IEEE Access
Multidisciplinary : Rapid Review : Open Access Journal

Received 25 April 2024, accepted 14 June 2024, date of publication 20 June 2024, date of current version 27 June 2024.

Digital Object Identifier 10.1109/ACCESS.2024.3417227

RESEARCH ARTICLE

Enhancing Node Localization Accuracy in Wireless Sensor Networks: A Hybrid Approach Leveraging Bounding Box and Harmony Search

MOHAMMED OMARI[1], (Member, IEEE), MOHAMMED KADDI[2], KHOULOUD SALAMEH[1], ALI ALNOMAN[1], KHADRA ELFATMI[2], AND FADILA BAARAB[2]

[1]Computer Science and Engineering Department, American University of Ras Al Khaimah, Ras Al Khaimah, United Arab Emirates
[2]LDDI Laboratory, Mathematics and Computer Science Department, Ahmed Draia University of Adrar, Adrar 01025, Algeria

Corresponding author: Mohammed Omari (mohammed.omari@aurak.ac.ae)

This work was supported by the Mohammed Bin Rashid Smart Learning Program (United Arab Emirates) under Grant MBRSLP/06/23.

ABSTRACT Accurate node localization in Wireless Sensor Networks (WSNs) is crucial for applications like environmental monitoring and military surveillance. Traditional localization methods, such as trilateration, often struggle with accuracy due to signal attenuation and environmental obstructions, leading to significant localization errors in practical scenarios. This paper aims to address the limitations of traditional localization methods by developing and evaluating a hybrid localization method that enhances accuracy and robustness in node localization within WSNs. The proposed method integrates the Bounding Box approach with the Harmony Search optimization algorithm, resulting in the Bounding Box Harmony Search (BBHS) method. The BBHS method utilizes the initial geometric constraints provided by the Bounding Box approach and refines these estimates using the global optimization capabilities of the Harmony Search algorithm. Simulation results, obtained using a custom-developed WSN localization simulator, demonstrate that the BBHS method significantly reduces localization errors compared to traditional trilateration and the standalone Bounding Box method. The BBHS method consistently provides enhanced accuracy and robustness across varying network conditions, highlighting its effectiveness in practical deployment scenarios. The advancements presented in this paper suggest that hybrid methods like BBHS represent a significant step forward in WSN localization technologies. By combining geometric constraints with optimization processes, the BBHS method overcomes the drawbacks of earlier techniques, paving the way for more reliable and resilient WSN operations.

INDEX TERMS Wireless sensor networks (WSNs), node localization, trilateration, bounding box method, harmony search algorithm, localization accuracy.

I. INTRODUCTION

The development of localization techniques for WSNs is an ongoing area of research, driven by the expanding scope of network applications and the continuous quest for better performance. Innovations in this field seek to address the challenges posed by varying environmental conditions, node mobility, and the inherent limitations of sensor hardware. As these technologies evolve, the emphasis is on creating scalable, resilient, and accurate localization solutions that can support the growing complexity and diversity of network deployments [1]. Node localization is a critical and complex component of wireless sensor networks (WSN), encompassing a range of techniques from trilateration and triangulation to hybrid methods that blend the best of range-based and range-free approaches. The choice of localization method

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

significantly impacts the network's performance, applicability, and operational costs. As the demand for WSNs grows across different sectors, so does the need for advanced localization techniques that can provide accurate and efficient positioning in diverse and challenging environments.

Node localization in wireless sensor networks plays a pivotal role in the functionality and operational efficiency of these networks. It entails determining the positions of nodes within the network, which is crucial for tasks ranging from environmental monitoring to precision agriculture, military operations, and emergency response (Fig. 1). The essence of node localization is to facilitate accurate data collection, target tracking, and network management, among other applications. Given the diverse environments and applications of WSNs, localization techniques must be robust, accurate, and resource-efficient.

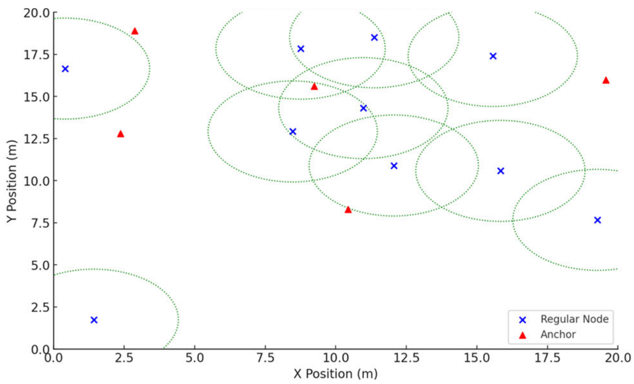


FIGURE 1. A wireless sensors network or regular nodes and anchors.

Localization techniques in WSNs are divided into two main categories: range-based and range-free methods. Range-based localization methods depend on the measurement of physical properties such as distance or angle between nodes [2]. These methods include trilateration, triangulation, time of arrival, time difference of arrival, angle of arrival, and received signal strength indicator (RSSI). Each method has its unique requirements and applications, with trilateration and triangulation being among the most prevalent due to their simplicity and effectiveness in certain contexts. Trilateration is a widely utilized range-based localization technique that determines the position of an unknown node by measuring its distance from at least three known points. This method is particularly effective in environments where the signal strength can accurately indicate the distance between nodes. By employing advanced signal processing techniques and optimizing the placement of reference nodes, the accuracy of trilateration-based localization can be significantly enhanced [3]. Research in this area focuses on minimizing the error margin and improving the reliability of distance measurements. Triangulation, another range-based method, calculates the position of an unknown node using the angles of arrival from at least two known points. Unlike trilateration, triangulation relies on geometric principles rather than distance measurements, making it suitable for scenarios where measuring distance is challenging. Studies leveraging triangulation strive to refine angle measurement techniques and develop algorithms that can efficiently process angular information to pinpoint node locations accurately.

