EAI Endorsed Transactions
on Scalable Information Systems

Research Article EAI.EU

HHO-LPWSN: Harris Hawks Optimization Algorithm for Sensor Nodes Localization Problem in Wireless Sensor Networks

Ravi Sharma[1,*] and Shiva Prakash[2]

[1]Department of Computer Science and Engineering, Madan Mohan Malaviya University of Technology – Gorakhpur, India
[2]Department of Information Technology and Computer Application, Madan Mohan Malaviya University of Technology – Gorakhpur, India.

Abstract

Wireless sensor network (WSN) is a prominent technology for remote area monitoring with the assimilation of the Internet of Things (IoT). Over the past decades, sensor node localization has become an essential challenge of WSNs. The sensor indicates localization challenges related to NP-hard problems. Nature-inspired computational intelligence algorithms are used to solve NP-hard problems efficiently for sensor node localization. After the rigorous advanced search in reputable research journals, efficient newly designed Harris Hawks Optimization (HHO) algorithm has not been used to sensor nodes localization until now. Therefore, this paper does and compares the proposed work from the recently available well-known optimization algorithms such as the Salp Swarm Algorithm (SSA), Equilibrium Optimizer (EO), and Grey Wolf Optimizer (GWO). The simulation results of the proposed work showed that it can outperform in terms of average localization error, the number of localized sensor nodes, and computational cost compared to other computational intelligence algorithms.

Keywords: Wireless Sensor Networks, Sensor Nodes, Localization Error, Computational Intelligence, Anchor Nodes, Location Optimization.

Received on 20 May 2020, accepted on 12 February 2021, published on 25 February 2021

doi: 10.4108/eai.25-2-2021.168807

*Corresponding author. ravi.cs.0904@gmail.com

1. Introduction

Today is the era of technological automation [1], where systems are designed with the help of global networks (Internet) in such a way that human intervention would be minimized. Researchers worked with the IoT system to meet all the requirements of technical automation [2-5]. These types of systems consume a lot of data [6] to solve real-time challenges. A large amount of realistic data can be collected using only WSNs. Researchers are more concerned about the design of WSN-IoT system integration [7]. Real-time data are collected by sensor nodes under the umbrella of a WSN. The collected data of the sensor nodes have no meaning until the WSN knows its actual state. Thus, the localization of sensor nodes becomes an important challenge for WSNs [8-10].

The localization algorithm is classified into two parts, such as range-based and range-free-based localization approaches. Range-based localization approaches [11] are designs based on distance or angle calculation between nodes and while range-free-based localization approaches [12] use hop count between sensor nodes to estimate the coordination of sensor nodes. The range-based localization approach is the Received Signal Strength Indicator (RSSI) [13], Time of Arrival (ToA) [14], Angle of Arrival (AoA) [15] and Time Difference of Arrival (TDoA) [16]. Range-free-based localization approaches are Distance Vector-Hop (DV-Hop) [17], Ad-Hoc Positioning System (APS) [18], and Multi-Dimensional Scaling (MDS) [19]. In range-based localization approaches, anchor nodes information is required to estimate the coordination of sensor nodes.

Anchor nodes are nodes whose coordinate information is known in the system. For sensor node localization the needs at least three anchor nodes. [20]. The cost of anchor nodes in the system is higher than the deployment of sensor nodes due to the additional cost of the Global Positioning System (GPS) equipped with anchor nodes [21]. The hardware cost of localization can be solved efficiently using computational intelligence algorithms for sensor nodes. Computational intelligence algorithms are usually designed based on the working principle of nature-induced behavior of humans-beings. Artificial intelligence incorporated into localization modules using computational intelligence algorithms [22]. Thus, there is a need to estimate the location of sensor nodes optimally using computational intelligence algorithms.

Numerous computational intelligence algorithms are available to find the optimal solution for sensor node localization problems but still, there is a need to achieve the fast convergence speed optimization algorithm for sensor node localization by optimally balancing the total number sensor node localization and mean error rate. The newly designed computational intelligence algorithm [23] in which the author claimed that the HHO algorithm outperforms in terms of statistical results compared to other well-known optimizers. Thus, the rigorous advanced search in reputable research journals found that this efficient newly designed HHO algorithm has not been used for sensor node localization until now. Therefore the main contributions of this paper are:

1. A newly designed HHO computational intelligence algorithm by Heidari et al. [23] is used to solve the localization problem of sensor nodes in WSN.
2. The proposed work implementation using the MatLab tool is presented.
3. The design work of this paper is compared with other computational intelligence algorithms such as SSA, GWO, and EO.
4. Performance analysis parameters for the suggested work in terms of mean localization error, computational cost, and the number of localized sensor nodes.

This paper is structured as follows: section two presents computational intelligence algorithms, section three presents the literature survey of esteemed existing works in the field of anchors-based sensor nodes localization, section four provides the proposed approaches model, flowchart and algorithm, section five provides the proposed work evaluation between them in terms of mean localization error, computation cost, the number of localized nodes and section six presents the conclusion of designed paper works.

2. Computational Intelligence Algorithms

Computational intelligence algorithms are nature-inspired algorithms; nowadays, popularly used in interdisciplinary to achieve optimal results. In this section, various well-known computational intelligence algorithms like SSA, GWO, EO, and HHO are presented in detail below:

Salp Swarm Algorithm (SSA)

Mirjalili et al. [24] proposed the SSA algorithm, which mimics the social interaction behavior of salp swarms.SSA is a population-based Swarm Intelligence (SI) algorithm. The slap has a transparent barrel-like body, and its tissues are like a jellyfish structure. They live underneath the sea and search for their food by the salp chains. The salps chain is divided into two categories as leaders and followers. Leader saps update their location according to equation 1:

$$x_j^1 = \begin{cases} F_j + c_1\left(\left(Ub_j + lb_j\right)c_2 + lb_j\right) c_3 \geq 0 \\ F_j - c_1\left(\left(Ub_j + lb_j\right)c_2 + lb_j\right) c_3 < 0 \end{cases} \quad (1)$$

Where $x_j^1$is represented as the leader's location in the j[th] dimension, Ub_j and lb_j represented as the upper bound and lower bound of the j[th] dimension of the target region, F_j denotes target localization of food, and c_1, c_2, c_3 are random variables.

