence increased. A hostile environment will characterize the coverage region in real time. The assessment shows that sensors are spread out over the sensing area at random, but there are still problems with recreating the real-world conditions in a virtual setting. Further research has to be done on how to split it to get the number of subregions which, in IMFO, is important.

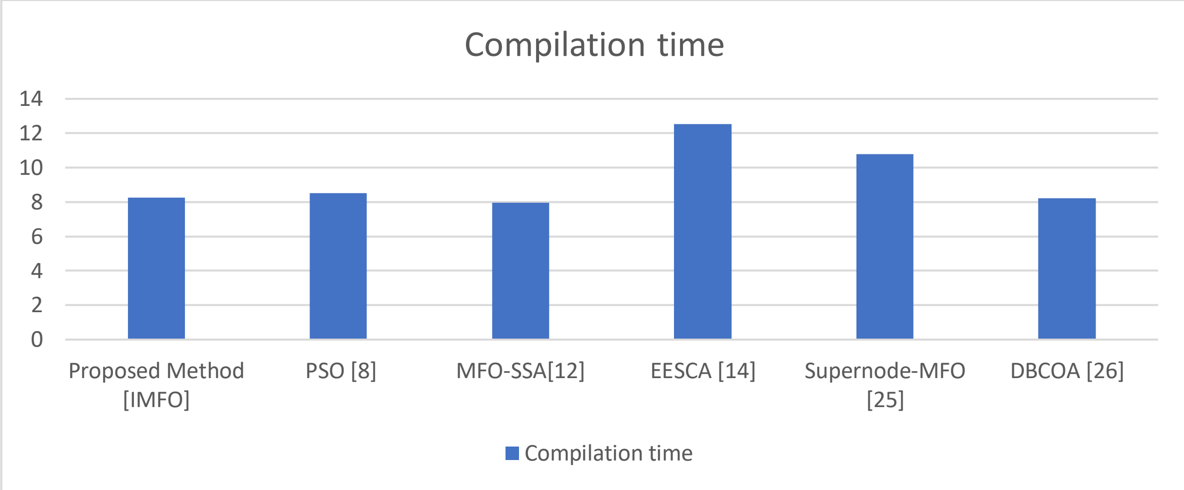

Fig. 7a) Comparison of Compilation time for IMFO, PSO, MFO-SSA, EESCA, SUPERNODE-MFO, DBCOA at 100 nodes

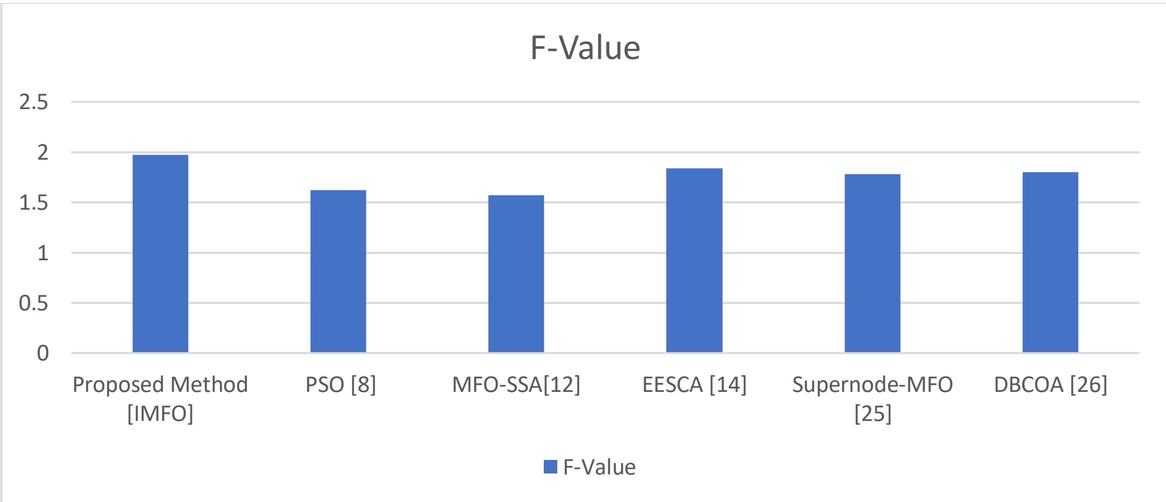

Fig. 7b) Comparison of F-value for IMFO, PSO, MFO-SSA, EESCA, SUPERNODE-MFO, DBCOA at 100 nodes