On the other hand, range-free localization methods do not require precise measurements of distance or angle [2]. Instead, these methods use connectivity information and relative proximity between nodes to estimate their positions. Range-free techniques are often preferred in large-scale deployments or applications where high accuracy is not paramount. These methods are more cost-effective and consume less energy compared to range-based techniques, making them suitable for resource-constrained environments. The selection between range-based and range-free methods depends on several factors, including the desired accuracy, environmental conditions, and resource availability. While range-based methods generally offer higher precision, they demand additional hardware and computational resources. This makes them more expensive and energy-intensive. In contrast, range-free methods are less precise but more economical and energy-efficient, catering to the needs of extensive networks where moderate accuracy suffices. Hybrid localization methods have emerged, combining range-based and range-free techniques to exploit the advantages of both approaches. These hybrid strategies aim to achieve a balance between accuracy and resource efficiency. By integrating machine learning algorithms, these methods optimize localization performance based on the available data and network conditions. Such approaches are gaining traction, offering adaptability and improved efficiency in dynamic environments.

This paper makes several key contributions to the field of WSN localization:

- We propose a novel hybrid localization method, the Bounding Box Harmony Search (BBHS) method, which combines geometric constraints with optimization techniques.
- We develop a comprehensive WSN localization simulator that allows for the detailed evaluation of localization methods under various network conditions.
- We conduct extensive simulations to demonstrate the improved accuracy and robustness of the BBHS method compared to traditional trilateration and the standalone Bounding Box method.
- We provide a thorough analysis of the performance metrics and the impact of different network parameters on localization accuracy.

The remainder of this paper is organized as follows. Section II reviews existing literature on WSN localization methods. Section III describes the implemented WSN localization methods, including the Trilateration Technique, Bounding Box, and Harmony Search. Section IV details the development of a network simulator for node localization, including the design, conceptual framework, and simulator interface. Section V presents the experimental results and analysis, including comparisons of localization errors under various conditions and the fine-tuning of Harmony Search hyperparameters. Section VI concludes the paper with a summary of findings and future research directions.

II. LITERATURE REVIEW

The problem of range measuring errors is a major concern in wireless node localization and was addressed by Ramadurai and Sichitiu [4]. While research on localization in wireless sensor networks has been conducted, the issues arising from inaccurate range measurements have received less attention. They handled these errors using a probabilistic methodology. To estimate the locations of sensor nodes, it makes use of Received Signal Strength Indicator (RSSI) readings, a widely used metric in wireless communication. Errors can always occur in RSSI data because of things like multipath fading and interference. To calculate RSSI as a function of distance, data of signal strength at different distances was collected, using mobile devices with wireless cards installed. Measurements were made to capture the effect of environmental elements from various positions and orientations. The fact that probability distributions are visible in RSSI measurements is one important finding from the data collection procedure. These distributions frequently exhibit characteristics of a normal distribution. Ramadurai and Mihail emphasize that position estimate is based on these probability distributions. Key factors in the localization process are the mean and standard deviation of these distributions for each measurement of the signal strength. Experimental evaluations were conducted based on unknown nodes and beacons in an actual outdoor setting. These tests show that the algorithm can produce reliable position estimations even with errors in the range measurement.

Tran and Nguyen [5] addressed the difficulty of localizing nodes in wireless sensor networks using Support Vector Machines (SVM). They stress how impractical centralized methods for large-scale networks can be because of the high processing and communication costs. As a response, they suggested a distributed approach to sensor location estimation, taking into account the resource constraints of low-cost sensor devices in WSNs. LSVM (Localization with Support Vector Machines) presents a different approach for node localization that is based on connectedness rather than signal strength. By assuming that nodes can connect with beacon nodes via a multi-hop connection, the method enables a more scalable solution appropriate for bigger networks. The LSVM algorithm estimates the location of beacon nodes using support vector machines with a radial basis function kernel. The authors used a decision tree for localization in both x and y dimensions. The error analysis acknowledged that SVM and LSVM are subject to errors, but LSVM's error is formulated under the effect of SVM. The paper presents a modified Mass-Spring Optimization (MSO) technique to improve LSVM. This technique modifies node locations based on a "spring force" calculated from neighboring nodes' positions.

The results showed that LSVM outperforms Diffusion and AFL (Anchor-Free Localization) in terms of accuracy and efficiency, even in networks with coverage holes. LSVM's improved performance is attributed to its reduced traffic generation and better error distribution across sensors.

TABLE 1. List of acronyms and their meanings.