The follower slaps his place according to equation 2.

$$x_j^i = \frac{1}{2}at^2 + v_0 t \quad (2)$$

Where $i \geq 2, x_j^i$ is the position of the follower slap in the j[th] dimension, v_0 and v_{final} are the initial and final velocities, $t$represented as time and $a = \frac{v_{final}}{v_0}$

Grey Wolf Optimizer (GWO)

Mirjalili et al. [25] proposed the GWO algorithm from leadership qualities inspired by Grey Wolves. It is a swarm computational intelligence algorithm similar to PSO, Ant Colony Optimization (ABC) algorithm. This mimics the lead pecking order and the relationship of wolves. The social pecking order is simulated by classifying the population of search agents based on their fitness:

- Level 1 (Alpha):
  This is the leader who is male or female. Alpha is mostly responsible for decision-making (such as hunting, sleeping places, etc.).Others accept alpha by putting their tails down.
- Level 2 (Beta):
  Betas are subordinate wolves that help alpha in making decisions. Beta is an advisor to the Alpha of this pack. They consider the best candidate to be an alpha when the alpha dies or becomes too old. Beta ensures Alpha's orders are followed, and it also provides them with feedback.
- Level 3 (Delta):
  Deltas are also subordinate wolves. Delta wolves dominate Omega and report to Alpha and Beta. The delta can be classified as scouts, sentinels, elders, hunters, caretakers.
- Level 4 (Omega):
  It is like a sacrificial goat in a pack.

GWO Search Process: The model demonstrated mimic hunting behavior of grey wolves to use three stages, searching, circling, and attacking prey. The first two stages are given to the exploration process, and the last one presents the exploitation process.

- Searching (Exploration): Grey wolves typically detect the search process according to alpha, beta, and delta positions. They distributed themselves from one another to exploit to locate prey and attack prey. The GWO algorithm uses the A constraint, in which A is a random value, and its value is greater than 1 or less than -1. The search agents may diverge from the prey when |A| > 1, and they force to diverge for finding a better one.
- Encircling (Exploration): Grey wolves encircling the prey before hunting. The encircling behavior calculated by using mathematical equations (3) and (4) are as follows:

$$\vec{D} = |\vec{C}.\overrightarrow{X_p}(t) - \vec{X}(t)| \quad (3)$$

$$\vec{X}(t+1) = \overrightarrow{X_p}(t) - \vec{A}.\vec{D} \quad (4)$$

  Where t represents the current iteration,$\vec{A}$and $\vec{C}$ are coefficient vectors, $\overrightarrow{X_p}$ is the prey position vector, $\vec{X}$ presents the Grey Wolves position vector and $\vec{X}(t+1)$ is the next position vector of Grey Wolves.
- Attacking Prey (Exploitation): Grey wolves end the hunt when the prey stops moving. In the GWO algorithm, when |A| < 1, then the wolves attack the prey.

Equilibrium Optimizer (EO)

Faramarzi et al. [26] proposed an optimization technique to induce control volume mass prototyping. In EO, each particle denotes solution and concentration as position. The concentration acts as a search agent in EO and is updated according to the best-so-far solution. The best solution obtained is known as the final equilibrium state. The EO algorithm is modeled in equation 5 by updating the rules.

$$\vec{C} = \vec{C}_{eq} + (\vec{C} - \vec{C}_{eq}).\vec{F} + \frac{\vec{G}}{\vec{\lambda}V}(1 - \vec{F}) \quad (5)$$

$\vec{C}$is represented as a concentration vector, $\vec{C}_{eq}$is presented as an equilibrium candidates vector, $\vec{F}$is represented as the exponential term vector for the concentration update rule, $\vec{\lambda}$denotes a random vector between [0, 1], $\vec{G}$ is represented as a generation rate vector, $V$is represented as the control volume of$C$.

Harris Hawks Optimization (HHO)

Heidari et al. [23] proposed a nature-inspired computational intelligence algorithm adopting the harris hawk's behavioral style of prey pursuit. The several hawks cooperatively pounce to surprise the prey. Harris Hawks has a unique cooperative pursuit strategy based on conditions of dynamic nature and escape strategies of prey. The hawks show innovative team spirit to chase strength in terms of hunting, encircling, and getting out of the hunt. The exploration and exploitation steps of the HHO algorithm are as follows:

- Exploration Phase:
  In exploration, the harris hawks use their powerful eyes to locate prey. Harris Hawks is randomly perched in several locations, and they explore the possibility of hunting on two occasions based on q value. If q> 0.5, they are close enough to attack prey, and they sit on the random tallest tree, which is modeled in the equation.

$$X(t+1) = \begin{cases} X_{rand}(t) - r_1|X_{rand}(t) - 2r_2X(t)| & q \geq 0.5 \\ (X_{rabbit}(t) - X_m(t)) - r_3(LB + r_4(UB - LB)) & q < 0.5 \end{cases} \quad (6)$$

  Where X (t + 1) is represented as the next t iteration of the hawk's vector position,$X_{rabbit}(t)$shows the current position of the rabbit, $X(t)$is shown as the current position of the hawkers, r_1, r_2, r_3, r_4 and $q$have random values in the interval (0, 1), LB and UB are the upper and lower limits of the variables., $X_{rand}(t)$is represented as a randomly selected hawk from the current population, and $X_m(t)$is denoted as the average position of Hawke's current position.
- Exploitation phase:
  In the exploitation phase, there is a chance to attack an already identified prey.

