IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received January 3, 2020, accepted January 20, 2020, date of publication January 23, 2020, date of current version January 30, 2020.

Digital Object Identifier 10.1109/ACCESS.2020.2968981

Optimal Sink Node Placement in Large Scale Wireless Sensor Networks Based on Harris' Hawk Optimization Algorithm

ESSAM H. HOUSSEIN[1], MOHAMMED R. SAAD[2], KASHIF HUSSAIN[3], WILLIAM ZHU[3], (Senior Member, IEEE), HASSAN SHABAN[1], AND M. HASSABALLAH[4]

[1]Faculty of Computers and Information, Minia University, Minya 61519, Egypt
[2]Faculty of Computers and Information, Luxor University, Luxor 85951, Egypt
[3]Institute of Fundamental and Frontier Sciences, University of Electronic Science and Technology of China, Chengdu 611731, China
[4]Department of Computer Science, Faculty of Computers and Information, South Valley University, Qena 83523, Egypt

Corresponding author: William Zhu (wfzhu@uestc.edu.cn)

This work was supported in part by the University of Electronic Science and Technology of China (UESTC), and in part by the National Natural Science Foundation of China (NSFC) under Grant 61772120.

ABSTRACT Large-scale wireless sensor network (LSWSN) is composed of a huge number of sensor nodes that are distributed in some region of interest (ROI), to sense and measure the environmental conditions like pressure, temperature, pollution levels, humidity, wind, and so on. The objective is to collect data for real-time monitoring so that appropriate actions can be taken promptly. One of the sensor nodes used in an LSWSN is called the sink node, which is responsible for processing and analyzing the collected information. It works as a station between the network sensor nodes and the administrator. Also, it is responsible for controlling the whole network. Determining the sink node location in an LSWSN is a challenging task, as it is crucial to the network lifetime, for keeping the network activity to the most possible extent. In this paper, the Harris' hawks optimization (HHO) algorithm is employed to solve this problem and subsequently the Prim's shortest path algorithm is used to reconstruct the network by making minimum transmission paths from the sink node to the rest of the sensor nodes. The performance of HHO is compared with other well-known algorithms such as particle swarm optimization (PSO), flower pollination algorithm (FPA), grey wolf optimizer (GWO), sine cosine algorithm (SCA), multi-verse optimizer (MVO), and whale optimization algorithm (WOA). The simulation results of different network sizes, with single and multiple sink nodes, show the superiority of the employed approach in terms of energy consumption and localization error, and ultimately prolonging the lifetime of the network in an efficacious way.

INDEX TERMS Large-scale wireless sensor network, Harris' hawks optimization, topology control, sink node placement.

I. INTRODUCTION

Along with the remarkable developments in wireless sensor networks (WSNs), large-scale wireless sensor networks (LSWSNs) have appeared, which are used in our daily lives for monitoring, tracking, sensing, measuring, and collecting real-time data in various settings [1], [2], such as smart buildings, health care monitoring, industrial monitoring, and other surveillance systems. The data collected by the sensor nodes in an LSWSN is relayed to a sink node for transferring to head nodes or end users. As shown in Fig. 1, a sink node is a temporary place where data is processed or redirected to the end users for different uses. Similar to other hard optimization problems posed by LSWSNs deployment, locating the sink node in LSWSNs is also a challenging task, as determining the best location of the sink node means reducing the number of message hops from a sensor node to its sink [3]. This subsequently lowers sensors' energy consumption ratios, because the process of sending and receiving data from a sensor to another consumes energy. Therefore, choosing the best location for the sink node will save the energy consumption in the whole network, which ultimately extends the network

The associate editor coordinating the review of this manuscript and approving it for publication was Abbas Jamalipour.

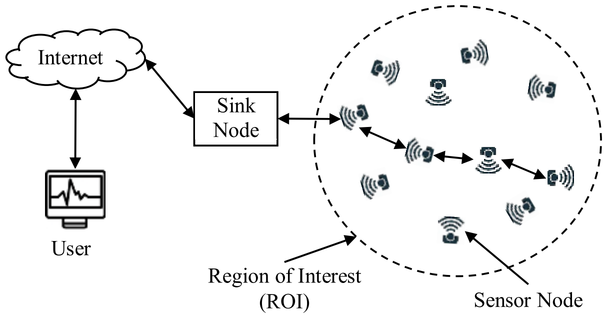


FIGURE 1. A wireless sensor network.

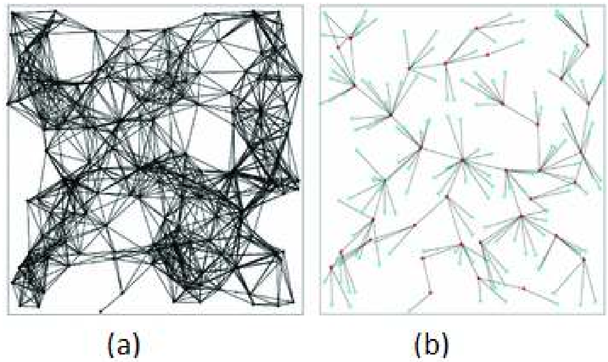


FIGURE 2. Reduced active nodes in a fully connected network. (a) A fully connected network, (b) Topology with reduced active nodes.

life. Hence, it can be inferred that the energy of LSWSNs is affected by nodes distribution and the sink node location [4].

Since wireless sensors are constrained with limited energy resources – because of being irreplaceable battery powered devices, it significantly affects the lifetime of WSNs when the sink node requires surrounding nodes to send the collected data, resulting in fast energy exhaustion by the nodes. This demands efficient workload distribution among the sensors for the enhanced network lifetime [5]. Generally, topology control is a technique, widely used in distributed computing, for making some changes in the underlying network that can be modeled as a graph to lower the distributed algorithms cost than the new resulting graphs. It is mainly used for establishing wireless ad-hoc and sensor networks. Recently, several opinions have been presented to divide the topology control algorithms into two sub-techniques. First, topology construction algorithms that are responsible for the initial reduction of the network, such as A3 [6], EECDS [7], and CDS- Rule K [8]. Second, topology maintenance algorithms that are responsible for making some changes to the first reduced topology, if it cannot perform its role completely and maintains the reduced topology in terms of connectivity and coverage like dynamic global topology recreation and static global topology rotation. The main purposes of the topology control are extending the network lifetime as a result of saving energy, reducing interference between the sensor nodes, and providing a connected topology [9].

There are many different ways to perform topology construction, such as changing the transmission range of the sensor nodes, turning off nodes from the network, creating a communication backbone, clustering, and adding new nodes to the network to preserve connectivity. Fig. 2 illustrates a topology with reduced active nodes in a fully connected network. Despite several approaches exist, the major problem with the topology construction algorithms remains that there is no generally an agreed mechanism for choosing the optimal location of the sink node [10]. Consequently, these algorithms locate the sink node in the center of the deployment area in all conditions. In fact, the central position in the ROI made by most of the topology construction algorithms is not effective position to receive the detected meaningful events from the sensor nodes and relieve pressure on the sensor nodes around it. This central position of the sink node is determined with the help of P-Median Problem (PMP) model, which has been proved to present the distinctive property of non-deterministic polynomial-time hard [11], [12]. Usually, the center of an LSWSN's coverage area is considered as the optimal position; be it geometric region where the sensor nodes are uniformly deployed, or rectangular or circular area.

However, it is discovered that the proposition discussed earlier is made without taking into consideration the hot spot problem and the regional barrier. These days, there are many types of LSWSNs deployed in different application fields but the most common one is called flat networks. So, if the sink node is placed in the optimal position, not on center of the deployment area, the lifetime of the whole network may be extended without increasing other costs [4]. Because the distance between the terminal sensor nodes and the sink node is one of the most important factors affecting energy consumption. For instance, if the distance between the terminal sensor nodes and the sink node is large, this will consume more energy in delivering the collected data to the sink node, consequently the energy consumption rates will surge and the network life will shorten. Choosing the optimal position of a sink node in LSWSNs is based on a set of criteria: (a) The number of neighbors around the sink node, (b) The residual energy of the sink node's neighbors, (c) The residual energy of the sink node itself, and lastly, (d) The distance between the sink node and center of the deployment area [13].

Nevertheless, the administrator of a network has no control over the position of the sink node, as changing its position to alleviate the energy consumption rates is a cumbersome exercise [9]. To address the problem of positioning the sink node, researchers have formulated this problem as a matter of optimization to be solved by metaheuristic algorithms [14]. Because, these algorithms have shown enormous success in solving hard optimization problems in various domains [15], [16]. Different popular algorithms like particle swarm optimization (PSO) [17] and ant colony optimization (ACO) [18] have been utilized in this area of research. However, there is still noticed an enormous gap where the potential of the latest and efficient metaheuristic algorithms on the LSWSNs is yet to be explored. Therefore, in this paper, a recent addition to metaheuristic algorithms Harris' hawks optimization (HHO) algorithm [19] is employed to achieve an optimal position of the sink node in LSWSNs considering both the single and multiple sink nodes placement scenarios. The Prim's greedy algorithm [20] is applied to find the minimum spanning tree (MST) [21] by creating the shortest transmission paths from the sensor nodes to the sink node. Ultimately, the performance of HHO is compared with several well-known metaheuristic algorithms of the literature, such as PSO [22], flower pollination algorithm (FPA) [23], grey wolf optimizer (GWO) [24], sine cosine algorithm (SCA) [25], multi-verse optimizer (MVO) [26] algorithm, and whale optimization algorithm (WOA) [27], in terms of energy consumption, localization error, and statistical evaluation criteria. In summary, the main contributions of this paper are as follows:

- Introducing an approach for determining the optimal sink node position and increasing the lifetime of LSWSN;
- Applying recent meta-heuristic algorithm called Harris' hawk optimization (HHO) algorithm to find the optimal placement of the single and multiple sink nodes in LSWSNs;
- Using a combination of different large-scale network sizes of up to 5000 sensor nodes;
- Comparing the employed algorithm with different meta-heuristic algorithms based on the convergence curves and statistical measures (e.g., mean, best, worst, and standard deviation);
- Proposing a new fitness function for determining the optimal position of the sink node in LSWSNs.

The remainder of this paper is organized as follows: in Section II, a review of previous studies related to optimization algorithms employed on WSNs is presented. Section III describes the mathematical model and the pseudo code of the employed HHO algorithm. The experimental results of HHO on LSWSNs with different network sizes are reported and analyzed in Section IV. Finally, the study is duly concluded in Section VI where potential future research is also highlighted.

II. LITERATURE REVIEW

Node placement in WSNs presents hard optimization problem, which is further complicated with the demand for energy minimization and the network lifetime maximization [3]. To address this, significant research has been performed on the sensor node localization in the existing literature. Here, some of the important previous studies, related to determining the optimal location of the sink node and maximizing WSNs lifetime using various optimization algorithms, are discussed.

Sensor localization is considered as NP hard optimization problem, therefore several metaheuristic techniques have been employed on solving this problem [28]. From recent studies, it can be suggested that population-based metaheuristic algorithms have shown tremendous success in this domain. These algorithms, as opposed to deterministic optimization methods, perform collective intelligence to generate an optimal solution with limited time and computational cost. In [4], the problem of determining the sink node placement is solved by proposing a method based on the cat swarm optimization (CSO) algorithm [29]. Compared with PSO, authors contend that the proposed approach proved the efficacy by extending the network lifetime. Efficient construction of minimum transmission paths from the sink node to the rest of the sensor nodes using the greedy algorithm made a significant contribution to saving the energy consumption of the process of sending and receiving the collected data. Another implementation of PSO in this domain can be found in [13]. The researchers proposed an energy-aware topology control protocol by a mechanism to choose the best position of the sink node in the whole network to prolong its life. To validate the ability of the proposed solution in reducing the energy consumption of the sensor nodes, it was compared with other topology construction protocols. The simulation results reveal the superiority of this method in topology construction and maintenance phases of the topology control protocols in terms of the operational network's lifetime, the number of topology reconstructions, and the number of active nodes.