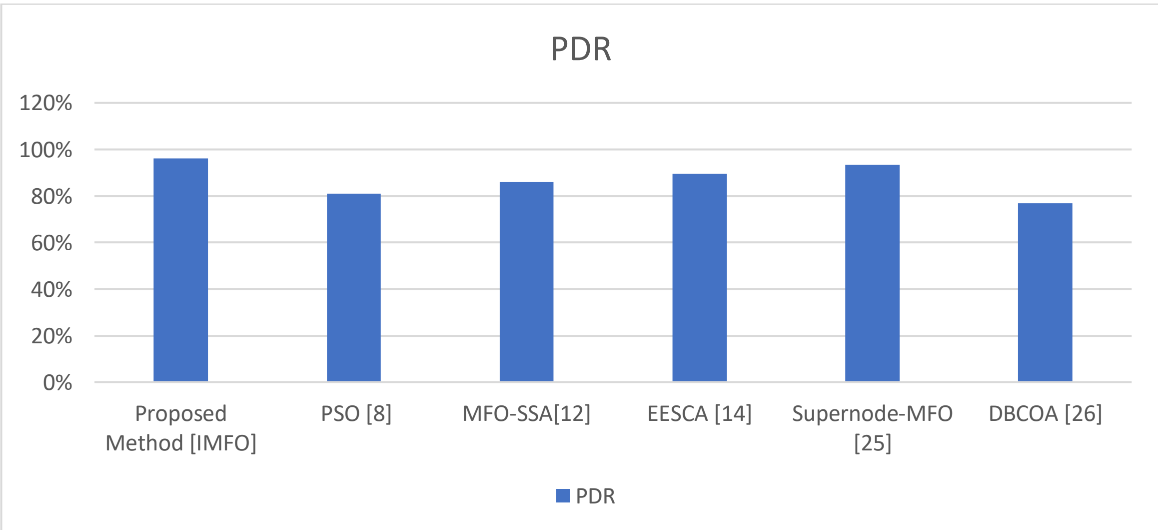

Fig. 7c) Comparison of PDR for IMFO, PSO, MFO-SSA, EESCA, SUPERNODE-MFO, DBCOA at 100 nodes

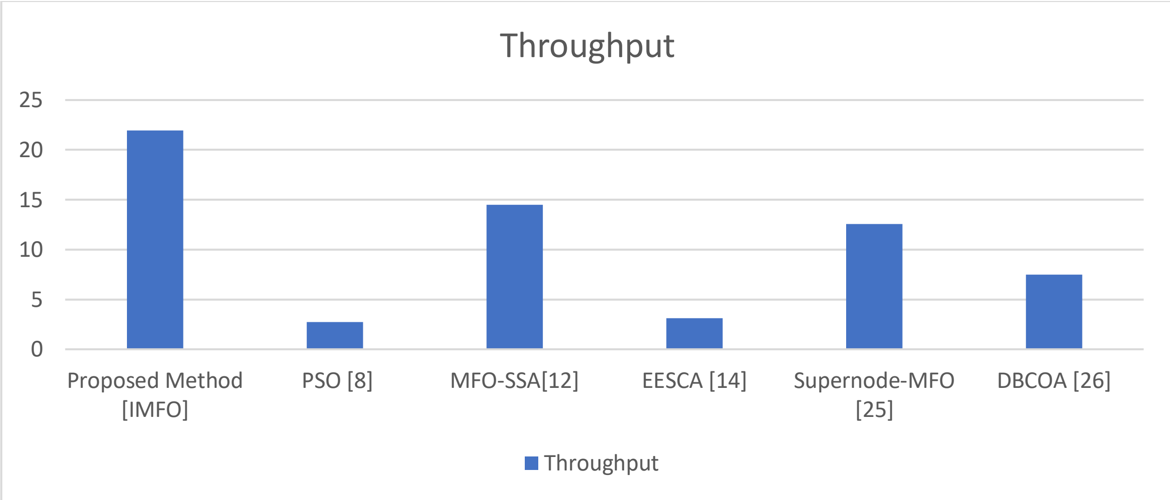

Fig. 7d) Comparison of Throughput for IMFO, PSO, MFO-SSA, EESCA, SUPERNODE-MFO, DBCOA at 100 nodes

4. CONCLUSION

Based on the findings of this research, an Inspired Moth Flame Optimization method may be utilized for the optimization of Wireless Sensor Networks. Simulation experiments using MATLAB have shown that WSNs optimized by the proposed IMFO had better F-value, packet delivery ratio, and throughput as compared to WSNs optimized using standard approaches. Inspired to detect the sensor in the sensing zone without other sensors invading the head node termed flame, the moth flame algorithm and its trajectory discover the flame in the transverse nature. To further enhance the PDR or throughput in IMFO, we provide the idea of doubling down on the overall coverage area refinement by breaking it down into subregions. A final improvement over MFO is a 1.03% increase in observed compilation time and a 1.12% rise in PDR when the subregions are included into the overall area. An experiment is carried out using parameters like compilation time, PDR, F-value, and throughput, and the number of sensors in the coverage area is increased from 50 to 100. Even though a smaller sensor is used and the compilation time is higher, the results show that the IMFO has enhanced PDR. We achieved 1.1%, 1.14%, and 1.18% of PDR at 50, 75, and 100 nodes, respectively. Compilation time with improved PDR increases with the number of subregions. It is essential to find the optimal number of subregions while increasing the overall area. It was also suitable for choosing energy-aware cluster heads, which could further enhance PDR and throughput. Based on the results, IMFO is a viable option for enhancing the functionality of WSNs in a range of contexts, such as medical, agricultural, and ecological monitoring, via parameter optimization. These results suggest that IMFO could be a useful alternative technique for WSN researchers and practitioners looking to enhance their networks. Finally, the proposed method has promise as an optimization tool for WSNs; it has the potential to improve application efficiency, data delivery rates, and network performance while reducing compilation time. In the future, we will analyze the node's energy usage with respect to its design specifications.