3. Literature Survey

This section provides a critical analysis of the latest research works available in the field of anchor-based localization in WSNs using computational algorithms.

The salp swarm optimization algorithm is proposed by Kanoosh et al. [27] for localizing sensor nodes in WSNs. In WSN, the location accuracy of sensor node localization is greatly affected by the salp swarm algorithm compared to particle swarm optimization, butterfly optimization algorithm, firefly algorithm, grey wolf optimizer. The simulation result shows that the performance of the proposed algorithm is much better than other localization algorithms in terms of the number of localized nodes, localization error, and computing cost.

Rajkumar et al. [28] proposed work by incorporating the Grey Wolf Optimization (GWO) algorithm to detect the accurate geographic location of unknown sensor nodes with the help of anchor nodes in WSNs. The GWO algorithm mimics the social behavior of a grey wolf leadership to attack targets. The suggested work is implemented using the MatLab tool for randomly deployed sensor nodes in the target region. Parameters such as computation cost, localized node percentage, the minimum number of error measures for analysis of GWO's ability, and other types of metaheuristic algorithms. The result of faster convergence and the success rate of the GWO algorithm is better than other PSO and other metaheuristics algorithms like the Modified BAT Algorithm (MBA).

Table 1. Taxonomy of anchor-based sensor nodes localization approaches for sensor nodes using computational intelligence algorithms in WSN.

Authors	Year of Publ-ication	Design approach	Techniques used	Compared approaches	Target parameters	Simulation Tool
Kanoosh et al. [27]	2019	Salp Swarm Algorithm for Node Localization in WSNs	PSO, BOA, FA, GWO	Salp Swarm Algorithm	Mean localization error, Number of localized nodes	MatLab
Rajakumar et al. [28]	2017	GWO algorithm for node localization problem in WSNs	GWO	PSO, MBA	Computation cost, minimum localization error, localized nodes	MatLab
Tuba et al. [29]	2018	FA-based sensor nodes localization in two-stage	Semi-mobile nodes, Firefly optimization algorithm	3D Localization, PSO Algorithm, (TLP), BA	Improve localization accuracy	MatLab
Strumberger et al. [30]	2018	Sensor nodes localization using MBO algorithm in WSN	Monarch butterfly optimization	PSO, MPSO, ABC, MSABC, MBO	2.5% of anchor nodes with (20 m 50 m), 10% of anchor nodes with 50m	MatLab
Alomari et al. [31]	2018	To obstacle avoidance for mobile anchor nodes using SI optimization algorithms	GWO, WOA	Snake-like, Z-curves	Localization ratio, Localization error, Computation cost	MatLab
Strumberger et al. [32]	2018	WSN localization using EHO algorithm	EHO algorithm	PSO, Multi step PSO, ABC, Multi step ABC	Mean squared error	Experim-ental setup
Strumberger et al. [33]	2019	A node localization in WSNs using EHO and tree growth algorithm	EHO algorithm, tree growth algorithm	Iterative best performance algorithm, taboo search, largest absolute difference algorithm	Localized number nodes, localization error, execution time	--
Tan et al. [34]	2019	A sensor node localization using distance mapping algorithm	DMA, optimized linear transforming function, GA	DV hop, MDS map	Localization error, Total consumption of energy	Network Simulator

Tuba et al. [29] proposed two-stage sensor node localization using a firefly algorithm. In the WSN, the RSSI (Received Signal Strength Signal) propagation model is used to estimate the distance between the anchor nodes and the semi anchor nodes. The proposed algorithm for the localization of the sensor node follows a two-part: first, four anchor nodes are placed at the corners of the target area coverage, and secondly the estimation of the optimal distance using distance calculation. The future direction of this work for an optimal approach for the localization of sensors with firefly algorithm modification and adjustment.

Monarch Butterfly Optimization (MBO) algorithm used by Strumberger et al. [30] to solve the NP-hard problem of WSN localization. The novel Monarch Butterfly SI approach uses multi-phase localization for sensor nodes. MBO is implemented and tested on several problem examples that are found in the literature.

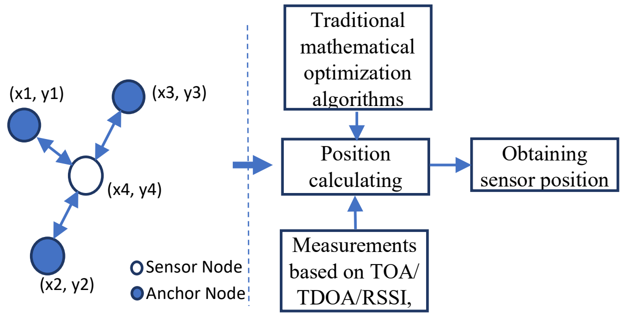

Figure 1. Anchor-based sensor nodes localization approach using traditional mathematical optimization algorithm WSNs

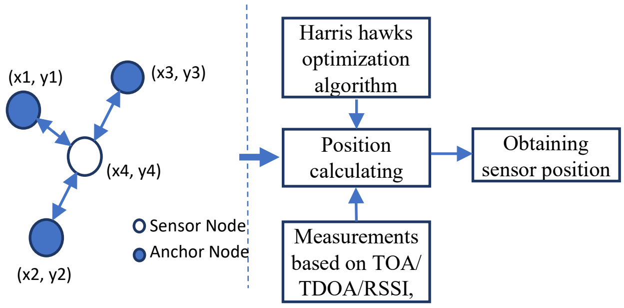

Figure 2. Anchor-based sensor nodes localization approach using harris hawks optimization algorithm WSNs

Experimental result analysis of the proposed work from other approaches has been successfully presented and has shown considerable potential in terms of solving the NP-hard problem of WSN localization.