Banka and Jana in [17] have also employed PSO to best place sink node in WSNs, and the results indicate the superiority of their approach compared to the exhaustive grid search algorithm. In [30], a multi-objective PSO is used to solve different optimization problems related to WSNs and their general application in various fields to find the best position in WSNs with fixed nodes. The authors have focused on finding the optimal sink node position with respect to relay nodes in order to prolong the network lifetime. An adaptive PSO (APSO) is proposed in [31] for optimum placement of sink node in WSNs. According to the findings, APSO outperformed PSO in achieving a prolonged network lifetime for a substantial operation time. The genetic algorithm (GA) [32] as classic metaheuristic algorithm has been implemented in [33] for optimal sink node placement. With different mutation and crossover settings, the research contends to achieve optimum sink location in short generations.

Another state-of-the-art metaheuristic technique, called ant colony optimization (ACO) [34], has been proposed for the sink node placement in [18]. In this research, the authors have employed ACO on finding the optimum transmission path with a strategy to enhance the single-sink WSN lifetime. Compared to the energy-oriented approach, this study contends to have achieved better results. Another application of ACO for maximizing the lifetime of heterogeneous WSNs is found in [35]. The authors have proposed an approach based on finding the maximum sensor network coverage by establishing an optimal path on the construction graph. Similarly, research in [36] also proposes an ACO-based technique for developing an energy-efficient solution for WSNs lifetime maximization and packet loss minimization. In [37], the authors have incorporated fuzzy logic in ACO-based approach for developing a rule-base for route classification for the enhanced energy efficiency of the network. Likewise, the findings of [38] also support the efficacy of ACO in solving the optimization problems in WSNs. Apart from ACO, a different well established method, cuckoo search (CS) [39], has been employed in [40] for sensor node localization in WSNs. In this study, CS algorithm is modified by a mutation strategy for the enhanced global search ability. The method, based on extensive experimental analysis, proves that it can effectively increase the coverage of the sensor nodes, along with the reduced localization error.

A recent swarm intelligence-based metaheuristic algorithm is implemented in [9] considering the optimal sink node position localization problem in LSWSNs. The authors have used a multi-objective WOA for solving the problem of choosing the lowest number of sink nodes that can feed the whole network in LSWSNs, for reduced energy consumption and extended network life. The algorithm, based on a certain number of experiments, shows higher performance in achieving the aforementioned aims than other well-known algorithms, such as multi-objective grasshopper optimization algorithm (GOA) [41], multi-objective salp swarm algorithm (SSA) [42], multi-objective GWO, and multi-objective PSO over different network sizes. A binary version of WOA has been proposed in [10] for dealing with discrete data in WSNs to determine active nodes and inactive node represented as 1 and 0, accordingly. The method ensures that each active node has coverage for rest of the nodes, using breadth-first search to overcome coverage problem of network. Another efficient swarm-based metaheuristic algorithms, namely brainstorm optimization (BSO) algorithm [43], has been employed in [3] for optimal deployment of the sink nodes in WSNs. The results, when compared with PSO and grid search-based approaches, suggest that BSO achieved energy efficient sink node placement with a prolonged network lifetime.

In [44], the authors have used GWO for treating the problem of central sink node position of the topology construction algorithms. In addition, the established approach is compared with the topology control protocols to evaluate its performance qualitatively and quantitatively. Based on a number of experiments for different network sizes, in many deployment scenarios, efficiency of the proposed approach is appeared to be efficient solution in terms of the energy cost and the number of active nodes, along with the time required to construct a reduced topology. Contrary to the earlier research, the study performed in [45] finds GWO generating inferior solution as compared to chicken swarm optimization (CSO) [46], when establishing the lowest active nodes for the WSN operation. The empirical results proved that CSO outperforms GWO with respect to showing the ability to achieve reduced set of active nodes with high residual energies.

Several other techniques have also been reported in the related literature. In [47], the authors have constructed optimal clustering architecture and designed energy-aware cluster-head rotation, as well as, a routing protocol to maximize the network lifetime. Similarly, an efficient approach

TABLE 1. Metaheuristic methods used in literature for optimum sink placement problem.

Methods	Network size (Nodes)	ROI
APSO [31]	100-900	600 m^2
BSO, PSO [3]	300	200 m^2
PSO [17]	300	200 m^2
PSO [13]	100-700	600 m^2
GA [33]	17	951 km^2
AFSO [50]	100, 200	100 m^2
ACO [18]	100-370	100-600 m^2
CSO, PSO [4]	100-600	200 m^2
PSO [30]	676	1000 m^2

to allocating M sink nodes in a 2D dimensional space is proposed in [48], whereby the recruiting of PSO has developed a deactivation scheme by storing copies of the location vectors from particles with better valuation results for controlling the extravagant conversion of a particle. On the other hand, works in [11] and [12] have proposed the P-Median Problem (PMP) model to determine the sink node placement. Also, in [12], the authors have proved that the center of the circle is the optimal position for a base station in WSNs, but the conclusion is only suitable for the uniform deployment of nodes. The sink node position is chosen to maximize the weight of data flows to reduce the energy consumption is introduced in [49].

The research reviewed in the existing literature suggests that the sink node placement problem has been solved with small to medium-sized networks in small to medium ROI. Table 1 reveals that this problem has been solved in WSNs with maximum of 900 nodes, except for the study in [9], which extends network size up to 10000 nodes. Moreover, there is a significant gap in this area of research where the latest and more efficient optimization techniques are yet to be explored, with extensive comparative analysis. Motivated by the potential of the metaheuristic techniques for solving a variety of WSNs problems, this study employs the most recent optimization technique, Harris' hawk optimization [19] on LSWSNs. In previous researches pertaining to different areas, this method has already generated promising results while solving optimization problems of different levels of difficulties. Following gives a brief introduction of the method, while for more details, the original work can be referred.

III. HARRIS HAWKS OPTIMIZATION (HHO)

Harris hawks optimization (HHO) [19] is a new nature-inspired algorithm. The basic inspiration lies with HHO is the hunting behavior of Harris' hawks, also known as dusky hawks. These birds perch in air, distantly locate the prey, and then pounce on it in a collaborative effort. The perching behavior of the hawks is modeled as an exploration phase, whereas their pouncing style is simulated as exploitation phase in HHO. The mathematical model of the HHO algorithm is explained in this section. Note that a candidate solution in HHO is termed as a hawk (x), whereas the best solution as a prey (x_{prey}).

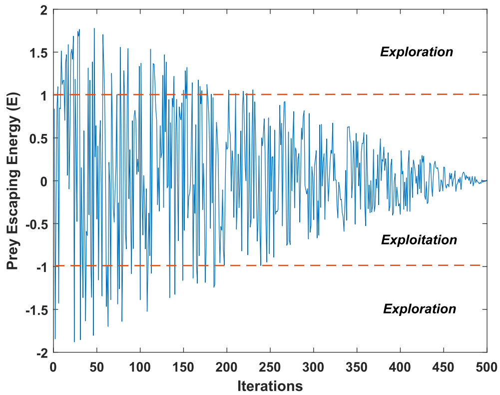


FIGURE 3. Exploration-exploitation phases of HHO depending on prey escaping energy (*E*).

A. EXPLORATION IN HHO

In optimization methods, in order to find the best from largely available solutions, a rigorous search of the problem landscape is performed. To discover optimal location amid hills and valleys in a search space, a metaheuristic starts the search with an exploration phase. During this phase, a thorough search is performed in the far reached locations. Here, in population-based metaheuristic algorithms like HHO, search agents spread widely in search space. The HHO algorithm performs this phase following the perching behavior of hawks, which first explore the area from high altitude, to locate a potential prey that can be either a rabbit, small mammal, or a big insect. Keeping in view the mentioned scenario, HHO starts by initializing N search agents (hawks) at random locations x_i^0, $i = \{1, 2, \ldots, N\}$ across search space using:

$$x_i^0 = lb_i + r_1 \times (ub_i - lb_i), r_1 = rand(), \tag{1}$$

where lb and ub are the problem bounds, and $rand()$ is a random number generator, which generates a different random value between [0,1] every time used. After the initialization of the population, the exploration phase continues until the escaping energy of prey $|E| \geq 1$, the value of E is calculated as:

$$E = 2E_0(1 - \frac{t}{T}), \quad t = \{1, 2, \ldots, T\}, \tag{2}$$

where E_0 and T are the initial energy of the prey and a maximum number of iterations, respectively. As depicted in Fig. 3, HHO performs exploration during initial iterations when $|E| \geq 1$. During this phase, a search agent performs search randomly around different other search agents or around potential optimal search regions identified as x_{prey}. This phenomenon is controlled by a random variable q using:

$$x_{new} = \begin{cases} x_{rand} - r_2 \left| x_{rand} - 2r_2 x_i \right|, & q \geq 0.5 \\ (x_{prey} - x_m) - r_3[lb_i + r_4(ub_i - lb_i)], & q < 0.5, \end{cases}$$
$$r_2 = rand(),\ r_3 = rand(),\ r_4 = rand(),\ r_5 = rand(),$$
$$x_m = \frac{1}{N}\sum_{i=1}^{N} x_i, \tag{3}$$

where x_{new}, x_m, and x_{rand} are new position, dimension-wise average of the population, and a randomly selection position, respectively.

B. EXPLOITATION IN HHO

Exploitation is when the population of candidate solutions converges towards the already identified promising location in search space. This phase activates after several iterations performed for exploration of the problem landscape. Once, determined a potential neighborhood by the collective experience of search agents, an exploitation strategy gradually makes the candidate solutions adopt information from a single global best solution found so far. Using this information, the solution improvement is performed. Though, it is important to mention that the early approach to the local region may cause premature convergence, which will result in a suboptimal solution. To address this, HHO adopts multiple exploitation strategies using different hunting situations for a hawk. For example, when approaching a prey, the hawk may either make a sudden dive for an immediate attack, or it may decide to wait for a surprise pounce as the prey is trying to escape.

The exploitation strategies implemented by HHO are: hard besiege, hard besiege with progressive rapid dives, soft besiege, and soft besiege with progressive rapid dives. Hard besiege implies that the prey is exhausted or very well positioned for a hawk to perform a successful catch. But, when the hawk perceives the situation not suitable enough for attempting the catch, it tries to get closer to the prey as soon as possible; hence performing hard besiege with progressive rapid dives. On the other hand, when a hawk has located a prey but from distance, and the prey has the energy to escape the hunt, the hawk tries to encircle it to make it exhausted. This situation is termed as soft besiege in HHO. The soft besiege with progressive rapid dives implies a situation when the hawk is trying to catch a prey by making progressive dives but the prey is energetically making random zigzag moves to avoid the catch.

To implement progressive movement towards the already identified promising local region, HHO uses Lévy flight $LF(D)$ function. The $LF(D)$ is used in both the soft and hard besieges with progressive rapid dives. In soft besiege, HHO uses jump strength J, which represents escaping effort by the prey, in order to inject some randomization for improved search results in the local region. Table 2, along with relevant mathematical expressions, lists the conditions for performing the four exploitation steps.

- *Hard besiege:*

$$x_{new} = x_{prey} - E \left| \Delta x_i \right|, \text{where } \Delta x_i = x_{prey} - x_i \tag{4}$$

- *Hard besiege with progressive rapid dives:*

$$x_{new} = \begin{cases} Y, & if\ F(Y) < F(x_i) \\ Z, & if\ F(Z) < F(x_i), \end{cases} \text{ where}$$
$$F(x_i) = \text{Fitness value of } x_i,$$

TABLE 2. HHO exploitation strategies.