Acronym	Meaning
AFL	Anchor-Free Localization
AMG	Adaptation Multi-Group
ANN	Artificial Neural Network
APIT	Area Point-in-Triangulation
APT	Accuracy-Priority Trilateration
BAT	Bats
BB	Bounding Box
BBHS	Bounding Box first, Harmony Search second
BBO	Biogeography-Based Optimization
CLPSO	Comprehensive Learning Particle Swarm Optimizer
CRB	Cramér-Rao bound
CRL	Cosine Rule-based Localization
CS	Cuckoo Search
CSO	Cat Swarm Optimization
DE	Differential evolution
DEA	Differential evolution algorithm
DMA	Distance Mapping Algorithm
DV-Hop	Distance Vector Hop
ELM	Extreme Learning Machine
FA	Firefly Algorithm
FL	Fuzzy Logic
GA	Genetic Algorithms
GADV	Genetic Algorithm – Distance Vector
GPS	Global Positioning System
GWO	Grey Wolf Optimization
HPPSO	Hybrid Discreate PSO
HS	Harmony Search
HSBB	Harmony Search first, Bounding Box second
HSPO	Hybrid Particle Swarm Optimization
HWSN	Heterogeneous Wireless Sensor Networks
ISSA	Improved Sparrow Search Algorithm
KELM-HQ	Kernel Extreme Learning Machines based on Hop-Count Quantization
LMAT	Localization with Mobile Anchor Node based on Trilateration
LoRaWAN	Low Power Wide Area Networking
LSVM	Localization with Support Vector Machines
MANET	Mobile Ad hoc Networks
MBA	Modified Bat Algorithm
MCB	Monte Carlo localization boxing
MDS	Multidimensional Scaling
ML	Maximum Likelihood
MLP	Multilayer Perceptron
MSO	Mass-Spring Optimization
NLA_MB	Node Localization Algorithm Mobile Beacon
NS-IPSO	Node Segmentation to improve PSO
ODE	Opposition-Based Differential Evolution
OLSL	Optimization-Based Least Square Localization
PCCSO	Parallel Compact Cat Swarm Optimization
PSO	Particle Swarm Optimization
QA	Quantum Annealing
QABA	Quantum Annealing Bat Algorithm
QUATRE	Quasi-Affine Transformation Evolution Algorithm
RBF	Radial Basis Function
RIM	Range Information Matrix
RMSE	Root Mean Square Error
RNN	Recurrent Neural Networks
RSSI	Received signal strength indicator
SA	Simulated Annealing
SLPSO	Social learning particle swarm optimization
SSA	Salp Swarm Algorithm
SVM	Support Vector Machines
UAV	Unmanned Aerial Vehicle
WSN	Wireless Sensor Networks

Gopakumar and Lillykutty Jacob in their paper, "Localization in wireless sensor networks using particle swarm optimization" [6], introduce a pioneering approach to WSN localization, employing PSO. Unlike traditional gradient search-based methods, PSO mitigates the risk of local minima, ensuring superior convergence characteristics. Operating within a centralized architecture, their algorithm optimizes distance measurements between anchor and target nodes in the WSN. By minimizing mean squared range error, it aims to estimate target node coordinates. The iterative nature of PSO allows for updates to particle positions representing target nodes, leveraging individual and population-wide best positions. Factors like anchor node density, transmission range, and range measurement standard deviation are systematically considered. Experimental findings reveal insights such as improved localization with increased anchor node density and transmission range. PSO consistently outperforms Simulated Annealing, particularly regarding convergence and overall performance. This comparative analysis underscores the efficacy of PSO in WSN localization, demonstrating its superiority over traditional methods.

Moses et al. discuss the derivation of a maximum likelihood (ML) estimator for sensor node location and orientation parameters in the context of self-calibration [7]. The ML algorithm involves iterative minimization of a cost function, solving nonlinear equations for unknown parameters, including nuisance parameters. The Fisher information matrix computes the Cramér-Rao bound (CRB) for parameter variance, serving as a benchmark for evaluating self-localization algorithms. While computational complexity isn't discussed, it's inferred to depend on factors like node and source numbers. Location uncertainty indirectly impacts separation between nodes and sources, influencing average uncertainty. Adding more sources improves accuracy and decreases uncertainty. Benefits of the method include versatility, flexibility in signal sources, and accurate estimation of node locations and orientations. Drawbacks include reliance on assumptions and factors affecting effectiveness and accuracy. Potential future research directions include scalability evaluation, impact of environmental factors, comparison with other methods, extension to dynamic networks, and integration of additional information sources to enhance accuracy and reliability. The paper concludes with suggestions for further investigation into these areas to advance self-localization techniques in wireless sensor networks.

In their paper "A soft computing approach to localization in wireless sensor networks" [8], Yun et al. propose two intelligent localization schemes utilizing received signal strength intensity (RSSI) from anchor nodes, demonstrating range-free localization for wireless sensor networks. These schemes address the simplicity and cost-effectiveness of range-free methods but acknowledge their lower accuracy due to the absence of angle or distance information from anchor nodes. Employing soft computing techniques to overcome these drawbacks, the proposed schemes require no complex hardware. Assumptions include positionally unaware sensor nodes, preconfigured or GPS-located anchor nodes, uniform transmission ranges, and perfectly spherical radio propagation. The first scheme divides localization into discrete problems, considering each anchor node's edge weight independently, modeling these weights with fuzzy logic, and optimizing them with a genetic algorithm (GA). This approach enhances localization precision but may still exhibit errors inherent to range-free methods and entail computational complexity. The second scheme employs a neural network to approximate sensor location mapping from anchor node signals, treating localization as a single problem and learning input-output relationships. Despite simplifying the localization process and achieving accurate results, challenges include precision issues compared to range-based techniques and the computational demands of network design and training. Both schemes outperform current approaches in simulations and outdoor experiments, suitable for large-scale networks due to autonomous sensor node positioning. Future research should focus on minimizing training time and adapting techniques to noisy indoor environments. Additionally, the paper clarifies the roles of localization beacons and anchor nodes in WSNs, noting variations in terminology and usage across literature.

In their article "Improved DV-Hop node localization algorithm in wireless sensor networks" [9], Chen and Zhang introduce enhancements to overcome limitations of the original DV-Hop algorithm, focusing on precision and reliability in range-free node localization. The study emphasizes the criticality of accurate sensor location for generating valid network information in WSNs. Key adaptations include strategic placement of anchor nodes, dispersing them around monitoring regions to improve global network consideration and enhance location accuracy. Additionally, the algorithm modifies the computation of unknown node average one-hop distances, integrating weights based on least squares error considerations for improved accuracy. Classical positioning techniques are replaced with a two-dimensional hyperbolic localization algorithm, providing more precise node location estimations. Moreover, PSO is incorporated to adjust locations computed by the hyperbolic algorithm, balancing precision and computation efficiency. Simulation results demonstrate the superiority of the enhanced DV-Hop algorithm over the original and other variants, with significantly reduced localization error and variance, enhancing precision and stability. While the inclusion of PSO slightly increases computation time, it offers a more reliable mechanism for node localization in real-life scenarios. Overall, the proposed modifications represent significant progress in WSNs, improving precision and reliability crucial for accurate information collection, and highlighting the importance of balancing accuracy with computational efficiency in practical WSN applications.