A location-aware Mobile Anchor (MA) uses path planning to optimize mobile nodes. The work of MA to traverse into the target region of interest to minimize localization error and maximize localization of the successful node. Alomari et al. [31] proposed two novel dynamic movement approaches that provide the obstacle avoidance path planning for mobile node localization in WSN. Movement planning of mobile nodes designed based on two SI-based algorithms, i.e., GWO and Whale Optimization Algorithm (WOA). Comparing this proposed approach to the snake-like and z-curve models, it has shown remarkable results in terms of localization ratio, localization accuracy, and computation time.

An Elephant Herring Optimization (EHO) algorithm is adopted by Strumberger et al. [32] to solve localization problems in WSN. New metaheuristic computational intelligence approach dealing with NP-hard problems to achieve a near-to-target coordination value. The purpose of this approach is for the localization of randomly deployed sensor nodes in the monitoring area. The implementation of EHO for node localization in a WSN and results in efficient metaheuristic approaches to deal with sensor nodes localization. The work presents a future direction of the EHO algorithm that can apply efficient solutions to the superset problem of node localization, i.e., the coverage problem in WSNs.

An improved version of metaheuristic algorithms, such as the tree development algorithm and the EHO algorithm, is proposed by Strumberger et al. [33] to solve the localization problem of WSNs. The improvement of the proposed algorithm is analyzed by varying the size of the sensor network from 25 to 150 target nodes. The state of the art of some SI algorithms is tested in comparison to the proposed algorithm. Simulation results indicate that the proposed algorithm achieves very efficient results in terms of accurate location estimation of the coordinate of the unknown sensor node.

A Distance Mapping Algorithm (DMA) is proposed by Tan et al. [34] to overcome the node localization problem in WSN. To detect node position with high accuracy using the estimation matrix, distance matrix, and optimized linear transformation function.GA is employed for the optimal detection coordinate value of nodes during the calculation of the proposed algorithm. The node localization approach was simulated using three anchor nodes by the researcher in the laboratory. The results of the proposed algorithm perform well in terms of localization accuracy and energy consumption other than the localization algorithm.

Current important works of literature in the field of anchor-based localization WSNs are based on various parameters such as authors' publication, design approach, the technique used, comparison approaches, target parameters, and simulation tools using computational intelligence algorithms, as shown in Table 1.

4. Proposed Model Formulation

The proposed work presented for sensor nodes location estimation challenges using an anchor-based localization approach with the computational intelligence algorithms. The localization proposed model formulation is further classified into a subsection of the proposed model, proposed flow chart, and proposed algorithm.

Proposed Model

The proposed model designed with the components of anchors node ((x1, y1), (x2, y2), (x3, y3)), sensor node (x4, y4), HHO is used as a computational intelligence algorithm and measuring techniques (RSSI) as the inputs for the positioning estimation of unknown sensor nodes. The traditional optimization-based localization model using GWO, SSA, and EO is depicted in Figure 1. The newly smart localization model for anchor-based localization using the HHO algorithm, as shown in Figure 2.

Proposed Flow Chart

The working principles of the proposed work are depicted in the form of the flow chart in Figure 3, which shows the flow control of a designed framework for sensor nodes localization in an anchors-based approach using HHO computational intelligence algorithms. The computational intelligence algorithms are used SSA, GWO, EO, and HHO algorithms to finding optimal localization.

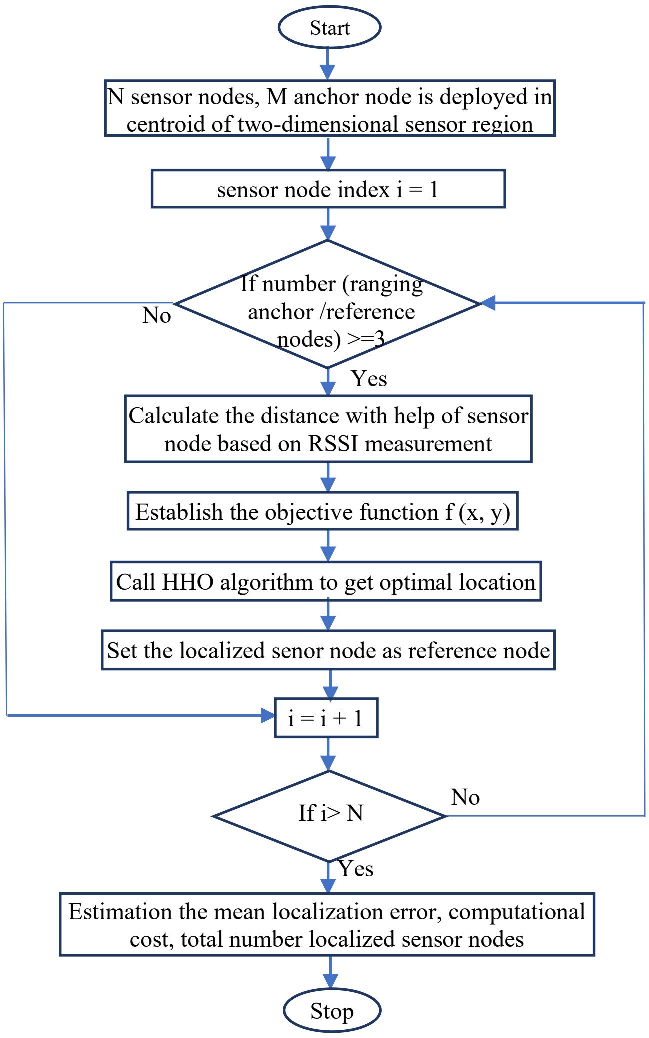

Figure 3. Flowchart of the WSN sensor nodes localization approach using the HHO computational intelligence algorithm.