Strategy	Condition (E=escaping energy r_6=random number between [0,1])	Formula
Hard besiege	$\|E\| < 0.5$ AND $r_6 \geq 0.5$	Eq. (4)
Hard besiege with progressive rapid dives	$\|E\| < 0.5$ AND $r_6 < 0.5$	Eq. (5)
Soft besiege	$\|E\| \geq 0.5$ AND $r_6 \geq 0.5$	Eq. (6)
Soft besiege with progressive rapid dives	$\|E\| \geq 0.5$ AND $r_6 < 0.5$	Eq. (7)

$$\begin{aligned}
Y &= x_{prey} - E \left|Jx_{prey} - x_m\right|, \\
J &= 2(1 - r_6),\, r_6 = rand(), \\
Z &= Y + S \times LF(D), \text{ where} \\
& S = \text{Random vector of size} 1 \times D,\, D = \text{Dimensions} \\
LF(D) &= 0.01 \times \frac{u \times \sigma}{|v|^{\frac{1}{\beta}}}, \\
\sigma &= \left(\frac{\Gamma(1+\beta) \times \sin(\frac{\pi\beta}{2})}{\Gamma(\frac{1+\beta}{2}) \times \beta \times 2^{(\frac{\beta-1}{2})})} \right)^{\frac{1}{\beta}}, \\
\beta &= 1.5,\, u = rand(),\, v = rand().
\end{aligned} \tag{5}$$

- *Soft besiege:*

$$\begin{aligned}
x_{new} &= \Delta x_i - E \left|JX_{prey} - x_i\right|, \text{ where} \\
\Delta x_i &= x_{prey} - x_i,\, J = 2(1 - r_7),\, r_7 = rand()
\end{aligned} \tag{6}$$

- *Soft besiege with progressive rapid dives:*

$$\begin{aligned}
x_{new} &= \begin{cases} Y, & if F(Y) < F(x_i) \\ Z, & if F(Z) < F(x_i), \end{cases} \text{ where} \\
Y &= x_{prey} - E \left|Jx_{prey} - x_i\right| \\
Z &= Y + S \times LF(D), \\
S &= \text{Random vector} 1 \times D,\, D = \text{Dimensions}
\end{aligned} \tag{7}$$

A step by step procedure for HHO is outlined in Algorithm 1.

Algorithm 1 Steps of HHO Algorithm

```
Input: A graph represents the sensor nodes and their neighbors with energies
Output: The final network with the optimal location of the sink node(s)
Initialize the random population x_i^0 (i = 1, 2, ..., N)
Initialize start of iteration t = 1 and T = maximum iterations
while t <= T do
    Calculate the fitness values of hawks (Sensor nodes)
    Set x_prey as the location of prey (best location)
    for (each hawk (x_i)) do
        Update the initial energy E_0 and jump strength J
        Update the E using Eq. (2)
        if (|E| ≥ 1) then                                   ▷ Exploration
            Update solution using Eq. (2)
        end if
        if (|E| < 1) then                                   ▷ Exploitation
            if (r ≥0.5 and |E| ≥ 0.5 ) then                 ▷ Soft besiege
                Update solution using Eq. (6)
            else if (r ≥0.5 and |E| < 0.5 ) then            ▷ Hard besiege
                Update solution using Eq. (4)
            else if (r <0.5 and |E| ≥ 0.5 ) then            ▷ Soft besiege with progressive rapid dives
                Update solution using Eq. (7)
            else if (r <0.5 and |E| < 0.5 ) then            ▷ Hard besiege with progressive rapid dives
                Update solution using Eq. (5)
            end if
        end if
    end for
    t = t + 1
end while
Return X_prey (Optimal position of the sink node)
```

IV. SYSTEMS MODEL

This section discusses the implementation of HHO and other metaheuristic algorithms on the sink node placement problem in LSWSNs. For evaluating the performance of HHO, it is compared with other well-known algorithms. We implemented these algorithms in MATLAB. Additionally, we used the Atarraya simulator [51] for generating the network graphs (datasets) for different network sizes, including sensor coordinates and the residual energies inside each sensor node in the deployment area. For validation of the technique, we compare HHO with six well-known algorithms, namely, PSO, FPA, GWO, SCA, MVO, and WOA.

A. NETWORK MODEL

In this work, a network is considered with a huge collection of sensing nodes having the same sensing and communication range. Two different scenarios are considered for sink placement, single sink node and multiple sink nodes. These nodes are distributed randomly in a convex region of interest $ROI = W \times H$, where W is the width of the region and H is its height. These nodes have limited facilities of wireless sensors such as limited memory, bandwidth, and uniform energy. The sink node has bandwidth and memory. The sensing nodes in the network have the ability to sense and distribute physical and environmental conditions between each other for sending to the sink node. This paper considers the network $N = \{n_1, \ldots, n_k\}$ where n is the sensing node and k is the number of nodes. These nodes are distributed randomly in ROI and there is no node, which has any global information. The time when the first node fails or loses its energy in the network is called the network lifetime.

B. PARAMETERS SETTING

Table 3 and Table 4 list the initial parameters for the experimental scenarios that were adjusted in LSWSNs and optimization algorithms. Crossbows Mica Mote sensors are

TABLE 3. Parameters setting for LSWSNs experiments.

Parameters	Values
Deployment area	600 m^2-1000 m^2
Number of nodes	100,...,500-1000,...,5000
Routing protocols	LEACH
Sensor node model	Mica Mote
Node communication (R_c)	Range 100 m
Node sensing (R_s)	Range 20 m
Node placement	Uniform
Node energy	Uniform
Max energy	2000 (mA-h)
Fitness function probability α_1	0.4
Fitness function probability α_2	0.1
Fitness function probability α_3	0.5

TABLE 4. Parameter settings of HHO and the competitor algorithms.

Algorithms	Parameters
PSO	Particles number = 100,..., 5000 Inertia coefficient = 0.75 Cognitive and social Coeff. = 1.8, 2 Number of generations = 1000
FPA	Population size = 100,..., 5000 Probability switch (p) = 0.8 Number of generations = 1000
GWO	Wolves number = 100,..., 5000 Control parameter (a) = [2, 0] Number of generations = 1000
SCA	Search agents = 100,..., 5000 Number of Elites = 2 Number of generations = 1000
MVO	Search agents = 100,..., 5000 Wormhole existence prob. = [0.2, 1] Traveling distance rate = [0.6, 1] Number of generations = 1000
WOA	Whales number = 100,..., 5000 a decreases linearly from 2 to 0 (Default) $a2$ decreases linearly from -1 to -2 (Default) Number of generations = 1000
HHO	Harris' Hawks number = 100,..., 5000 β = 1.5 E_0 changes from -1 to 1 (Default) Number of generations = 1000

utilized to assume the nodes within simulation with the energy model defined in [6]. Note that for all experiments, we have utilized 100 to 5000 search agents and a maximum of 1000 iterations, and ran algorithms for 30 times to obtain mean, standard deviation (Std.), best, and worst results.

C. FITNESS FUNCTION

To evaluate the performance of the algorithms and choose the best one, the energy consumption rates, the optimal sink node position, and the localization errors are considered. A new fitness function is proposed to determine the optimal placement of the sink node in LSWSNs, and it is utilized by these algorithms. Eq. (8) defines the proposed fitness function.

$$\min f(x) = \frac{1}{\alpha_1 \sum_{i=1}^{N_{nbr}} E_{nbr}(i) + E_x + \alpha_2 N_{nbr} + \alpha_3 d_x}, \tag{8}$$

where N_{nbr} denotes the number of sensor neighbor served by the sink node, E_{nbr} energy for each sensor node for sinks, E_x residual energy inside the sensor node x, d_x the distance between the sensor node x and center of the deployment area, and the parameters α_1, α_2, and α_3 are random numbers in range of [0, 1]. This fitness function is dependent of the position vector x of all sensor nodes.

TABLE 5. Comparison of energy consumption between the competitive algorithms.

Nodes	PSO	FPA	GWO	SCA	MVO	WOA	HHO
100	36911	38537	36446	34920	35820	39222	34101
200	60637	60957	67000	73931	79044	80569	51805
300	88115	87028	83732	100829	113868	123869	60224
400	98697	191788	94931	114198	260659	270659	66005
500	105190	109630	120544	121820	392475	402475	70233
1000	131419	165651	133185	209715	468689	488989	107456
2000	198464	194892	188526	236876	591674	601674	124777
3000	247826	279863	223074	313128	643032	693089	150659
4000	267899	289874	244088	349696	665284	764098	160659
5000	298755	299995	267896	389654	699978	834058	180999

D. STATISTICAL EVALUATION CRITERIA

The following standard metrics are used to measure the performance and validate the algorithms based on the fitness function defined in Eq. (8):

1) Mean is the average of fitness values, which the algorithm produced after M runs, and it is given by:

$$Mean = \frac{\sum_{i=1}^{M} (f_i)}{M} \tag{9}$$

2) Standard deviation (Std.), which performs the difference of the objective function values obtained from executing the algorithm for M times. Small values of standard derivation are an indicator of the ability of the algorithm to converge to the same value most of the times, which shows its robustness and stability. Large values are an indicator that the algorithm produces wandering results. The standard deviation is given by:

$$Std = \sqrt{\frac{1}{M-1} \Sigma_{i=1}^{M} (f_i - Mean)^2} \tag{10}$$

3) Best is the minimum fitness value obtained in M runs. The best fitness value is calculated using:

$$Best = \min_{1 \le i \le M} f_i \tag{11}$$

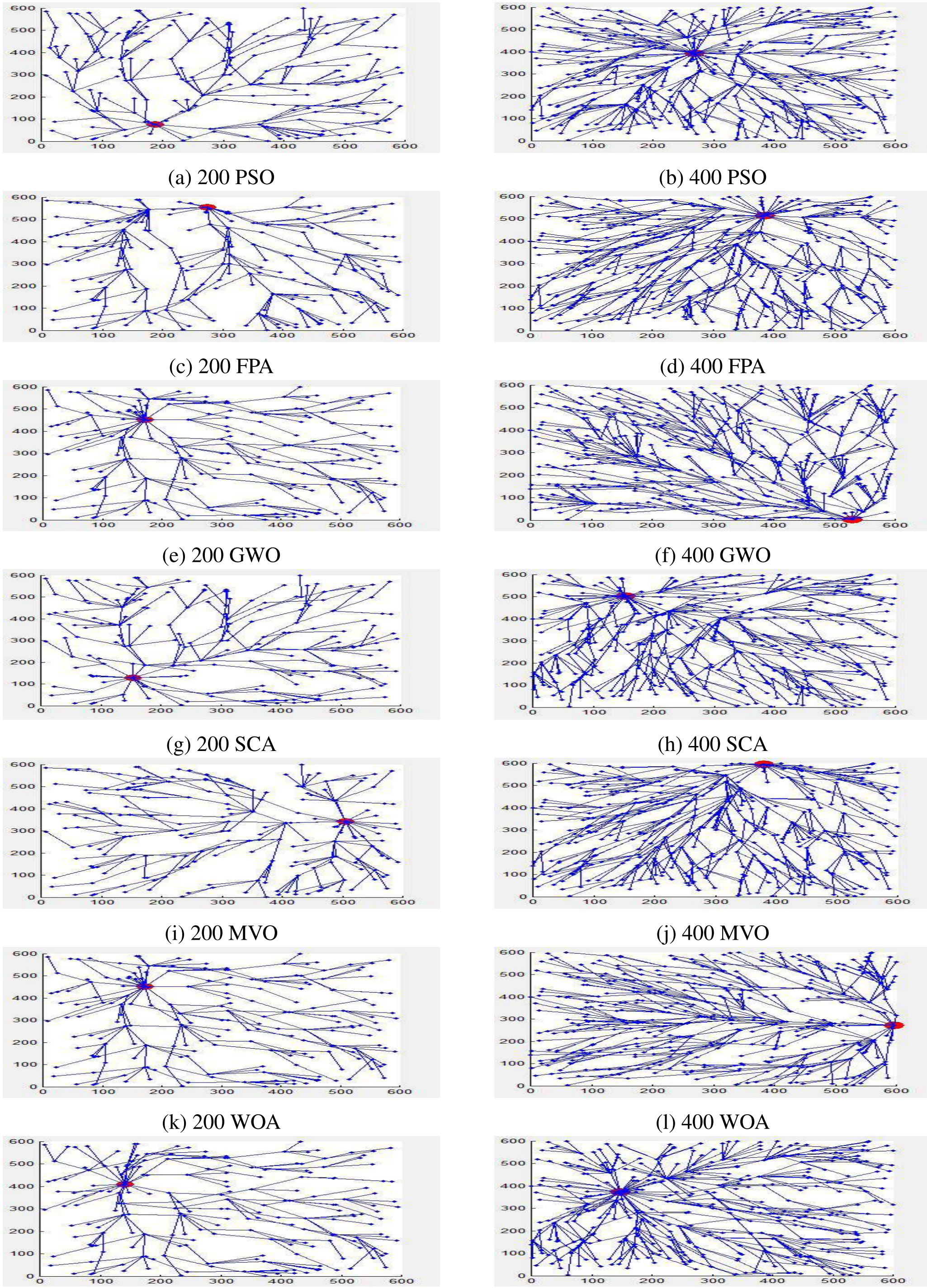
4) Worst is the maximum fitness value obtained in M runs. The worst fitness value is calculated as:

$$Worst = \max_{1 \le i \le M} f_i, \tag{12}$$

where f_i is the best fitness value obtained at ith run.

V. RESULTS AND DISCUSSION

In this section, the results of the single sink node placement including energy consumption, localization errors, convergence curves, statistical results, running time, and their discussions are introduced. Also, the results of multiple sink node placements are introduced sequentially.



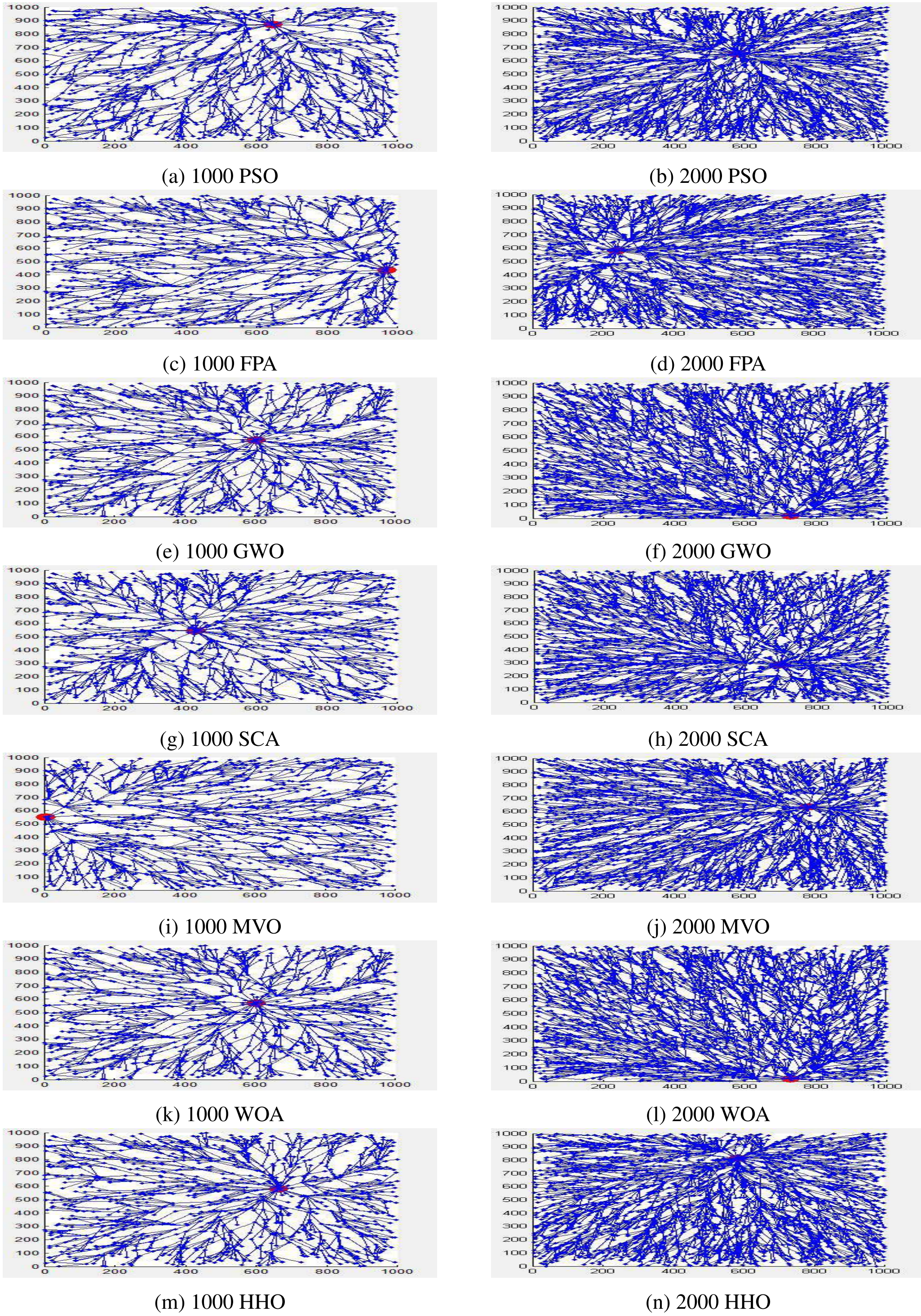
(a) 200 PSO (b) 400 PSO
(c) 200 FPA (d) 400 FPA
(e) 200 GWO (f) 400 GWO
(g) 200 SCA (h) 400 SCA
(i) 200 MVO (j) 400 MVO
(k) 200 WOA (l) 400 WOA
(m) 200 HHO (n) 400 HHO

FIGURE 4. Applying the Prim's algorithm after choosing the optimal position of the sink node in 200 and 400 network size for all algorithms.

A. SINGLE SINK NODE PLACEMENT

1) ENERGY CONSUMPTION

After choosing the optimal location of the sink node, which will extend the network lifetime based on Eq. 8, the Prim's minimum spanning tree (MST) algorithm is applied to reconstruct the network and establish minimum transmission paths from the chosen sink node to the rest of the sensor nodes. Fig. 4 and Fig. 5 demonstrate the location of the sink node for 200, 400, 1000, and 2000 nodes in a network. After determining the position of the sink node in



(a) 1000 PSO (b) 2000 PSO

(c) 1000 FPA (d) 2000 FPA

(e) 1000 GWO (f) 2000 GWO

(g) 1000 SCA (h) 2000 SCA

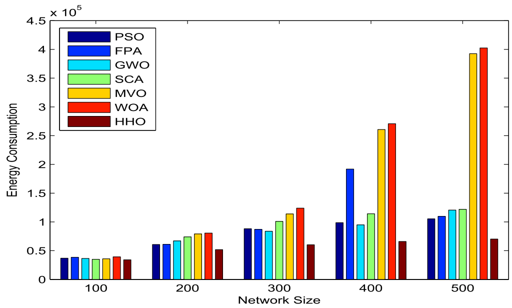
(i) 1000 MVO (j) 2000 MVO

(k) 1000 WOA (l) 2000 WOA

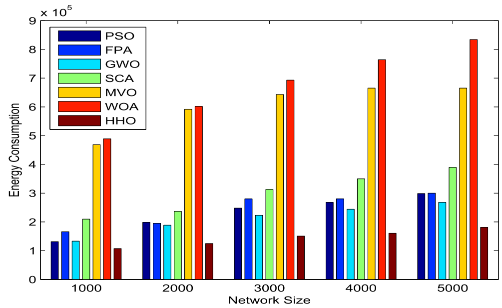
(m) 1000 HHO (n) 2000 HHO

FIGURE 5. Applying the Prim's MST algorithm after choosing the optimal position of the sink node in 1000 and 2000 network size for all algorithms.

a network using HHO with the proposed fitness function, the greedy algorithm with minimum spanning tree is applied to create the data transmission paths. The sink node localization problem is also solved by PSO, FPA, GWO, SCA, MVO, and WOA approaches with the transmission paths built by the greedy algorithm for the comparison purpose. Since the location of the sink node is individually decided by PSO, FPA, GWO, SCA, MVO, WOA, and HHO before applying the greedy algorithm to construct the transmission paths, the whole LSWSNs network with the deployed sink



(a) From 100 To 500 Nodes



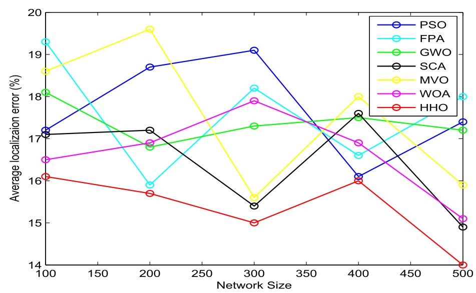
(b) From 1000 To 5000 Nodes

FIGURE 6. Energy consumption by the competitive algorithms for LSWSNs.

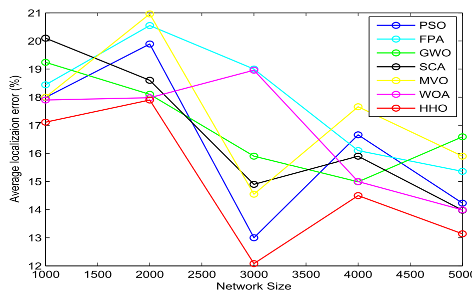
node and the transmitting paths with 200 and 400 nodes, for example, are revealed in Fig. 4 for comparison. Also, Fig. 5 reveals 1000 and 2000 nodes in LSWSNs with the sink node deployment strategy by all the aforementioned approaches. This may be explained by the proposed fitness function of the employed algorithm that enforces to place the sink node within the region that contains the highest number of the sensor nodes.

The main idea of such saving energy for a network, through choosing the nearest best location of the sink node, and reducing transmission path from the sink node to all the sensor nodes in the whole network, a large number of the sensor nodes around the sink node are to be marked as children to that node. Therefore, when the network size is increased, the employed HHO algorithm, in terms of energy consumption, has achieved better results as compared to other algorithms.

Subsequently, the energy consumption test must be applied to the network. The employed algorithm is tested with 10 different numbers of sensor node environments. The results are compared to the power consumption of the sink node allocated by the counterpart algorithms. The experimental results proved that the employed HHO algorithm achieved better efficiency in reducing the total power consumption in the whole LSWSNs. In this sub-experiment, the search agents in algorithms simulate the network sensor nodes' life in consuming the residual energy under a certain number of iterations based on the fitness function. And then, the total energy consumption for each network size is reported. Table 5 shows the total energy consumption for the seven algorithms with respect to each network scenario. The employed HHO algorithm is compared with other well-known algorithms in the area of energy consumption rates. As it is obvious from Table 5 that HHO consumes the lowest energy amount over all the different network sizes from 100 through 5000, among other algorithms used in this sub-experiment. According to Table 5, which reports total power consumption of the LSWSNs with selected algorithms, the total power consumption in the 1000 nodes environment with PSO, FPA, GWO, SCA, MVO, WOA, and HHO approaches



(a) From 100 To 500 Nodes



(b) From 1000 To 5000 Nodes

FIGURE 7. Localization error by the algorithms on LSWSNs.

are 131419, 165651, 133185, 209715, 468689, 488989 and 107456, respectively.