In their paper "Node localization in wireless sensor networks using butterfly optimization algorithm" [10], Arora and Singh present a node localization scheme employing the butterfly optimization algorithm, crucial for enhancing network performance in wireless sensor networks. The scheme's efficacy is compared against established methods such as PSO and firefly algorithm (FA), with the butterfly optimization algorithm demonstrating superior and more consistent node localization accuracy. The article offers a concise review of nature-inspired metaheuristic algorithms like PSO, GA, and the butterfly optimization algorithm, outlining the iterative process of node localization involving initialization, distance estimation, and position estimation using optimization algorithms. Simulation results reveal that the butterfly optimization algorithm achieves better accuracy and faster computation times compared to PSO and FA. Localization performance is influenced by factors including anchor node density, transmission range, and the number of iterations, with increased density and range correlating with improved accuracy and localization of more nodes. Furthermore, more iterations lead to enhanced accuracy. The proposed scheme outperforms alternative algorithms in terms of both accuracy and computation time, offering a comprehensive analysis of its performance across diverse scenarios.

In their paper "A new fuzzy logic-based node localization mechanism for wireless sensor networks" [11], Amri et al. propose a node localization mechanism utilizing the butterfly optimization algorithm, which differs from purely range-based or range-free methods by incorporating RSSI information into a fuzzy inference system for node position estimation. The mechanism integrates multi-hop routing protocols and a hierarchical communication model to enhance geographic routing precision in WSNs. Through a cluster-based approach, individual clusters are formed, with member nodes communicating with cluster heads for data aggregation and transmission to the base station. The technique measures flow between anchor and sensor nodes, estimating distances using RSSI and employing a weighted centroid formula for sensor node localization. Utilizing fuzzy Mamdani and Takagi-Sugeno-Kang inference systems enhances processing time and accuracy, with the Sugeno method replacing the laborious defuzzification procedure. Fuzzy logic is further employed to select the next cluster head, reducing energy dissipation and improving position estimation precision. Simulation results demonstrate the superiority of the proposed mechanism over alternative approaches, with reduced localization error and improved energy efficiency, validated across various scenarios. While the mechanism assumes precise anchor node positioning and stable wireless channels, its applicability to practical situations and potential complexity in implementation warrant further study. Nevertheless, the mechanism's performance underscores its significance in IoT applications, such as environmental monitoring, industrial automation, and medical tracking, where precise node localization is essential for optimizing data transfer and network longevity.

In their paper "An Effective Cuckoo Search Algorithm for Node Localization in Wireless Sensor Network" [12], Cheng and Xia highlight the growing significance of wireless sensor networks in various domains such as healthcare, transportation, and environmental monitoring. They emphasize the importance of accurate node localization in WSN applications for data differentiation, geographical routing, and power conservation. To address challenges like computational complexity and communication overhead that can hinder efficient localization, the authors propose a new variation of the Cuckoo Search (CS) algorithm tailored for node localization. This nature-inspired algorithm mimics the behavior of cuckoos and introduces modifications such as adjusting step size, using solutions' fitness to generate mutation probabilities, and limiting the range for population, aimed at optimizing global solution search while minimizing unnecessary computational expenditure. Through extensive simulations considering node and anchor numbers, as well as distance, the authors demonstrate the efficacy of the modified CS algorithm in reducing average localization error and improving localization success ratio compared to standard CS and PSO algorithms. The modified CS algorithm exhibits faster convergence and reduced localization error, particularly noticeable in initial generations, indicating its potential for enhancing WSN precision and energy efficiency. This improvement is crucial for applications relying on accurate location information and efficient performance, underscoring the significance of the modified CS algorithm for future WSN advancements.

Kumar et al. in their article "Meta-heuristic range-based node localization algorithm for wireless sensor networks" [13], delve into the integration of Hybrid PSO (HPSO) and Biogeography-Based Optimization (BBO) algorithms for enhancing node localization in wireless sensor networks. HPSO, derived from PSO, subdivides the swarm into smaller sub-swarms to improve accuracy and convergence speed, with each particle moving towards its personal best, global best, and the best position encountered by its sub-swarm. BBO, inspired by biological principles, utilizes a mathematical model based on the Habitat Suitability Index (HSI) to optimize solutions. The proposed algorithm combines these techniques, iteratively localizing nodes by estimating distances from neighboring nodes and anchors. Two case studies demonstrate the effectiveness of HPSO and BBO, with simulation results showing their superior performance compared to PSO in terms of accuracy and convergence speed. HPSO offers fast convergence and better accuracy, while BBO provides even higher accuracy at a slower convergence rate. The choice between HPSO and BBO depends on specific requirements, such as accuracy or fast convergence. Overall, the integration of HPSO and BBO offers improved accuracy, convergence speed, and energy conservation in WSNs, with implications for enhancing the capabilities of WSNs in various practical settings.