Proposed Algorithm

The proposed work is designed for anchor-based localization using HHO computational intelligence algorithms. The algorithm for anchor-based localization of sensor nodes using the HHO algorithm is presented below:

Inputs:

$Target_{area}$ is a given target area where sensor nodes are to deploy randomly, l is a length and b is a breath of the target area, AN (x, y) in anchor nodes coordinate, centroid (a, b, c, d) is a function to calculate the centroid of the given area and a, b, c, d are the sides of the given target area, SN (x, y) is a current location of sensor nodes, SN_{total} is a total number of sensor nodes, dim is represent the dimensional of the target area, i is denoted the index of sensor nodes, SN_{ref} calculates the total number of anchor nodes are in their range, $dist_i$ is estimating the distance between sensor nodes and anchor nodes, the position is to save the best location of optimization algorithm in each iteration, Max_{iter} represents the maximum of iteration to position refinement, SearchAgent is agents are required to finding an optimal position, lb is a lower bound and ub is an upper bound of the given target area.

```
Begin:
    Target_area= l * b
    AN (x, y) = centroid (a, b, c, d)
    SN (x, y)= Taget_area * rand (SN_total,dim)
    for i =1 to SN_total
    do
        SN_ref =RSSI_recvied(AN)
        If (size (SN_ref)<= three))
        then
            Distance between anchor nodes and
            sensor node is calculated using the below
            equation:
            dist_i =√((x_t − x )² + (y_t − y)²)
            Estimate the coordinate value of SN (x,
            y,z) using below equations:
            let's z=0 for two-dimensional area
            (x–x1)²+ (y–y1)² + (z–z1)²= dist1²
            (x–x2)²+ (y–y2)² + (z–z2)²= dist2²
            (x–x3)²+ (y–y3)² + (z–z3)²= dist3²
            Call Harris Hawks Optimization
            computational intelligence algorithm:
            Initialize the random population
            Positions=initialization
            (SearchAgents_no, dim, ub, lb)
            while (1 < Max_Iter)
            do
            Update the position of search agents in
            the exploration phase using escaping
            energy of prey |E|.
            End while
        End if
    End For
END
```

Outputs:

Number of localized sensor nodes, mean localization error, and computational cost

5. Simulation Results and Analysis

Performance analysis of the proposed HHO algorithm along with comparative analysis of SSA, GWO, and EO algorithms in an anchor-based localization approach. The performance is analyzed with the help MatLab tool

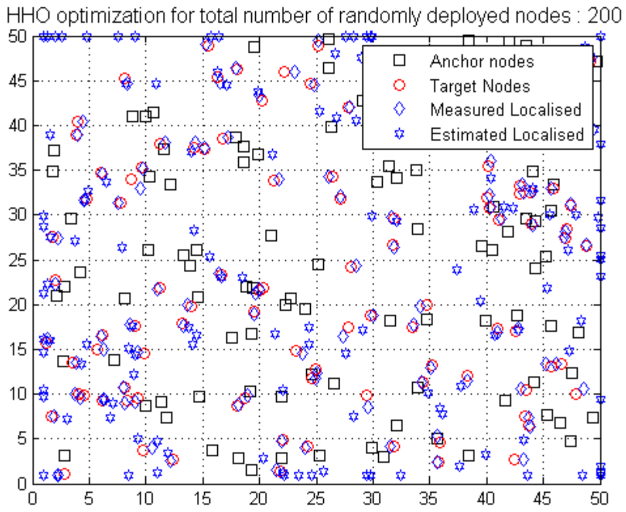

Figure 4. HHO algorithm for randomly deployed sensor nodes located in the target area.

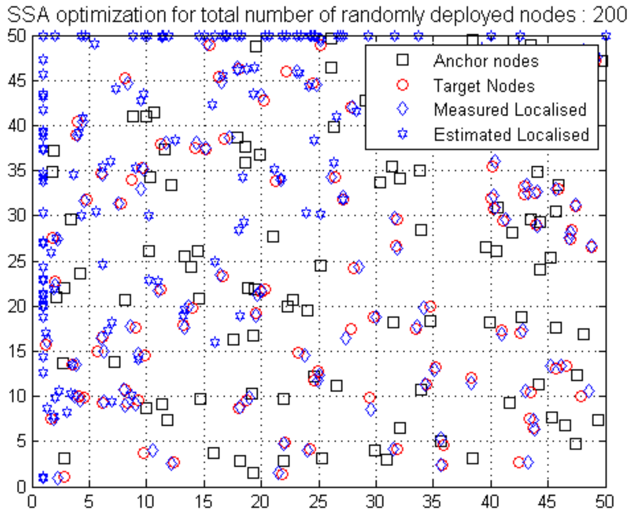

Figure 5. SSA algorithm for randomly deployed sensor nodes located in the target area

on the PC with an intel core i7 processor, 3.40 GHz CPU, and 4 GB RAM. This section is divided into two parts such as the simulation scenario and performance evaluation criteria.