Obviously, the power consumption achieved by the HHO algorithm is less than the ones produced by rest of the algorithms in all the test conditions. The HHO algorithm aims to produce a network with minimal energy consumption to prolong the lifetime of LSWSNs. The experimental result is optimal for deploying the sink node in the LSWSNs in case of HHO. Fig. 6 represents the graphical results of energy consumption rates obtained from the employed HHO algorithm and other algorithms PSO, FPA, GWO, SCA, MVO and WOA in all LSWSNs size. Also, Fig. 6 characterizes the energy consumption achieved by the employed algorithm has been decreased compared to other algorithms.

To enhance the network lifetime, the aim of HHO is to reduce the energy consumption through deploying the sink

TABLE 6. Localization errors for the considered network sizes.

Nodes	PSO	FPA	GWO	SCA	MVO	WOA	HHO
100	17.20%	19.30%	18.10%	17.10%	18.60%	16.50%	16.10%
200	18.70%	15.90%	16.80%	17.20%	19.60%	16.90%	15.70%
300	19.10%	18.20%	17.30%	15.40%	15.60%	17.90%	15.00%
400	16.10%	16.60%	17.50%	17.60%	18.00%	16.90%	16.00%
500	17.40%	18.00%	17.20%	14.90%	15.90%	15.10%	14.00%
1000	17.99%	18.44%	19.24%	20.10%	18.00%	17.90%	17.11%
2000	19.89%	20.55%	18.10%	18.60%	20.96%	17.98%	17.90%
3000	13.00%	19.00%	15.90%	14.90%	14.55%	18.96%	12.08%
4000	16.66%	16.10%	14.99%	15.90%	17.66%	15.00%	14.50%
5000	14.23%	15.36%	16.59%	13.98%	15.90%	13.99%	13.14%

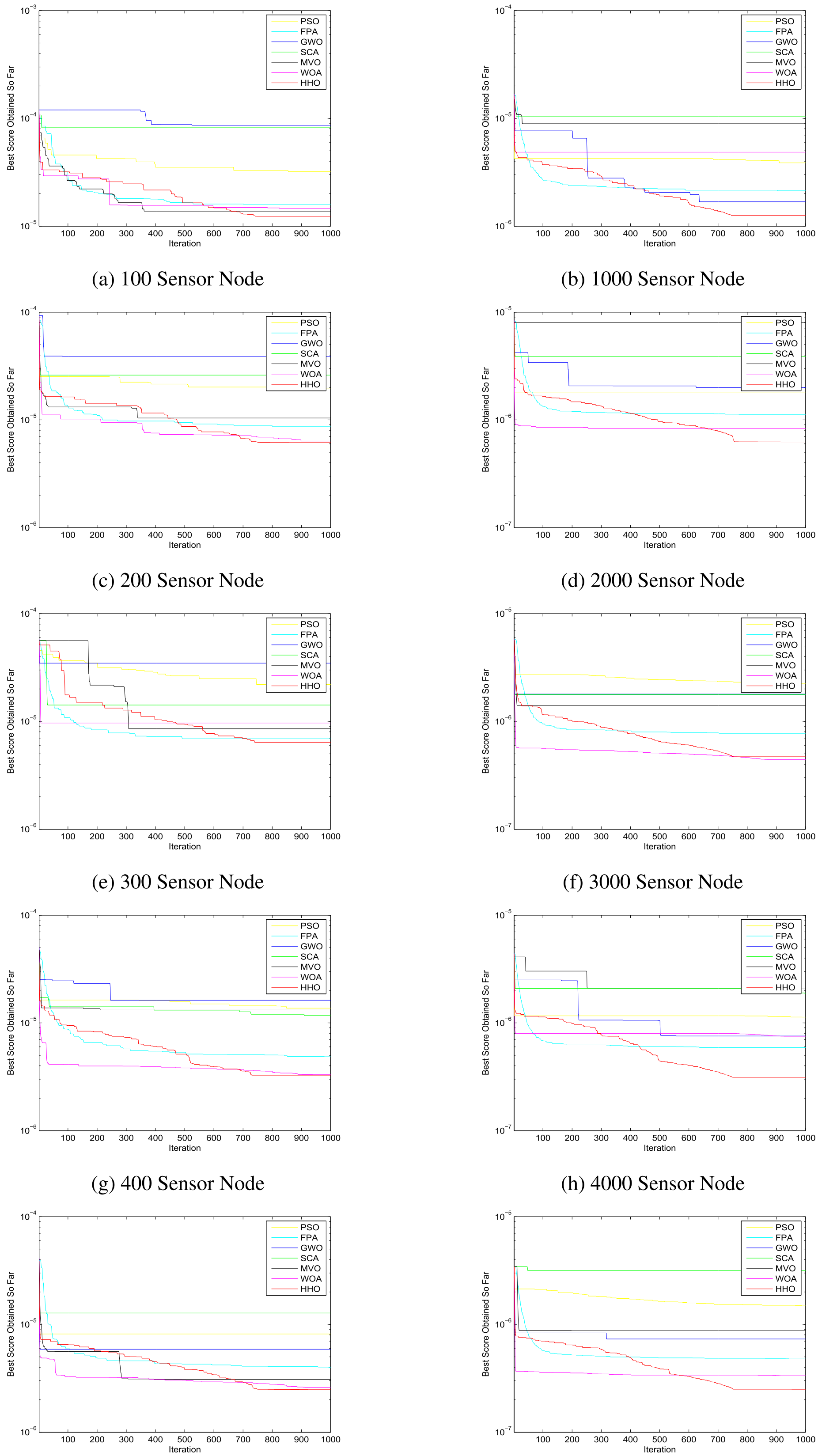
TABLE 7. Statistical results of the competitive algorithms on fitness function for network size 100 through 5000 nodes.

Nodes	EC	PSO	FPA	GWO	SCA	MVO	WOA	HHO
100	Mean	3.68E-05	5.56E-05	8.75E-05	6.29E-05	1.74E-05	2.92E-05	1.23E-05
	Std	1.86E-06	1.04E-05	2.20E-05	2.62E-05	1.12E-05	2.21E-05	5.25E-07
	Best	3.22E-05	3.69E-05	4.20E-05	1.88E-05	1.21E-05	1.26E-05	1.21E-05
	Worst	4.08E-05	7.62E-05	1.17E-04	1.07E-04	6.60E-05	1.08E-04	1.44E-05
200	Mean	2.14E-05	2.62E-05	4.92E-05	2.46E-05	1.05E-05	8.16E-06	6.23E-06
	Std	9.96E-07	6.14E-06	1.91E-05	1.07E-05	1.09E-05	2.85E-06	3.37E-07
	Best	1.85E-05	1.85E-05	1.82E-05	1.07E-05	6.15E-06	6.26E-06	6.12E-06
	Worst	2.32E-05	4.37E-05	8.67E-05	5.08E-05	5.98E-05	2.05E-05	8.00E-06
300	Mean	2.62E-05	2.48E-05	1.97E-05	5.63E-05	1.28E-05	7.19E-06	8.53E-06
	Std	9.95E-07	5.27E-06	9.07E-06	7.08E-07	1.44E-05	2.55E-06	3.18E-06
	Best	2.43E-05	1.53E-05	8.20E-06	5.25E-05	4.13E-06	4.18E-06	4.13E-06
	Worst	2.81E-05	3.60E-05	4.30E-05	5.64E-05	5.33E-05	1.21E-05	1.39E-05
400	Mean	1.49E-05	1.78E-05	2.00E-05	1.70E-05	5.37E-06	3.75E-06	3.46E-06
	Std	5.39E-07	4.66E-06	1.10E-05	7.99E-06	3.17E-06	1.64E-06	3.89E-07
	Best	1.35E-05	1.16E-05	6.80E-06	6.12E-06	3.10E-06	3.14E-06	3.09E-06
	Worst	1.59E-05	2.83E-05	4.39E-05	3.59E-05	1.57E-05	1.18E-05	4.46E-06
500	Mean	7.89E-06	1.21E-05	8.96E-06	1.48E-05	5.75E-06	2.72E-06	2.54E-06
	Std	4.36E-07	3.23E-06	4.49E-06	5.61E-06	4.85E-06	3.85E-07	9.07E-08
	Best	7.03E-06	8.45E-06	2.81E-06	4.97E-06	2.49E-06	2.53E-06	2.48E-06
	Worst	9.01E-06	2.06E-05	2.04E-05	2.54E-05	1.75E-05	4.66E-06	2.80E-06
1000	Mean	4.44E-06	3.79E-06	4.21E-06	6.74E-06	3.58E-06	6.76E-06	2.37E-06
	Std	7.80E-08	1.34E-07	1.27E-06	4.31E-21	2.08E-06	4.31E-21	9.24E-07
	Best	4.24E-06	3.56E-06	1.81E-06	6.74E-06	1.26E-06	6.76E-06	1.25E-06
	Worst	4.63E-06	4.03E-06	5.71E-06	6.74E-06	6.56E-06	6.76E-06	4.65E-06
2000	Mean	1.95E-06	8.24E-06	1.27E-06	7.16E-06	1.10E-06	7.15E-07	7.01E-07
	Std	6.66E-08	0.00E+00	4.20E-07	5.16E-07	7.07E-07	1.68E-08	3.00E-08
	Best	1.89E-06	8.24E-06	7.59E-07	6.32E-06	6.71E-07	6.95E-07	6.55E-07
	Worst	2.06E-06	8.24E-06	1.92E-06	7.54E-06	2.36E-06	7.30E-07	7.31E-07
3000	Mean	2.48E-06	5.74E-06	2.04E-06	4.38E-06	9.63E-07	1.08E-06	9.35E-07
	Std	6.75E-08	0.00E+00	7.63E-07	4.68E-07	5.45E-07	1.65E-07	7.87E-08
	Best	2.39E-06	5.74E-06	1.31E-06	3.81E-06	8.54E-07	8.89E-07	8.21E-07
	Worst	2.57E-06	5.74E-06	3.24E-06	4.87E-06	1.81E-06	1.28E-06	1.01E-06
4000	Mean	1.26E-06	4.38E-06	8.10E-07	3.75E-06	4.80E-07	1.26E-06	3.46E-07
	Std	2.15E-08	5.40E-09	2.84E-07	9.45E-08	4.57E-08	2.10E-07	3.07E-08
	Best	1.24E-06	4.37E-06	5.81E-07	3.67E-06	4.50E-07	1.03E-06	3.27E-07
	Worst	1.28E-06	4.38E-06	1.13E-06	3.85E-06	5.32E-07	1.44E-06	3.82E-07
5000	Mean	1.98E-06	3.45E-06	9.66E-07	3.22E-06	5.61E-07	8.88E-07	5.48E-07
	Std	4.19E-08	0.00E+00	2.24E-07	1.41E-07	1.66E-07	2.24E-07	1.81E-07
	Best	1.94E-06	3.45E-06	7.35E-07	3.13E-06	3.77E-07	6.39E-07	3.34E-07
	Worst	2.02E-06	3.45E-06	1.18E-06	3.38E-06	7.00E-07	1.07E-06	7.51E-07

node on the best location. From the empirical results, it can be suggested that HHO can still provide the network lifetime that is slightly higher than PSO, FPA, GWO, SCA, MVO, and WOA algorithms. Based on results of the sink node location and energy consumption for the network, it can be stated that when the number of nodes is increased, HHO finds more optimal sink node location than the counterparts. This result may be attributed to the process of HHO algorithm of determining the moving directions and distances of the nodes by building the transmission path from the sink node to other nodes in the network, using transmission path construction by greedy algorithms such as MST.