ABBREVIATIONS

WSN - Wireless Sensor Network

IMFO - Inspired Moth-Flame optimization method

MFS - Moth Flame Search

DA - Data Aggregation

EPEGASIS - Enhanced Power-Efficient Gathering in Sensor Information Systems

STF - Sparse Tensor Factorization

DP - Differential Privacy

PSO - Particle Swarm Optimization

CHs - Cluster Heads

MFO-SSA - Moth–Flame Optimization (MFO) algorithm and the Salp Swarm Algorithm

EESCA - Efficient Electro Static Discharge Algorithm

HOHDST - High-Order, High-Dimension, and Sparse Tensor data

LEACH - Low-Energy Adaptive Clustering Hierarchy protocol

PSO –Particle Swarm Optimization

SUPERNODE-MFO– Supernode Moth Flame Optimization

DBCOA - Directed Bee Colony Optimization

PDR - Packet Delivery Ratio

DECLARATIONS

Acknowledgments

The authors acknowledged the Department of ECE, Saveetha Engineering College, for the laboratory support in this research.

Author Contributions

All Authors contributed equally.

Availability of Data and Materials

The datasets used and analyzed during the current study are available from the corresponding author upon reasonable request.

Ethics approval and consent to participate.

This article contains no studies with human participants or animals performed by authors.

Funding

Funding needs to be received for this work.

Consent for publication

Not applicable.

Competing interests

The authors declare that they have no competing interests.

REFERENCE

[1] Trinh, C., Huynh, B., Bidaki, M. *et al.* Optimized fuzzy clustering using moth-flame optimization algorithm in wireless sensor networks. *ArtifIntell Rev* 55, 1915–1945 (2022).

[2] Mistarihi, M.Z.; Bany Salameh, H.A.; Alsaadi, M.A.; Beyca, O.F.; Heilat, L.; Al-Shobaki, R. Energy-Efficient Bi-Objective Optimization Based on the Moth–Flame Algorithm for Cluster Head Selection in a Wireless Sensor Network. Processes 2023, 11, 534.

[3] M. Mukherjee, L. Shu, R. V. Prasad, D. Wang, and G. P. Hancke, "Sleep scheduling for unbalanced energy harvesting in industrial wireless sensor networks," *IEEE Commun. Mag.*, vol. 57, no. 2, pp. 108-115, Feb. 2019.

[4] Lin L, Shi Y, Chen J, Ali S (2020) A Novel Fuzzy PID Congestion Control Model Based on Cuckoo Search in WSNs. Sensors 20:1862

[5] El-Omari, D.N.K.T. Sea Lion Optimization Algorithm for Solving the Maximum Flow Problem. Int. J. Comput. Sci. Netw. Secur. 2020, 20, 30–68.

[6] Singh S, Singh S, Banga VK (2020) Design of fuzzy logic system framework using evolutionary techniques. Soft Comput 24:4455–4468

[7] Syed, M.A.; Syed, R. Weighted Salp Swarm Algorithm and its applications towards optimal sensor deployment. J. King Saud Univ. -Comput. Inf. Sci. Jul. 2022, 34, 1285–1295

[8] Shehab, M.; Abualigah, L.; AlHamad, H.; Alabool, H.; Al Shinwan, M.; Khasawneh, A. Moth–flame optimization algorithm: Variants and applications. Neural Comput. Appl. 2020, 32, 9859–9884.

[9] Wang, R.; Wu, N.; Yu, X. Design of Digital Aerial Photography System for Unmanned Aerial Vehicle Based on Wireless Sensor Network. J. Appl. Sci. Eng. Technol. Educ. 2020, 14, 12.

[10] Anandh, S. J., & Baburaj, E. (2020). Energy efficient routing technique for wireless sensor networks using ant colony optimization. *Wireless Personal Communications, 114*(4), 3419–3433.

[11] Pandey, A., Rajan, A., Nandi, A. *et al.* Lifetime Enhancement of Sensor Networks by the Moth Flame Optimization. *Wireless PersCommun* 118, 2807–2820 (2021).