Simulation Scenario

In the simulation configuration, the transmission range of anchor and sensor nodes is fixed at 20 m. The random deployment of sensor nodes in the target area of 50 x 50 m^2. Each simulation setup of up to 100 has randomly deployed anchor nodes in the target region with a variation of 10, and a free space path loss & fading model are considered. The RSSI measurement technique is used to distance estimation between sensor nodes and the anchor node in a range-based localization approach. The optimization algorithms are taken by SSA, GWO, EO, and HHO to the simulation of a single localization approach. In optimization algorithms, the search agents are ten and

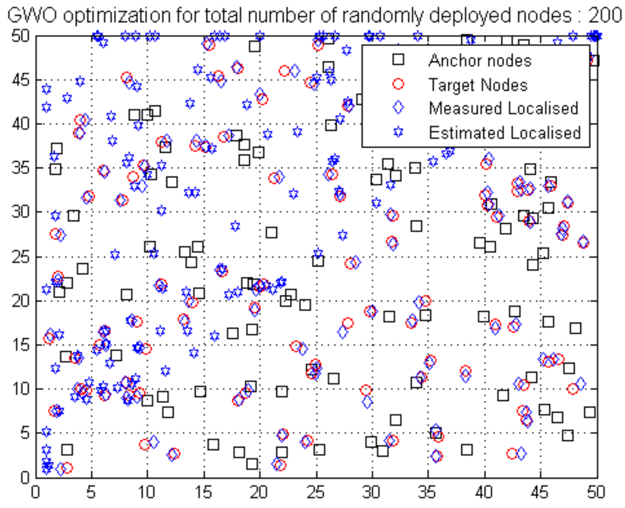

Figure 6. GWO algorithm for randomly deployed sensor nodes located in the target area.

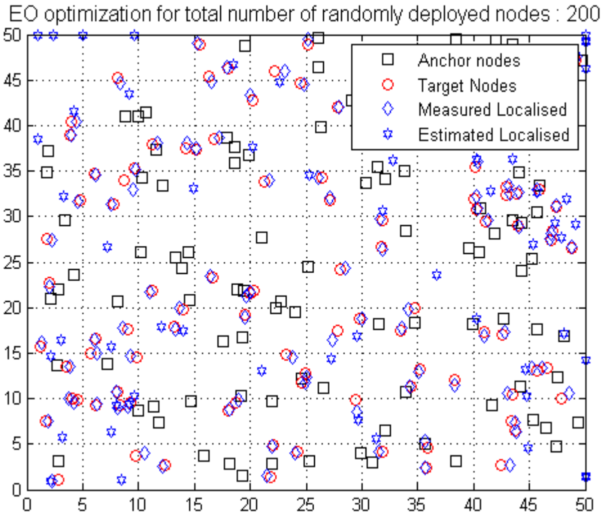

Figure 7. EO algorithm for randomly deployed sensor nodes located in the target area.

the maximum iteration is set 10 times for estimated position refinement.

Performance Evaluation Criteria

The performance evaluation criteria for the anchors-based localization approach using the HHO algorithm are mean localization error, computation cost, and the number of sensors localized with the variation of the number of randomly deployed sensor nodes. The number of randomly deployed anchor nodes by varying from 10 to 100, with a difference of 10 in each simulation. The anchor-based localization approach using HHO, SSA, GWO, and EO algorithms shown in Figure 4, Figure 5, Figure 6, and Figure 7 for randomly deployed of 200 sensor nodes.

Table 2. Minimum and maximum mean localization error of computation intelligence algorithms

Computational intelligence algorithm	Minimum value (m)	Maximum value (m)
HHO	0.8703	1.9835
SSA	1.5882	2.5719
GWO	1.1399	1.9756
EO	3.8334	18.2536

Table 3. Minimum and maximum computational cost of computational intelligence algorithms

Computational intelligence algorithm	Minimum value (sec)	Maximum value (sec)
HHO	120.0025	184.5612
SSA	123.5735	223.5646
GWO	136.2160	226.8486
EO	117.7639	3408.506

Table 4. Minimum and maximum number of localized nodes of computation intelligence algorithms

Computational intelligence algorithm	Minimum value	Maximum value
HHO	100	173
SSA	100	167
GWO	100	155
EO	32	107

- Mean Localization Error:
  The average difference between actual sensor nodes and estimated sensor nodes coordinate values. The mean localization error for sensor node localization for each randomly deployed anchor node from the variation 10 to 100 with a difference of 10 is shown in Table 2 and Figure 8. The resultant graph shows that the HHO algorithm is much better than the SSA, GWO, and EO algorithms for the anchors-based localization approach.
- Computational Cost:
  The total time is required to complete the localization process for randomly deployed sensor nodes is known as computation cost, and it is generally measured in terms of seconds (sec) unit. The computational cost of anchor-based localization using the HHO algorithm approximates better compared to SSA, GWO, and EO algorithms. By variation of 10 to 200 anchor nodes deployment with a difference of 10, the computation cost is calculated as shown in Table 3 and Figure 9.

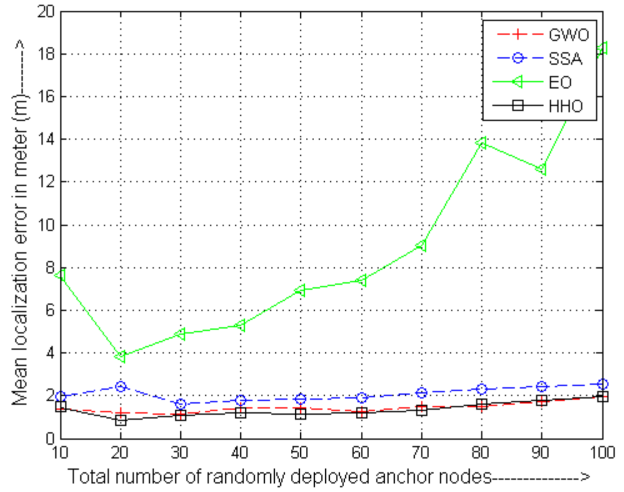

Figure 8. The mean localization error required for sensor nodes localized in the target area

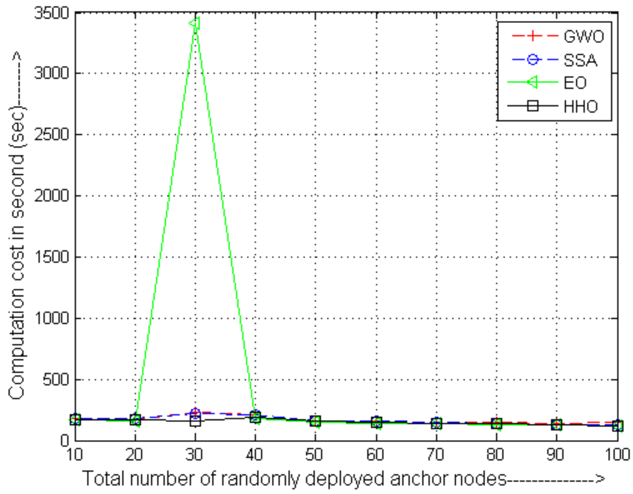

Figure 9. The total computational cost required for sensor nodes localized in the target area.