2) LOCALIZATION ERRORS

The localization error (LocErr) is generally related to the node communication radius R_c. The LocErr is usually used to evaluate the localization performance of the employed algorithm and it can be obtained from Eq. (13). It is used



(a) 100 Sensor Node
(b) 1000 Sensor Node
(c) 200 Sensor Node
(d) 2000 Sensor Node
(e) 300 Sensor Node
(f) 3000 Sensor Node
(g) 400 Sensor Node
(h) 4000 Sensor Node
(i) 500 Sensor Node
(j) 5000 Sensor Node

FIGURE 8. **Convergence of the competitive algorithms over 100-5000 network sizes.**

TABLE 8. Average evaluation criteria for the competitive algorithms.

AEC	PSO	FPA	GWO	SCA	MVO	WOA	HHO
Mean	1.19E-05	1.62E-05	1.94E-05	2.00E-05	5.85E-06	6.17E-06	3.79E-06
Best	1.07E-05	1.16E-05	8.32E-06	1.16E-05	3.15E-06	3.87E-06	2.91E-06
Worst	1.29E-05	2.30E-05	3.24E-05	3.01E-05	2.24E-05	1.68E-05	5.10E-06

TABLE 9. Running time of the algorithms in seconds.

Nodes	PSO	FPA	GWO	SCA	MVO	WOA	HHO
100	239.04	243.05	81.17	84.22	138.51	80.84	79.15
200	671.42	548.67	275.73	275.75	332.00	277.50	271.11
300	1330.27	1190.13	600.07	590.91	660.07	590.70	550.91
400	2130.81	2120.82	1050.23	1030.73	1120.37	1040.53	1000.99
500	3330.82	3230.48	1630.24	1610.89	1670.84	1610.30	1560.87
1000	12880.65	12860.67	6400.65	6400.69	6410.03	6430.15	6100.32
2000	52800.99	51550.75	25630.24	25600.57	25470.79	25720.56	25291.36
3000	116340.06	116340.68	58940.97	58130.05	58220.91	57580.49	56830.82
4000	208450.58	208650.75	102760.31	103110.37	103220.90	103000.24	102230.73
5000	326070.31	323330.62	160450.65	166680.06	201780.75	174400.41	140326.88

TABLE 10. Number of sink nodes obtained by HHO and the other algorithms.

Nodes	PSO	FPA	GWO	SCA	MVO	WOA	HHO
100	12	11	15	14	11	11	10
200	9	8	11	10	10	12	7
300	9	8	10	9	8	10	7
400	7	6	7	6	8	9	5
500	8	7	8	7	9	7	6
1000	12	10	10	13	11	9	8
2000	13	11	10	11	12	11	10
3000	12	9	11	9	11	12	8
4000	14	10	11	11	12	11	9
5000	11	11	13	10	14	13	10

in this study to evaluate the performance of each node being estimated, which is further applied to evaluate performance of all nodes being estimated.

$$\mathrm{LocErr} = \frac{\sqrt{(x' - x)^2 + (y' - y)^2}}{R_c} \times 100\%, \tag{13}$$

where (x', y') are the sink node estimated coordinates, (x, y) are the sink node real coordinates, and R_c refers to the node communication range. One of the most important goals of the employed algorithms is to reduce the localization error of the estimated sink node coordinates. Table 6 depicts the localization error for the proposed HHO algorithm according to the effect of changing the number of network sizes. Also, Fig. 7 depicts the graphical analysis according to the localization error values obtained by each algorithm with respect to all the considered network sizes. As it can be observed from Table 6 and Fig. 7 that the employed algorithm has in general small localization errors with respect to the network sizes.

3) CONVERGENCE

Convergence curve is one of the most important graphical analysis with respect to generating an optimal solution via optimization algorithms. To evaluate performance of the considered algorithms in applying and minimizing the fitness function for achieving the lowest energy consumption rates, the convergence curve is extracted from each of the employed algorithm after 1000 iterations, to clearly notice and analyze the convergence ability of all the algorithms. Line graphs of the convergence curves for all the considered network size,

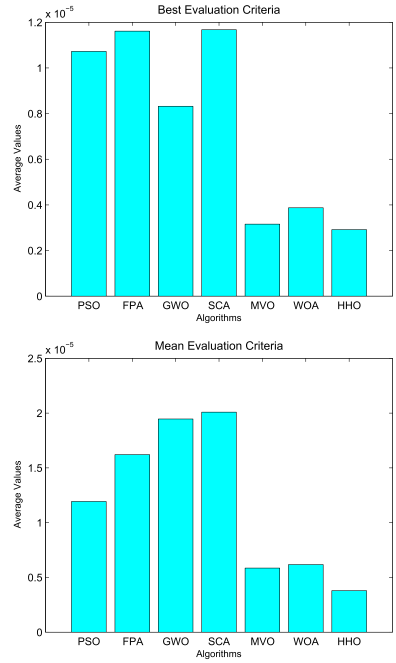
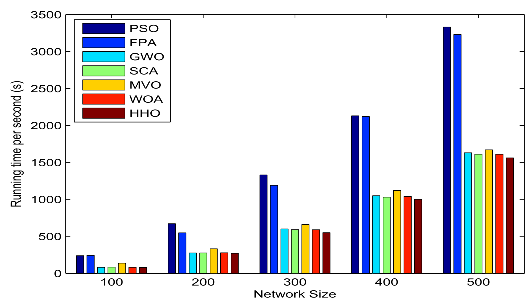
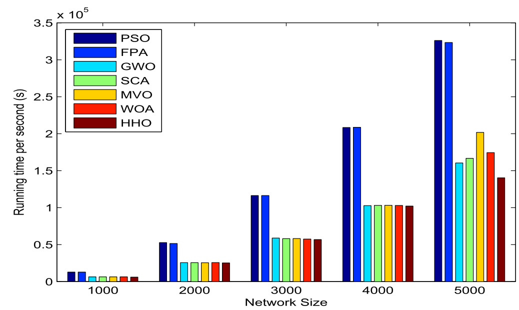


FIGURE 9. Best and mean fitness values achieved by the competitive algorithms.



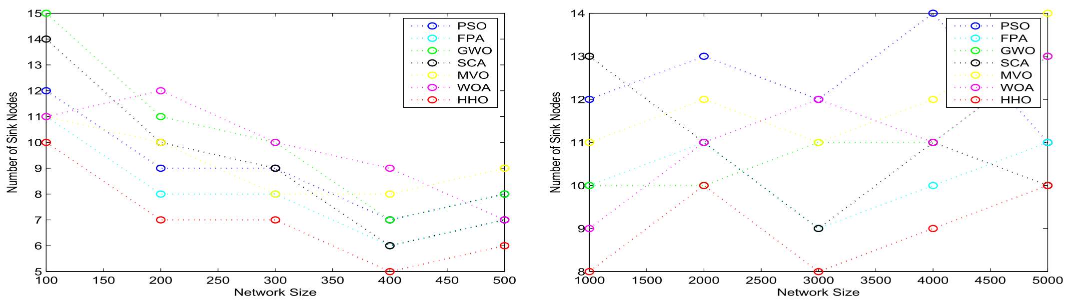
(a) From 100 To 500 Nodes



(b) From 1000 To 5000 Nodes

FIGURE 10. Run time for topology construction by the competitive algorithms.

using HHO algorithm and the comparative algorithms, are shown in Fig. 8. It is easily understandable from Fig. 8 that the HHO algorithm is the fastest one of all in the context of convergence towards the optima. But, the ten curves in Fig. 8 are not the same, for example, Fig. 8c and Fig. 8e differ from Fig. 8a, Fig. 8g, and Fig. 8i. This is because of the different



(a) From 100 To 500 Nodes

(b) From 1000 To 5000 Nodes

FIGURE 11. Number of sink nodes for compared algorithms.

network sizes in each curve. When the network size differs, the parameters in the fitness function such as the number of sensor neighbors served by sink node, residual energy inside the sensor nodes, and the distance between the sensor nodes will differ and consequently, the value of the fitness function (best score obtained so far) will differ according to each algorithm.

4) RESULTS DISCUSSION

The LSWSNs datasets that consist of network sizes (N) from 100 through 5000 nodes are utilized. The algorithms are run repeatedly for M times, for determining the statistical significance of the results. The aim of this experiment is to evaluate performance of the employed HHO algorithm using fitness function mentioned earlier. For results verification, six of the most well-known algorithms, such as PSO, FPA, GWO, SCA, MVO, and WOA, are used and employed for comparison purposes. The statistical results are collected and presented qualitatively and quantitatively in this section. Note that we have utilized 1000 iterations, 30 runs, and from 100 through 5000 search agents according to different network sizes in the experiments. For the quantitative results, it should be noted that we have employed a wide range of performance metrics to quantify performance of the employed algorithms. The performance measures in terms of mean, best, worst, and standard deviation for all the competitive algorithms are summarized in Table 7, which proves that that the proposed HHO algorithm obtained better results than the competitive algorithms for all network sizes. Also, these empirical results confirm the ability of HHO to choose the nearest best position of sink node with low energy consumption.

To sum up the obtained statistical results, Table 8 presents the average evaluation criteria in terms of mean, best, and worst obtained over a certain number of generations and according to all the considered network sizes for competitive algorithms via the fitness function mentioned. The best performance is achieved by the employed HHO algorithm, proving its ability to minimize the fitness function and locating the sink node in the optimal position effectively. Also, the graphical representation is illustrated in Fig. 9. As it is clear in Table 8 and Fig. 9, the HHO algorithm is the best one in achieving the fitness function.

To analyze time complexity of the seven algorithms after 1000 iterations, Table 9 and Fig. 10 demonstrate the running time in seconds. It is obvious that the running time of HHO is better than the other algorithms, which take long time to find the nearest best position of the sink node in the whole network with low energy consumption.

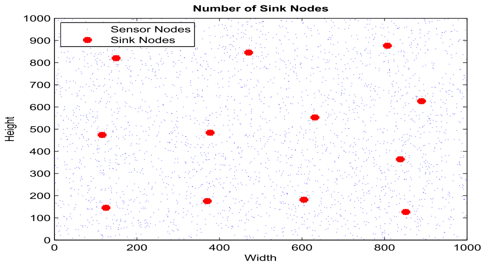
B. MULTIPLE SINK NODE PLACEMENT

Another important issue in LSWSNs is how to determine the minimum number of multiple sink nodes and checking the best cardinality of the sink nodes that each algorithm can obtain. To obtain the minimum number of the sink nodes generated by each algorithm, Eq. 14 is used as a fitness function by each algorithm to produce the minimum number of the sink nodes that can serve the whole network.

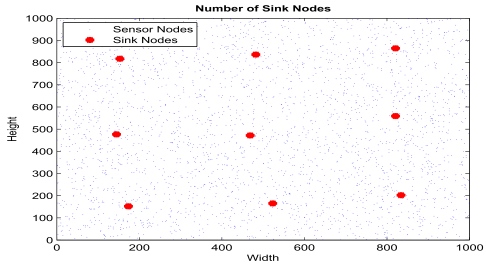
$$f(x) = \frac{ACN}{NUM_{nbr}}, \tag{14}$$

where ACN denotes the number of sensor neighbor served by the sink node, NUM_{nbr} represents the number of active nodes. This function is dependent on the position vector x of all nodes. The localization optimization can be formulated as: $\min f(x)$.