Chen et al. proposed the Node Localization Algorithm for Wireless Sensor Networks with Mobile Beacon Node (NLA_MB) in [14] to address challenges related to the maximum movement distance of mobile beacon nodes in WSNs. Aimed at improving sensor node localization accuracy, NLA_MB seeks to enhance the number of sojourn locations for the beacon node, increase the average number of anchor nodes for sensor nodes, and reduce the average localization error. In the context of multi-hop ad hoc networks, where accurate sensor node locations are crucial for various applications like smart grids, traditional satellite localization systems are deemed impractical due to high energy consumption. NLA_MB operates within a 2-dimensional area with static sensor nodes, anchor nodes, and a mobile beacon node equipped with GPS or Beidou satellite localization module. The algorithm employs a heuristic approach, based on virtual force theory, to optimize node localization errors while considering constraints such as movement path and distance. By dividing the search area into hexagonal grids and utilizing beacon node movement constrained by mathematical formulas, NLA_MB enables sensor nodes to estimate their coordinates based on received beacon node location information. Simulation results demonstrate NLA_MB's ability to cover more sensor nodes with limited beacon node movement distance, yielding higher anchor node numbers and improved localization accuracy across various node distributions. Despite considerations of communication overhead due to frequent location information exchange, NLA_MB presents a promising distributed solution for enhancing sensor node localization accuracy in WSNs with mobile beacon nodes.

Wu et al. present in [15] the "A Hybrid Mobile Node Localization Algorithm Based on Adaptive MCB-PSO Approach in Wireless Sensor Networks," offering a novel approach to node localization in dynamic wireless environments. By integrating PSO with Monte Carlo localization boxing (MCB), the method addresses challenges associated with locating nodes in three-dimensional mobile spaces. Emphasizing the importance of localization in wireless sensor networks, where energy-constrained smart sensors are widely deployed, the research underscores the need for accurate data representation amidst changing network conditions. The proposed MCB-PSO hybrid method introduces enhancements over traditional MCB by incorporating a random waypoint moving model to account for mobile node behavior and adapt to real-life scenarios. This modification improves the algorithm's ability to handle both known and unknown locations efficiently. Additionally, the inclusion of a novel anchor selection method enhances the effectiveness of the PSO component by dynamically adjusting the search strategy based on temporal and spatial considerations. The study evaluates the performance of the hybrid approach against existing localization methods like DV-Hop and Centroid through simulations, demonstrating its superiority in terms of accuracy, speed, and efficiency. The results highlight the effectiveness of the MCB-PSO hybrid algorithm in swiftly and accurately localizing mobile nodes in dynamic wireless environments, representing a significant advancement over traditional stationary network models.

In [16], Hao et al. introduce "A node localization algorithm based on Voronoi diagram and support vector machine for wireless sensor networks," addressing the challenge of accurately determining the locations of nodes within wireless sensor networks. Recognizing the significance of node localization in diverse applications such as healthcare, environmental monitoring, and industrial control, the paper identifies limitations in current localization methods, categorizing them as distance-based or not and advocating for improved efficiency and accuracy. To overcome these challenges, the authors propose a novel approach that combines Voronoi diagrams with SVM. Initially, Voronoi diagrams are employed to partition the sensor network area, providing an initial estimation of node locations and facilitating the management of localization complexity by dividing the space into smaller regions. Subsequently, SVM, a robust machine learning technique, is utilized to refine these estimations by leveraging data collected from the sensor network. This integration enables the algorithm to effectively identify node locations by capitalizing on SVM's pattern recognition capabilities. The paper underscores the importance of striking a balance between localization accuracy and resource constraints inherent in WSNs, highlighting the method's practicality in real-world scenarios. Experimental validation demonstrates the efficacy and flexibility of the proposed approach across diverse indoor and outdoor environments, substantiating its utility in challenging conditions. Ultimately, the study contributes significantly to WSN localization technologies, offering a robust solution that promises to enhance localization efficiency and accuracy across various applications. The comprehensive approach presented, from theoretical development to real-world testing, underscores the transformative potential of the proposed method and opens avenues for further research and innovation in WSN localization techniques.

In [17], Li et al. present "A parallel compact cat swarm optimization and its application in DV-Hop node localization for wireless sensor network," introducing the Parallel Compact Cat Swarm Optimization (PCCSO) algorithm as a solution to the convergence and memory consumption challenges associated with Cat Swarm Optimization (CSO). The PCCSO algorithm is designed to enhance local search capabilities while conserving computational memory, particularly focusing on its application in the DV-Hop node localization for wireless sensor networks. The authors demonstrate that incorporating PCCSO into DV-Hop improves localization accuracy and reduces memory usage compared to other DV-Hop-based optimization algorithms. The paper begins by discussing the widespread adoption of global optimization algorithms like CSO in various fields, highlighting CSO's applications in WSNs for tasks such as energy-aware routing and sensor configuration. However, CSO's significant memory consumption hampers its convergence speed and optimization accuracy, motivating the development of a more memory-efficient solution. The proposed PCCSO algorithm is described in detail, featuring a parallel communication strategy and a compact scheme to enhance its efficiency and effectiveness in optimization tasks. Experimental results, conducted using CEC2013 benchmark functions, demonstrate PCCSO's superior performance across different function types, showcasing its ability to escape local optima and solve various function problems with lower memory requirements. Furthermore, the application of PCCSO to DV-Hop node localization is explored, revealing its effectiveness in addressing issues related to asynchronous non-uniform distribution and improving the accuracy of WSN localization. The paper concludes by highlighting the robustness of PCCSO-based DV-Hop localization, emphasizing its ability to handle changes in communication radius and its superiority over traditional CSO and PSO algorithms. Overall, the study presents PCCSO as a promising algorithm for WSN localization, offering improvements in optimization efficiency and memory usage, particularly in DV-Hop localization scenarios.