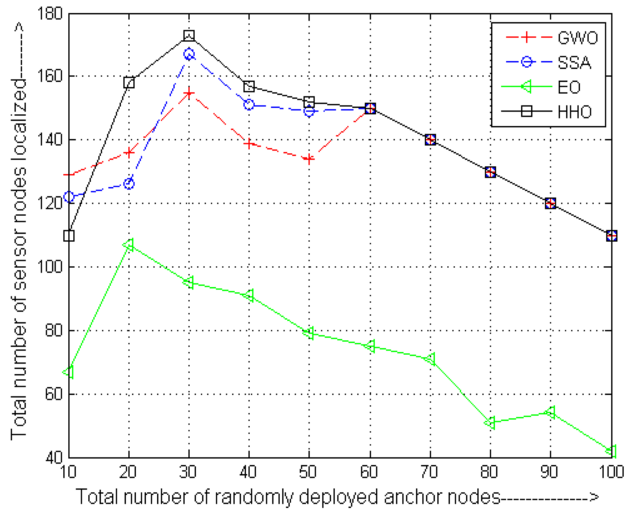

Figure 10. The total number of sensor nodes localized in the target area.

- Number of Localized Nodes:
  The number of localized sensor nodes over the number of randomly deployed anchor nodes by the variation of 10 to 100 sensor nodes with a difference of 10. The number of localized sensor nodes in an anchor-based localization approach using the HHO algorithm performs better than SSA, GWO, and the EO algorithm is shown in Table 4 and Figure 10.

6. Conclusion

The sensor node's localization became a crucial challenge for WSN. The technology advancement leads to WSN-IoT integration in order to reduce human intervention. Reduce the extra cost of GPS components is also minimized using an anchor-based localization approach. The optimal coordinate value calculation of the sensor nodes is done using the newly designed HHO algorithm in this paper. The simulated results and analysis of the HHO algorithm are compared with the SSA, GWO, and EO algorithms in an anchors-based localization approach. The percentage improvement of the HHO algorithm for localization problems over the SSA, GWO, and EO in terms of mean localization error, computational cost, and the number of localization nodes is presented in Table 5.

Table 5. Percentage improvement of the HHO algorithm over other algorithms.

	SSA	GWO	EO
Mean localization error	45.46 %	8.4 %	672.03 %
Computational Cost	13.97 %	19.2 %	1057.75 %
Number of localization nodes	2.2 %	6.6 %	49.08 %

From two newly designed algorithms i.e., EO and HHO algorithm in which EO algorithm failed to solve localization problem. Table 5 shows the HHO algorithm's overall performance analysis parameters for the efficiently estimated location of sensor nodes in WSN compared to other computational algorithms. The future direction of this proposed work can be implemented for the three-dimensional target area.

References

[1] Balaji, S., Nathani, K., & Santhakumar, R. (2019). IoT technology, applications, and challenges: a contemporary survey. Wireless personal communications, 108(1), 363-388.

[2] Vikram, N., Harish, K. S., Nihaal, M. S., Umesh, R., Shetty, A., & Kumar, A. (2017, January). A low cost home automation system using Wi-Fi based wireless sensor network incorporating Internet of Things (IoT). In 2017 IEEE 7th International Advance Computing Conference (IACC) (pp. 174-178). IEEE.

[3] Bhatt, J. G., Jani, O. K., & Bhatt, C. B. (2020). Automation Based Smart Environment Resource Management in Smart Building of Smart City. In Smart Environment for Smart Cities (pp. 93-107). Springer, Singapore.

[4] He, J., Rong, J., Sun, L., Wang, H., Zhang, Y., & Ma, J. (2020). A framework for cardiac arrhythmia detection from IoT-based ECGs. World Wide Web, 1-16.

[5] Jiang, H., Zhou, R., Zhang, L., Wang, H., & Zhang, Y. (2019). Sentence level topic models for associated topics extraction. World Wide Web, 22(6), 2545-2560.

[6] Qin, Y., Sheng, Q. Z., Falkner, N. J., Dustdar, S., Wang, H., & Vasilakos, A. V. (2016). When things matter: A survey on data-centric internet of things. Journal of Network and Computer Applications, 64, 137-153.

[7] Bajaj, K., Sharma, B., & Singh, R. (2020). Integration of WSN with IoT Applications: A Vision, Architecture, and Future Challenges. In Integration of WSN and IoT for Smart Cities (pp. 79-102). Springer, Cham.

[8] Kaur, A., Gupta, G. P., & Mittal, S. (2020). Impact of Nature-Inspired Algorithms on Localization Algorithms in Wireless Sensor Networks. In Nature-Inspired Computing Applications in Advanced Communication Networks (pp. 1-18). IGI Global.

[9] Sharma, R., & Prakash, S. (2018, October). Emerging Trends in Localization Techniques for WSNs: A Review. In 2018 International Conference on Advances in Computing, Communication Control and Networking (ICACCCN) (pp. 1119-1124). IEEE.

[10] Sharma, R., & Prakash, S. (2019, March). Latest Trends and Future Directions of Localization Algorithms in Wireless Sensor Networks. In 2019 6th International Conference on Computing for Sustainable Global Development (INDIACom) (pp. 626-631). IEEE.