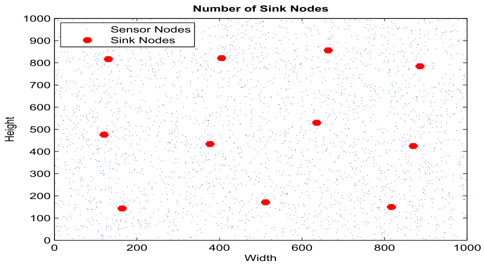
All obtained results are summarized in Table 10 and illustrated in Fig. 11, which depicts the cardinality results of the sink nodes curve from HHO compared with other algorithms for LSWSNs. As shown in Fig. 11, the number of the sink nodes obtained from the employed algorithm has been balanced between network size and the number of the sink nodes through all network sizes, compared with other algorithms. Fig. 12a-g demonstrates the best cardinality of the sink nodes has been obtained from PSO, FPA, GWO, SCA, MVO, WOA, and HHO respectively. Because of limited space, only the network size with 3000 nodes is illustrated, where red points show the location of the sink nodes in the network. As observed in Table 10, Fig. 11 and Fig. 12 that the HHO algorithm achieves the best results compared to other algorithms. The HHO algorithm located minimum sink nodes in all different network sizes compared to the competitive algorithms, which reflects that it achieved best prolonged lifetime of the networks.



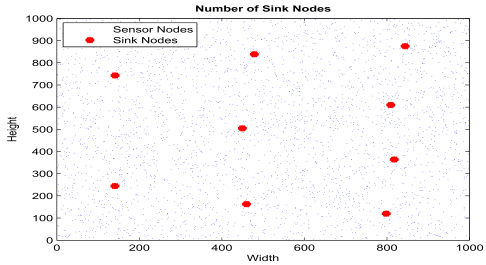
(a) Particle Swarm Optimization



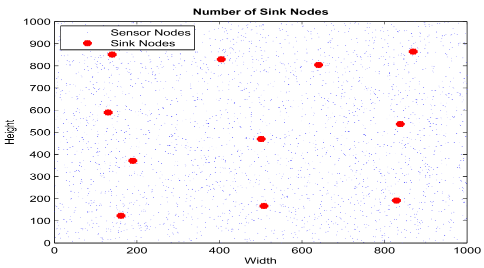
(b) Flower Pollination Algorithm



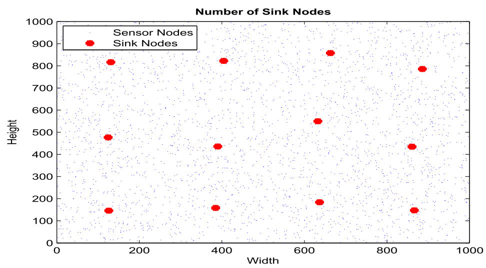
(c) Grey Wolf Optimizer



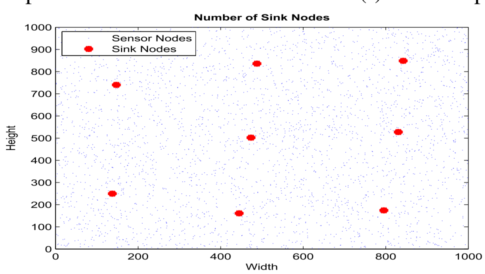
(d) Sine Cosine Algorithm



(e) Multi-Verse Optimizer



(f) Whale Optimization Algorithm



(g) Harris Hawks Optimization

FIGURE 12. **Multiple sink node positions for network size of 3000 nodes.**

VI. CONCLUSION

The sink node placement in large-scale wireless sensor networks (LSWSNs) plays a crucial role in extending its lifetime, as the sensors deployed in the field are often energy-constrained. Formulating this issue as an optimization problem resulted in proposing several metaheuristic algorithms that have achieved remarkable results on small to medium-sized networks. To address this issue on the large scale networks, the current work employed a new efficient optimization algorithm called Harris' hawk optimization (HHO) on the sink node placement with network size ranging from 100 to 5000 nodes. The major objective was to maximize the network lifetime by choosing the optimal position for the sink node in the whole network. To this end, we utilized the greedy algorithm with minimum spanning tree to create the minimum data transmission paths, for constructing the network according to the new sink node location. Considering both the single sink node and multiple sink nodes placement problems, we compared HHO results with seven other well-known metaheuristic algorithms. Based on several evaluation metrics, the HHO algorithm achieved the best solutions with the lowest energy consumption and localization error by using the proposed fitness function. Additionally, the experimental results of ten different network sizes indicated that the

HHO algorithm showed significant improvement in topology construction time compared to the counterparts.

COMPLIANCE WITH ETHICAL STANDARDS

The authors declare that they have no conflict of interests.

REFERENCES

[1] A. Mainwaring, D. Culler, J. Polastre, R. Szewczyk, and J. Anderson, ''Wireless sensor networks for habitat monitoring,'' in *Proc. 1st ACM Int. Workshop Wireless Sensor Netw. Appl. (WSNA)*, Atlanta, GA, USA, 2002, pp. 88–97, doi: 10.1145/570738.570751.

[2] I. F. Akyildiz, W. Su, Y. Sankarasubramaniam, and E. Cayirci, ''Wireless sensor networks: A survey,'' *Comput. Netw.*, vol. 38, no. 4, pp. 393–422, Mar. 2002, doi: 10.1016/S1389-1286(01)00302-4.

[3] E. Tuba, D. Simian, E. Dolicanin, R. Jovanovic, and M. Tuba, ''Energy efficient sink placement in wireless sensor networks by brain storm optimization algorithm,'' in *Proc. 14th Int. Wireless Commun. Mobile Comput. Conf. (IWCMC)*, Limassol, Cyprus, Jun. 2018, pp. 718–723, doi: 10.1109/iwcmc.2018.8450333.

[4] V. Snasel, L. Kong, P. Tsai, and J.-S. Pan, ''Sink node placement strategies based on cat swarm optimization algorithm,'' *J. Netw. Intell.*, vol. 1, no. 2, pp. 52–60, May 2016.

[5] F. Castaño, E. Bourreau, A. Rossi, M. Sevaux, and N. Velasco, ''Partial target coverage to extend the lifetime in wireless multi-role sensor networks,'' *Networks*, vol. 68, no. 1, pp. 34–53, Aug. 2016, doi: 10.1002/net.21682.

[6] P. M. Wightman and M. A. Labrador, ''A3: A topology construction algorithm for wireless sensor networks,'' in *Proc. IEEE Global Telecommun. Conf. (GLOBECOM)*, New Orleans, LO, USA, Dec. 2008, pp. 1–6.

[7] Z. Yuanyuan, X. Jia, and H. Yanxiang, ''Energy efficient distributed connected dominating sets construction in wireless sensor networks,'' in *Proc. Int. Conf. Commun. Mobile Comput. (IWCMC)*, Vancouver, BC, Canada, 2006, pp. 797–802, doi: 10.1145/1143549.1143709.

[8] J. Wu, M. Cardei, F. Dai, and S. Yang, ''Extended dominating set and its applications in ad hoc networks using cooperative communication,'' *IEEE Trans. Parallel Distrib. Syst.*, vol. 17, no. 8, pp. 851–864, Aug. 2006.

[9] M. M. Ahmed, E. H. Houssein, A. E. Hassanien, A. Taha, and E. Hassanien, ''Maximizing lifetime of large-scale wireless sensor networks using multi-objective whale optimization algorithm,'' *Telecommun. Syst.*, vol. 72, no. 2, pp. 243–259, Oct. 2019, doi: 10.1007/s11235-019-00559-7.

[10] M. M. Ahmed, E. H. Houssein, A. E. Hassanien, A. Taha, and E. Hassanien, ''Maximizing lifetime of wireless sensor networks based on whale optimization algorithm,'' in *Proc. AISI*, Cairo, Egypt, 2017, pp. 724–733, doi: 10.1007/978-3-319-64861-3_68.

[11] A. Efrat, S. Har-Peled, and J. Mitchell, ''Approximation algorithms for two optimal location problems in sensor networks,'' in *Proc. 2nd Int. Conf. Broadband Netw.*, Boston, MA, USA, Feb. 2006, pp. 714–723, doi: 10.1109/icbn.2005.1589677.

[12] J. Luo and J.-P. Hubaux, ''Joint mobility and routing for lifetime elongation in wireless sensor networks,'' in *Proc. IEEE 24th Annu. Joint Conf. IEEE Comput. Commun. Soci.*, Miami, FL, USA, Aug. 2005, pp. 1735–1746, doi: 10.1109/infcom.2005.1498454.

[13] M. M. Fouad, V. Snasel, and A. E. Hassanien, ''Energy-aware sink node localization algorithm for wireless sensor networks,'' *Int. J. Distrib. Sensor Netw.*, vol. 11, no. 7, Jul. 2015, Art. no. 810356, doi: 10.1155/2015/810356.

[14] F. A. Hashim, E. H. Houssein, M. S. Mabrouk, W. Al-Atabany, and S. Mirjalili, ''Henry gas solubility optimization: A novel physics-based algorithm,'' *Future Gener. Comput. Syst.*, vol. 101, pp. 646–667, Dec. 2019.

[15] E. H. Houssein, M. E. Hosney, D. Oliva, W. M. Mohamed, and M. Hassaballah, ''A novel hybrid Harris hawks optimization and support vector machines for drug design and discovery,'' *Comput. Chem. Eng.*, vol. 133, Feb. 2020, Art. no. 106656.

[16] K. Hussain, M. N. Mohd Salleh, S. Cheng, and Y. Shi, ''Metaheuristic research: A comprehensive survey,'' *Artif. Intell. Rev.*, vol. 52, no. 4, pp. 2191–2233, Dec. 2019, doi: 10.1007/s10462-017-9605-z.

[17] H. Banka and P. K. Jana, ''PSO-based multiple-sink placement algorithm for protracting the lifetime of wireless sensor networks,'' in *Proc. 2nd Int. Conf. Comput. Commun. Technol.* New Delhi, India: Springer, 2016, pp. 605–616, doi: 10.1007/978-81-322-2517-1_58.

[18] F. Chen and R. Li, ''Sink node placement strategies for wireless sensor networks,'' *Wireless Pers. Commun.*, vol. 68, no. 2, pp. 303–319, Jan. 2013, doi: 10.1007/s11277-011-0453-x.

[19] A. A. Heidari, S. Mirjalili, H. Faris, I. Aljarah, M. Mafarja, and H. Chen, ''Harris hawks optimization: Algorithm and applications,'' *Future Gener. Comput. Syst.*, vol. 97, pp. 849–872, Aug. 2019, doi: 10.1016/j.future.2019.02.028.

[20] B. Y. Wu and K.-M. Chao, *Spanning Trees and Optimization Problems*. Boca Raton, FL, USA: CRC Press, 2004.

[21] D. Cheriton and R. E. Tarjan, ''Finding minimum spanning trees,'' *SIAM J. Comput.*, vol. 5, no. 4, pp. 724–742, Dec. 1976, doi: 10.1137/0205051.

[22] R. Eberhart and J. Kennedy, ''A new optimizer using particle swarm theory,'' in *Proc. 6th Int. Symp. Micro Mach. Hum. Sci. (MHS)*, Nagoya, Japan, Nov. 1995, pp. 39–43, doi: 10.1109/mhs.1995.494215.

[23] X.-S. Yang, ''Flower pollination algorithm for global optimization,'' in *Unconventional Computation and Natural Computation* (Lecture Notes in Computer Science), vol. 7445. Berlin, Germany: Springer, 1995, pp. 39–43, doi: 10.1007/978-3-642-32894-7_27.

[24] S. Mirjalili, S. M. Mirjalili, and A. Lewis, ''Grey wolf optimizer,'' *Adv. Eng. Softw.*, vol. 69, pp. 46–61, Mar. 2014, doi: 10.1016/j.advengsoft.2013.12.007.