In their paper [18], Liu et al. introduce an adaptation of the multi-group quasi-affine transformation evolutionary algorithm, aiming to overcome limitations of the original QUATRE algorithm and enhance its efficiency. This novel algorithm, named AMG-QUATRE, involves randomly dividing the population into three groups, each employing a different mutation strategy to balance population diversity and convergence speed. By dynamically adjusting the mutation scale factor during the search process, AMG-QUATRE overcomes manual tuning complexities associated with control parameters, thus improving exploration and development capabilities. The authors apply the AMG-QUATRE algorithm to the node localization problem in wireless sensor networks, specifically enhancing the DV-Hop algorithm. Through experiments using standard benchmark functions, AMG-QUATRE demonstrates superior optimization accuracy compared to QUATRE variants, DE, ODE, CLPSO, and SLPSO algorithms. Additionally, simulations evaluating the proposed DV-Hop algorithm based on AMG-QUATRE show better localization accuracy compared to standard DV-Hop, Hyperbolic-DV-hop, PSO-DV-hop, and DE-DV-hop algorithms across different scenarios. The results highlight AMG-QUATRE's sensitivity to varying anchor node ratios, communication ranges, and node densities, demonstrating its effectiveness in practical localization scenarios. Overall, the study showcases the improved performance of the AMG-QUATRE algorithm in global optimization problems and its effectiveness in enhancing node localization accuracy in WSNs, positioning it as a viable solution for complex optimization tasks.

In their paper [19], Phoemphon et al. introduce a hybrid localization model, termed NS-IPSO, designed to enhance localization accuracy in wireless sensor networks particularly in environments with obstacles. The model combines node segmentation with an improved PSO algorithm to address the challenges posed by obstructions affecting signal transmission in range-based localization techniques. Through node segmentation, sensor nodes are clustered to facilitate more precise distance calculations between anchor nodes and unknown nodes, thereby increasing overall localization accuracy. The NS-IPSO model incorporates significant enhancements to the PSO algorithm, including a tailored fitness function for each anchor node based on the number of hops between anchor nodes and unknown nodes, and mechanisms to mitigate convergence to local optima, ensuring a more robust search for global optimum solutions. Rigorous simulations, considering various node configurations and obstacle-laden environments, demonstrate the superiority of the NS-IPSO model over existing methods such as hybrid discrete PSO (HDPSO), Hybrid PSO, and min-max PSO techniques, particularly in scenarios with obstacles. This research represents a significant advancement in WSN localization, offering a reliable and precise localization mechanism vital for numerous applications in challenging environments where physical obstacles may impede traditional approaches.

The paper [20] by Sabale and Mini delves into the intricacies of localization in wireless sensor networks with a focus on a mobile anchor node path planning mechanism. Addressing the crucial need for accurate sensor node localization in WSNs deployed across various applications, the authors propose a novel Cosine Rule-based Localization (CRL) algorithm, distinct from conventional trilateration methods. The CRL algorithm leverages the cosine rule and Received Signal Strength Indicator (RSSI) to achieve high precision by creating intersecting lines at specific points, enhancing accuracy beyond traditional trilateration techniques. The paper categorizes localization algorithms based on triggering mechanisms, range computation methods, and anchor node availability, while emphasizing the importance of optimizing trajectories for mobile beacon nodes economically. Various trajectory patterns such as Scan, Double Scan, Hilbert, Z-curve, and LMAT are explored, each with its own trade-offs between localization accuracy and energy efficiency. The implementation of the CRL algorithm involves defining communication range parameters, forming curves with beacon positions, and employing the cosine rule for accurate sensor node localization. Simulation results demonstrate the superiority of the CRL algorithm over dominant methods like Accuracy-Priority Trilateration (APT) and Bary–Hilbert localization algorithms in terms of localization precision across different node densities and environmental conditions. The comprehensive study not only provides theoretical foundations and algorithmic descriptions but also validates its efficacy through simulations, highlighting the CRL algorithm's efficiency and feasibility in real-world WSN applications. Overall, the paper contributes valuable insights and solutions to the ongoing challenge of accurate localization in wireless sensor networks.

The paper [21] by Wang et al. introduces the Kernel Extreme Learning Machines based on Hop-count Quantization (KELM-HQ) algorithm to improve node localization accuracy in wireless sensor networks (WSNs). Traditional hop-count-based localization algorithms often use integer hop-count values, leading to suboptimal accuracy in practical scenarios. To address this issue, the KELM-HQ algorithm enhances localization accuracy by transforming integer hop-count values into real-number hop-counts through partitioning a node's one-hop neighbor set into three subsets and calculating intersection region areas to estimate distance. The core innovation of KELM-HQ lies in its utilization of Kernel Extreme Learning Machines, which outperforms existing algorithms such as fast-SVM, GADV-Hop, and DV-Hop ELM in terms of accuracy. The algorithm transforms hop-count values into real numbers to estimate hop boundary distance and transmission range, leading to improved accuracy in unknown node localization. During the training process, the algorithm minimizes both training error and the norm of the output weight using the Extreme Learning Machine (ELM) model with L hidden nodes. The trained KELM model is then used to determine unknown node locations, reducing computation time significantly. Simulation results demonstrate the superior performance of KELM-HQ compared to fast-SVM, GADV-Hop, and DV-Hop-ELM algorithms, with localization error decreasing as the number of anchors increases. KELM-HQ consistently outperforms its counterparts, with significantly smaller localization errors. The algorithm's effectiveness in reducing localization errors while maintaining high accuracy, along with its efficiency in computation time, positions it as a promising solution for node localization in WSNs. Results from the simulation show that the KELM-HQ algorithm improves localization error by 34.6% compared to fast-SVM, 19.2% compared to GADV-Hop, and 11.9% compared to DV-Hop-ELM.