[11] Stoleru, R., He, T., & Stankovic, J. A. (2007). Range-free localization. In Secure Localization and Time Synchronization for Wireless Sensor and Ad Hoc Networks (pp. 3-31). Springer, Boston, MA.

[12] Stoleru, R., He, T., & Stankovic, J. A. (2007). Range-free localization. In Secure Localization and Time Synchronization for Wireless Sensor and Ad Hoc Networks (pp. 3-31). Springer, Boston, MA.

[13] Adewumi, O. G., Djouani, K., & Kurien, A. M. (2013, February). RSSI based indoor and outdoor distance estimation for localization in WSN. In 2013 IEEE international conference on Industrial technology (ICIT) (pp. 1534-1539). IEEE.

[14] Ravindra, S., & Jagadeesha, S. N. (2014). Time of arrival based localization in wireless sensor networks: A linear approach. arXiva preprint arXiv:1403.6697.

[15] Rong, P., & Sichitiu, M. L. (2006, September). Angle of arrival localization for wireless sensor networks. In 2006 3rd annual IEEE communications society on sensor and ad hoc communications and networks (Vol. 1, pp. 374-382). IEEE.

[16] Xiong, H., Chen, Z., Yang, B., & Ni, R. (2015). TDOA localization algorithm with compensation of clock offset for wireless sensor networks. China Communications, 12(10), 193-201.

[17] Xiao, H., Zhang, H., Wang, Z., & Gulliver, T. A. (2017, August). An RSSI based DV-hop algorithm for wireless sensor networks. In 2017 IEEE Pacific Rim Conference on Communications, Computers and Signal Processing (PACRIM) (pp. 1-6). IEEE.

[18] Blumenthal, J., Reichenbach, F., & Timmermann, D. (2005, March). Position estimation in ad hoc wireless sensor networks with low complexity. In Joint 2nd workshop on positioning, navigation and communication (pp. 41-49).

[19] Zhou, Z. D., Hu, P., Liu, Q., & Li, F. M. (2007). MDS-based fast localization algorithm for wireless sensor networks. Chinese Journal of Sensors and Actuators, 20(10), 2303-2307.

[20] Albowicz, J., Chen, A., & Zhang, L. (2001, November). Recursive position estimation in sensor networks. In Proceedings Ninth International Conference on Network Protocols. ICNP 2001 (pp. 35-41). IEEE.

[21] Sharma, R., & Prakash, S. (2020). An adaptive ensemble localization approach for sensor nodes in WSN-IoT. EAI Endorsed Transactions on Energy Web, 7(29).

[22] Sharma, R., & Prakash, S. (2020). Eurasian Wolves-Cuckoo Search Optimizer for Localization of Mobile Sensor Nodes using Single Beacon Node in WSN.EAI Endorsed Transactions on Scalable Information Systems, 7(28).

[23] Heidari, A. A., Mirjalili, S., Faris, H., Aljarah, I., Mafarja, M., & Chen, H. (2019). Harris hawks optimization: Algorithm and applications. Future generation computer systems, 97, 849-872.

[24] Mirjalili, S., Gandomi, A. H., Mirjalili, S. Z., Saremi, S., Faris, H., & Mirjalili, S. M. (2017). Salp Swarm Algorithm: A bio-inspired optimizer for engineering design problems. Advances in Engineering Software, 114, 163-191.

[25] Mirjalili, S., Mirjalili, S. M., & Lewis, A. (2014). Grey wolf optimizer. Advances in engineering software, 69, 46-61.

[26] Faramarzi, A., Heidarinejad, M., Stephens, B., & Mirjalili, S. (2020). Equilibrium optimizer: A novel optimization algorithm. Knowledge-Based Systems, 191, 105190.

[27] Kanoosh, H. M., Houssein, E. H., & Selim, M. M. (2019). Salp swarm algorithm for node localization in wireless sensor networks. Journal of Computer Networks and Communications, 2019.

[28] Rajakumar, R., Amudhavel, J., Dhavachelvan, P., & Vengattaraman, T. (2017). GWO-LPWSN: Grey wolf optimization algorithm for node localization problem in wireless sensor networks. Journal of Computer Networks and Communications, 2017.

[29] Tuba, E., Tuba, M., & Beko, M. (2018). Two stage wireless sensor node localization using firefly algorithm. In Smart trends in systems, security and sustainability (pp. 113-120). Springer, Singapore.

[30] Strumberger, I., Tuba, E., Bacanin, N., Beko, M., & Tuba, M. (2018, April). Monarch butterfly optimization algorithm for localization in wireless sensor networks. In 2018 28th International Conference Radioelektronika (RADIOELEKTRONIKA) (pp. 1-6). IEEE.

[31] Alomari, A., Phillips, W., Aslam, N., & Comeau, F. (2018). Swarm intelligence optimization techniques for obstacle-avoidance mobility-assisted localization in wireless sensor networks. IEEE Access, 6, 22368-22385.

[32] Strumberger, I., Beko, M., Tuba, M., Minovic, M., & Bacanin, N. (2018, May). Elephant herding optimization algorithm for wireless sensor network localization problem. In Doctoral Conference on Computing, Electrical and Industrial Systems (pp. 175-184). Springer, Cham.

[33] Strumberger, I., Minovic, M., Tuba, M., & Bacanin, N. (2019). Performance of Elephant Herding Optimization and Tree Growth Algorithm Adapted for Node Localization in Wireless Sensor Networks. Sensors, 19(11), 2515.

[34] Tan, R., Li, Y., Shao, Y., & Si, W. (2019). Distance Mapping Algorithm for Sensor Node Localization in WSNs. International Journal of Wireless Information Networks, 1-10.