[12] Mistarihi, Mahmoud &Bany, H &Alsaadi, Mohammad &Heilat, Laila & Al-Shobaki, Raya &Alsaadi, Mohammad &Heilat, Laila & Al-Shobaki, Raya& Al Saadi, Mohammed. (2023) "Energy-Efficient Bi-Objective Optimization Based on the Moth-Flame Algorithm for Cluster Head Selection in a Wireless Sensor Network." Processes. 11. 10.3390/pr11020534.

[13] F. Fanian, M. K. Rafsanjani, and A. B. Saeid, "Fuzzy multi-hop clustering protocol: selection fuzzy input parameters and rule tuning for wsns," *Applied Soft Computing*, vol. 99, Article ID 106923, 2021.

[14]G. C. Jagan, P. JesuJayarin, "Wireless Sensor Network Cluster Head Selection and Short Routing Using Energy Efficient ElectroStatic Discharge Algorithm," *Journal of Engineering*, vol. 2022, Article ID 8429285, 10 pages, 2022.

[15] Y. Yao, S. Hu, Y. Li, and Q. Wen, "A Node Deployment Optimization Algorithm of WSNs Based on Improved Moth Flame Search," in IEEE Sensors Journal, vol. 22, no. 10, pp. 10018-10030, 15 May 15, 2022.

[16] J. Wen and W. Dargie, "A Simple Clustering Strategy for Wireless Sensor Networks," in IEEE Sensors Letters, vol. 4, no. 6, pp. 1-4, June 2020, Art no. 7500804.

[17] I. W. Mustika, F. B. Setiawan and K. Xaphakdy, "Performance Enhancement in Macro-Femto Network Using a Modified Discrete Moth-flame Optimization Algorithm," 2020 3rd International Seminar on Research of Information Technology and Intelligent Systems (ISRITI), Yogyakarta, Indonesia, 2020, pp. 215-220.

[18] P. Kumar Sharma, H. Li, S. He, J. Wang, H. Han, and L. Chen, "Multiple Strategies Differential Privacy on Sparse Tensor Factorization for Network Traffic Analysis in 5G," in IEEE Transactions on Industrial Informatics, vol. 18, no. 3, pp. 1939-1948, March 2022.

[19] Pandey, A., Rajan, A., Nandi, A. et al. Lifetime Enhancement of Sensor Networks by the Moth Flame Optimization. Wireless Pers Commun 118, 2807–2820 (2021).

[20] Y. Yao, S. Hu, Y. Li and Q. Wen, "A Node Deployment Optimization Algorithm of WSNs Based on Improved Moth Flame Search," in IEEE Sensors Journal, vol. 22, no. 10, pp. 10018-10030, 15 May15, 2022.

[21] A. P. Chandrakasan, W. B. Heinzelman and H. Balakrishnan, "An application-specific protocol architecture for wireless microsensor networks," in IEEE Transactions on Wireless Communications, vol. 1, no. 4, pp. 660-670, Oct. 2019.

[22] C. Vasanthanayaki and J. RejinaParvin "Particle Swarm Optimization-Based Clustering by Preventing Residual Nodes in Wireless Sensor Networks," in IEEE Sensors Journal, vol. 15, no. 8, pp. 4264-4274, Aug. 2019, doi: 10.1109/JSEN.2015.2416208.

[23] A. Pandey, A. Rajan and A. Nandi, "Lifetime Enhancement of Wireless Sensor Networks by using MFO Algorithm," 2018 International Conference on Computing, Power and Communication Technologies (GUCON), Greater Noida, India, 2018, pp. 868-872.

[24] W. Fei, B. Hexiang, L. Deyu, and W. Jianjun, "Energy-Efficient Clustering Algorithm in Underwater Sensor Networks Based on Fuzzy C Means and Moth-Flame Optimization Method," in IEEE Access, vol. 8, pp. 97474-97484, 2020.

[25] Pandey, A., Rajan, A., Nandi, A. et al. Lifetime Enhancement of Sensor Networks by the Moth Flame Optimization. Wireless Personal Communications **118**, 2807–2820 (2021). https://doi.org/10.1007/s11277-021-08156-1

[26] M. Rajeswari, J. Amudhavel, SujathaPothula and P. Dhavachelvan, Directed Bee Colony Optimization Algorithm to Solve the Nurse Rostering Problem, Computational Intelligence and Neuroscience, vol. 2020, Article ID 6563498, 26 pages, 2019. doi:10.1155/2017/6563498.

[27] Hitesh Mohapatra and Amiya Kumar Rath, Fault tolerance in WSN through PE-LEACH protocol, IET Wireless Sensor Systems, Vol 9, 358-365 (2019). https://doi.org/10.1049/iet-wss.2018.5229