The paper [22] by Rout et al. presents a dynamic genetic algorithm designed to address the challenges of node localization in wireless sensor networks (WSNs), emphasizing accuracy while minimizing energy consumption. By leveraging received RSSI data, the algorithm employs genetic operators including selection, mutation, crossover, and reproduction to maintain a stable population size of candidate solutions. Through multiple RSSI readings and an arithmetic crossover operator, the algorithm aims to minimize the discrepancy between actual and estimated distances, thereby enhancing localization accuracy. The initialization phase involves randomly assigning anchor nodes to the network, followed by estimating distances using signal intensity and computing unknown sensor node coordinates based on an objective function. During the localization process, anchor nodes transmit their coordinate values and power levels to facilitate distance estimation, leading to the computation of the unknown node's position once the distance measurement procedure is complete. Subsequently, the evolutionary algorithm is applied to refine the objective function operation, optimizing the node localization process. The proposed method is evaluated through simulations, which demonstrate its effectiveness in determining wireless device locations with high accuracy. The results indicate that the error decreases as the number of generations increases, with the genetic algorithm finding a satisfactory solution in less than 0.01 seconds around the 33rd generation. The simulation outcomes underscore the efficiency of the proposed approach, with less than one percent average error achieved within a short timeframe. In conclusion, the dynamic genetic algorithm offers a promising solution for node localization in WSNs, mitigating existing challenges and delivering efficient performance in terms of accuracy and energy consumption.

The paper by Zhang et al. [23] introduces a Multi-Strategy Improved Sparrow Search Algorithm (ISSA) aimed at enhancing the performance of the Sparrow Search Algorithm (SSA) and applying it to solve the node localization problem in Heterogeneous Wireless Sensor Networks (HWSNs). While SSA is inspired by the foraging behavior of sparrows and divides the population into producers, scroungers, and investigators, it suffers from slow convergence and low accuracy due to its location updating formulas. To overcome these limitations, ISSA incorporates three key strategies. Firstly, it integrates the golden sine algorithm into the producers' position update to enhance global exploration capability by traversing the search space using the relationship between the unit circle and sine function. Secondly, it incorporates the idea of individual optimality from Particle Swarm Optimization into investigators' updates to improve convergence speed by attracting their positions toward the historical best position. Finally, Gaussian perturbation is applied to globally optimal individuals to prevent them from getting trapped in local optima. Evaluation on 23 benchmark functions demonstrates ISSA's superiority over SSA and other benchmark algorithms in terms of average value, standard deviation, and convergence accuracy. ISSA also exhibits faster convergence speed in most cases. Moreover, when applied to node localization in HWSNs, ISSA outperforms traditional methods like least squares and SSA, yielding higher positioning precision. However, there is room for further improvement, particularly in reducing computation time and exploring distance estimation methods for better results. Overall, ISSA presents a promising solution for real-world engineering optimization problems in HWSNs.

Tan et al. [24] present the Distance Mapping Algorithm (DMA), a novel localization approach tailored for wireless sensor networks, aiming to achieve accurate node positioning while minimizing energy consumption. The algorithm addresses the challenges posed by manual configuration and the high cost of equipping each node with a GPS receiver in large-scale WSN deployments. It begins with an overview of WSNs, emphasizing the significance of localization in meeting accuracy and power consumption requirements, while highlighting the limitations of GPS in such networks. The DMA algorithm comprises three main stages: preprocessing, parameter determination, and non-anchor localization. Preprocessing involves initializing nodes, setting thresholds, and broadcasting packets for anchor and non-anchor nodes. Parameter determination updates link lists and computes matrices for localization variables, enabling anchor nodes to send updated data packages to each other. The localization stage employs trilateration principles optimized with a GA to minimize energy consumption and enhance hardware accuracy. The GA iteratively improves node positions by generating populations of individuals and selecting those with better fitness values through successive generations. Experimental results demonstrate DMA's superior localization accuracy compared to other algorithms like DV-Hop and MDS-map, even with varying numbers of anchor nodes. The study also highlights DMA's energy efficiency, attributed to anchor nodes automatically rejecting nodes outside their range and utilizing neighboring nodes for transmitting local positioning information. In conclusion, DMA presents a promising solution for high-accuracy WSN localization with a focus on energy efficiency, though further enhancements integrating path planning and prediction strategies are suggested for future work.

Annepu and Anbazhagan propose an efficient Extreme Learning Machine (ELM) for node localization in Unmanned Aerial Vehicle (UAV) assisted wireless sensor networks [25]. Their methodology focuses on enhancing localization accuracy using a single UAV and involves UAV-assisted localization and a modified optimization-based least square localization (OLSL) problem. Least square problems are formulated to optimize the UAV's flying height, while Received Signal Strength Indicator (RSSI) multilateration and DEA-assisted OLSL techniques aid in distance measurement between anchor nodes and unknown nodes. The authors categorize localization techniques into fixed terrestrial and mobile aerial anchor-based methods, highlighting the advantages of the latter in providing high accuracy due to the reliability of the air-to-ground channel link. They introduce MLP and ELM models as effective solutions for localization, with ELM showing superiority in computer simulations over multilateration, DEA, and MLP techniques. The paper concludes by underscoring the importance of accurate node localization in various networking protocols and applications and presents UAV-assisted localization as a cost-effective and accurate alternative to classical fixed ground anchor-based methods, with ELM offering significant complexity gains and addressing challenges in WSN node localization effectively. Their findings suggest that the ELM technique outperforms multilateration and DEA methods, providing higher accuracy even for UNs located far from all anchor nodes, thus offering promising avenues for improved node localization in WSNs.

Ingabire et al. [26] introduce a novel approach for outdoor node localization in large-scale urban IoT LoRa networks using Random Neural Networks (RNNs). They highlight the significance of localization in IoT and Wireless Sensor Networks, discussing various range-based localization techniques such as Time-of-Arrival, Received Signal Strength Indicator (RSSI) ranging, and Time-Difference-of-Arrival, commonly employed for determining end device coordinates. The paper focuses on addressing challenges like high-power consumption and hardware costs in dense wireless sensor networks by utilizing RNNs to develop a low-power, large-scale localization system. This system leverages LoRaWAN RSSI values for predicting unknown 2D coordinates on a LoRaWAN dataset for Antwerp, Belgium. The RNN-based localization models outperform other systems in related works, achieving a minimum mean localization error of 0.29 meters. LoRaWAN networks, employing LoRa end devices connected in a star topology, utilize RSSI values for fingerprint localization algorithms. The proposed RNN algorithm predicts the X and Y coordinates of LoRa end devices based on RSSI values received by gateways. Simulation results demonstrate the system's accuracy, with the RNN-based localization approach achieving a minimum mean localization error of 0.39 meters for a small-scale urban area, comparable to traditional approaches like Multilateration and Deep Learning. Notably, it outperforms other RSSI fingerprint LoRaWAN-based localization systems, showcasing its potential effectiveness in large, dense urban environments. The study underscores the promising prospects of the proposed RNN-based approach while suggesting future optimizations and extensions for addressing challenges in more complex environments.