[25] S. Mirjalili, ''SCA: A sine cosine algorithm for solving optimization problems,'' *Knowl.-Based Syst.*, vol. 96, pp. 120–133, Mar. 2016, doi: 10.1016/j.knosys.2015.12.022.

[26] S. Mirjalili, S. M. Mirjalili, and A. Hatamlou, ''Multi-verse optimizer: A nature-inspired algorithm for global optimization,'' *Neural Comput. Appl.*, vol. 27, no. 2, pp. 495–513, Feb. 2016, doi: 10.1007/s00521-015-1870-7.

[27] S. Mirjalili and A. Lewis, ''The whale optimization algorithm,'' *Adv. Eng. Softw.*, vol. 95, pp. 51–67, May 2016, doi: 10.1016/j.advengsoft.2016.01.008.

[28] I. Strumberger, M. Minovic, M. Tuba, and N. Bacanin, ''Performance of elephant herding optimization and tree growth algorithm adapted for node localization in wireless sensor networks,'' *Sensors*, vol. 19, no. 11, p. 2515, Jun. 2019, doi: 10.3390/s19112515.

[29] S.-C. Chu, P. W. Tsai, and J. S. Pan, ''Cat swarm optimization,'' in *Proc. PRICAI*, Guilin, China, 2016, pp. 854–858, doi: 10.1007/978-3-540-36668-3_94.

[30] M. N. Rahman and M. A. Matin, ''Efficient algorithm for prolonging network lifetime of wireless sensor networks,'' *Tinshhua Sci. Technol.*, vol. 16, no. 6, pp. 561–568, Dec. 2011, doi: 10.1016/s1007-0214(11)70075-x.

[31] M. M. Fouad, V. Snasel, and A. E. Hassanien, ''An adaptive PSO-based sink node localization approach for wireless sensor networks,'' in *Proc. AECIA*, Villejuif, France, 2015, pp. 679–688, doi: 10.1007/978-3-319-29504-6_62.

[32] J. H. Holland, *Adaptation in Natural and Artificial Systems: An Introductory Analysis with Applications to Biology, Control and Artificial Intelligence*. Cambridge, MA, USA: MIT Press, 1992.

[33] S. Ghosh, I. Snigdh, and A. Singh, ''GA optimal sink placement for maximizing coverage in wireless sensor networks,'' in *Proc. Int. Conf. Wireless Commun., Signal Process. Netw. (WiSPNET)*, Chennai, India, Mar. 2016, pp. 737–741, doi: 10.1109/wispnet.2016.7566231.

[34] M. Dorigo, M. Birattari, and T. Stutzle, ''Ant colony optimization,'' *IEEE Comput. Intell. Mag.*, vol. 1, no. 4, pp. 28–39, Nov. 2006, doi: 10.1109/MCI.2006.329691.

[35] Y. Lin, J. Zhang, H. S.-H. Chung, W. H. Ip, Y. Li, and Y.-H. Shi, ''An ant colony optimization approach for maximizing the lifetime of heterogeneous wireless sensor networks,'' *IEEE Trans. Syst., Man, Cybern. C, Appl. Rev.*, vol. 42, no. 3, pp. 408–420, May 2012.

[36] A. M. Shamsan Saleh, B. Mohd Ali, M. F. A. Rasid, and A. Ismail, ''A self-optimizing scheme for energy balanced routing in wireless sensor networks using sensorAnt,'' *Sensors*, vol. 12, no. 8, pp. 11307–11333, Aug. 2012, doi: 10.3390/s120811307.

[37] J. V. V. Sobral, R. A. L. Rabelo, H. S. Araujo, R. A. R. S. Baluz, and R. H. Filho, ''Automated design of fuzzy rule base using ant colony optimization for improving the performance in wireless sensor networks,'' in *Proc. IEEE Int. Conf. Fuzzy Syst. (FUZZ-IEEE)*, Hyderabad, India, Jul. 2013, pp. 1–8, doi: 10.1109/fuzz-ieee.2013.6622416.

[38] J. Zhang and L. Gao, ''Clustering routing algorithm for WSN based on improved ant colony algorithm,'' in *Proc. Int. Conf. Electr. Control Eng.*, Sep. 2011, pp. 2924–2928, doi: 10.1109/iceceng.2011.6057312.

[39] X.-S. Yang and S. Deb, ''Cuckoo search: Recent advances and applications,'' *Neural Comput. Appl.*, vol. 24, no. 1, pp. 169–174, Jan. 2014, doi: 10.1007/s00521-013-1367-1.

[40] J. Cheng and L. Xia, ''An effective cuckoo search algorithm for node localization in wireless sensor network,'' *Sensors*, vol. 16, no. 9, p. 1390, Aug. 2016, doi: 10.3390/s16091390.

[41] S. Saremi, S. Mirjalili, and A. Lewis, ''Grasshopper optimisation algorithm: Theory and application,'' *Adv. Eng. Softw.*, vol. 105, pp. 30–47, Mar. 2017, doi: 10.1016/j.advengsoft.2017.01.004.

[42] S. Mirjalili, A. H. Gandomi, S. Z. Mirjalili, S. Saremi, H. Faris, and S. M. Mirjalili, ''Salp swarm algorithm: A bio-inspired optimizer for engineering design problems,'' *Adv. Eng. Softw.*, vol. 114, pp. 163–191, Dec. 2017, doi: 10.1016/j.advengsoft.2017.07.002.

[43] Y. Shi, ''Brain storm optimization algorithm,'' in *Proc. ICSI*, Chiang Mai, Thailand, 2011, pp. 303–309, doi: 10.1007/978-3-642-21515-5_36.

[44] M. M. Fouad, A. I. Hafez, A. E. Hassanien, and V. Snasel, ''Grey wolves optimizer-based localization approach in WSNs,'' in *Proc. 11th Int. Comput. Eng. Conf. (ICENCO)*, Cairo, Egypt, Dec. 2015, pp. 256–260, doi: 10.1109/icenco.2015.7416358.

[45] M. M. Fouad, A. I. Hafez, and A. E. Hassanien, ''Optimizing topologies in wireless sensor networks: A comparative analysis between the grey wolves and the chicken swarm optimization algorithms,'' *Comput. Netw.*, vol. 163, Nov. 2019, Art. no. 106882, doi: 10.1016/j.comnet.2019.106882.

[46] Y. Tan, ''A new bio-inspired algorithm: Chicken swarm optimization,'' in *Advances in Swarm Intelligence*, vol. 8794. Cham, Switzerland: Springer, 2014, pp. 86–94, doi: 10.1007/978-3-319-11857-4_10.

[47] H. Li, Y. Liu, W. Chen, W. Jia, B. Li, and J. Xiong, ''COCA: Constructing optimal clustering architecture to maximize sensor network lifetime,'' *Comput. Commun.*, vol. 36, no. 3, pp. 256–268, Feb. 2013.

[48] H. Nakano, M. Yoshimura, A. Utani, A. Miyauchi, and H. Yamamoto, ''A sink node allocation scheme in wireless sensor networks using suppression particle swarm optimization,'' *Sustain. Wireless Sensor Netw.*, to be published, doi: 10.5772/13762.

[49] A. Bogdanov, E. Maneva, and S. Riesenfeld, ''Power-aware base station positioning for sensor networks,'' in *Proc. IEEE INFOCOM*, Hong Kong, Dec. 2004, pp. 575–585.

[50] D. Mechta and S. Harous, ''Prolonging WSN lifetime using a new scheme for sink moving based on artificial fish swarm algorithm,'' in *Proc. 2nd Int. Conf. Adv. Wireless Inf., Data, Commun. Technol. (AWICT)*, Paris, France, 2017, pp. 1–5, doi: 10.1145/3231830.3231837.

[51] P. M. Wightman and M. A. Labrador, ''Atarraya: A simulation tool to teach and research topology control algorithms for wireless sensor networks,'' in *Proc. 2nd Int. ICST Conf. Simulation Tools Techn.*, 2009, p. 26, doi: 10.4108/icst.simutools2009.5565.

ESSAM H. HOUSSEIN received the Ph.D. degree in computer science wireless networks based on artificial intelligence, in 2012. He is currently works as an Associate Professor with the Computer Science Department, Faculty of Computers and Information, Minia University, Egypt. He has more than 50 scientific research articles published in prestigious international journals in the topics of machine learning and its applications. His research interests include wireless sensor networks, cloud computing, security, soft computing, image processing, artificial intelligence, data mining, optimization, and meta-heuristics techniques.

MOHAMMED R. SAAD received the B.Sc. degree in computer science from the Faculty of Computers and Information, Minia University, in 2016. His research interests include wireless sensor networks, soft computing, and data mining.

KASHIF HUSSAIN received the B.S. degree in information technology from the Mehran University of Engineering and Technology, Jamshoro, Pakistan, in 2006, and the M.S. and PhD. degrees in information technology from Universiti Tun Hussein Onn Malaysia, Johor, Malaysia, in 2015 and 2018, respectively. Then, he spent seven years in industry as a Software Engineer for several software development companies in Karachi, Pakistan. From 2014 to 2018, he served as a Research Assistant with Universiti Tun Hussein Onn Malaysia. Since 2019, he has been serving with the Institute of Fundamental and Frontier Sciences, University of Electronic Science and Technology of China, Chengdu, China, as a Postdoctoral Researcher. His major research interests include machine learning, optimization, fuzzy neural networks, and metaheuristic algorithms.

WILLIAM ZHU (Senior Member, IEEE) received the M.S. degree in mathematics from Xinjiang University, Wulumuqi, Xinjiang, China, in 1986, the M.S. degree in systems engineering from the University of Arizona, Tucson, AZ, USA, in 2002, and the Ph.D. degrees from Northwestern Polytechnical University, Xi'an, China, in 2001, from the University of Arizona, in 2003, and from the University of Auckland, Auckland, New Zealand, in 2006. He worked at a university in China from 1986 to 2001 and was promoted to an Associate Professor in computer science, in 1999. Since 2000, he has been collaborating with Dr. F.-Y. Wang from the Institute of Automation, Chinese Academy of Sciences, Beijing. He is currently a Professor with the Institute of Fundamental and Frontier Sciences, University of Electronic Science and Technology of China, Chengdu, China. His research interests include granular computing, rough set, data mining, and artificial intelligence. He was listed in Highly Cited Researchers in computer science by Thomson Reuters in 2015, 2016, and 2017.

HASSAN SHABAN received the Ph.D. degree in computer science. He currently works as a Lecturer with the Computer Science Department, Faculty of Computers and Information, Minia University, Egypt. His research interests include wireless sensor networks, security, optimization, and metaheuristics.

M. HASSABALLAH received the B.Sc. degree in mathematics and the M.Sc. degree in computer science from South Valley University, Egypt, in 1997 and 2003, respectively, and the D.Eng. degree in computer science from Ehime University, Japan, in 2011. He was a Visiting Scholar with the Department of Computer and Communication Science, Wakayama University, Japan, in 2013, and the GREAH laboratory, Le Havre Normandie University, France, in 2019. He is currently an Associate Professor of computer science with the Faculty of Computers and Information, South Valley University, Egypt. He has published five books and more than 50 research articles in refereed international journals and conferences. His research interests include feature extraction, object detection/recognition, artificial intelligence, biometrics, image processing, computer vision, machine learning, and data hiding. He served as a Reviewer for several journals such as the IEEE Transactions on Image Processing, the IEEE Transactions on Fuzzy Systems, *Pattern Recognition*, *Pattern Recognition Letters*, *IET Image Processing*, *IET Computer Vision*, *IET Biometrics*, the *Journal of Real-Time Image Processing*, the *Journal of Electronic Imaging*, and *Optical Engineering*.