In their study, Kumar et al. [27] propose a range-free 3D node localization technique tailored for anisotropic wireless sensor networks, deploying anchor nodes solely at the top layer and scattering target nodes randomly across bottom layers. Anchor nodes emit beacon signals to assist target nodes in self-localization, with target node coordinates determined using neighboring anchor nodes' known coordinates. The study emphasizes the significance of accounting for device heterogeneity and anisotropic properties in the Range Information Matrix (RIM), introducing fuzzy rule bases, Mamdani fuzzy models, and weight generation through HPSO and BBO to model the relationship between anchor node weight and RSSI for estimating distances. Practical implementation with 80 randomly deployed target nodes in a 10×10 sensor field demonstrates the superiority of the proposed methods over centroid-based approaches and earlier range-free methods, with algorithms optimizing edge weights computing values between 0 and 1 to assess the distance between estimated and actual positions of target nodes. Scalability evaluation reveals performance improvement as the number of anchor and target nodes increases, with anchor node impact diminishing beyond a certain threshold. Comparison results between range-based and range-free methods for 3D node coordinates using HPSO and BBO suggest that while range-based methods outperform range-free methods, the latter are cost-effective and easier to compute.

Notably, the range-free HPSO algorithm offers better accuracy and faster convergence, while the range-free BBO algorithm delivers superior accuracy albeit with slower convergence. In conclusion, the paper introduces two range-free 3D node localization techniques employing HPSO and BBO algorithms for anisotropic WSNs, demonstrating improved performance over centroid-based approaches in terms of error and scalability, while future endeavors could involve refining the algorithms, exploring additional soft computing techniques, and addressing specific challenges in anisotropic WSNs to enhance accuracy and applicability.

Kanoosh et al. [28] tackle the crucial challenge of achieving precise node localization in Wireless Sensor Networks (WSNs), recognizing the impracticality of equipping each sensor node with GPS. Instead, they propose a Salp Swarm Algorithm (SSA) for localization, comparing its performance favorably against other metaheuristic algorithms such as PSO, Butterfly Optimization Algorithm (BOA), Firefly Algorithm, and Grey Wolf Optimizer. The Swarm Intelligence Algorithms section introduces fundamental concepts such as PSO, BOA, Firefly Algorithm, and Grey Wolf Optimizer, elucidating their mechanisms inspired by natural phenomena. SSA, inspired by the collective behavior of salps in deep oceans, updates follower positions using Newton's law of motion. The formulation of the WSN Localization Problem involves a single-hop range-based distribution technique and anchor nodes, with SSA iterating through initialization, anchor communication, and node localization steps. Results demonstrate SSA's superiority in terms of localization accuracy and computing time, with simulations showing increased iterations leading to more localized nodes and reduced localization error, albeit with higher computing time. Moreover, increasing the number of unknown nodes and anchor nodes escalates computing time for all algorithms. The paper concludes by highlighting SSA's outperformance of other algorithms, positioning it as a promising solution for WSN node localization. In summary, the study provides a thorough exploration of node localization algorithms in WSNs, introducing the innovative SSA and showcasing its efficacy through comprehensive experiments, thus contributing significantly to advancing WSN localization techniques.

Latha and Rekha [29] address the vital task of node localization in wireless sensor networks, essential for monitoring environmental changes in remote areas inaccessible to humans, proposing a hybrid metaheuristic approach that combines the BAT algorithm with Simulated Annealing (SA) to improve traditional localization methods like the Distance Vector Hop (DV-Hop) algorithm. In WSNs, nodes are categorized as unknown and Beacon nodes, with Beacon nodes equipped with GPS devices for self-location. While distance-based algorithms rely on parameters like Time of Arrival and Received Signal Strength Indicator, distance-free algorithms such as DV-HOP estimate distance based on the maximum hop count between nodes. The BAT algorithm, inspired by echolocation in bats, explores the solution space using dynamically adjusted frequency pulses, balancing global and local searches. Simulated Annealing enhances the BAT algorithm's efficiency by exploring the solution space and accepting worse solutions with decreasing probability, allowing it to escape local optima. The proposed BAT with SA algorithm optimizes paths and energy consumption across the network, leading to enhanced energy efficiency and an extended network lifetime. Performance evaluation in a 1000m radius network area demonstrates that the proposed algorithm outperforms DV-HOP and the conventional BAT algorithm in accuracy, computing rate, network scalability, and success rate, contributing to a more optimized WSN. Overall, the hybrid BAT with SA algorithm presents a promising solution for WSN localization, consistently demonstrating superior performance compared to traditional methods.

Latha et al. [30] introduce modifications to the APIT algorithm for node localization in wireless sensor networks, partitioning the application area into overlapping and non-overlapping subregions to address challenges like small areas and narrow triangles. The APIT algorithm utilizes beacon signals from anchor nodes to determine the position of target nodes, with each target node comparing the beacon RSSI received from anchor nodes to its neighboring sensor nodes. However, APIT has limitations, including dependency on neighbor nodes and issues with increased network density and specific node distributions. To overcome these challenges, the authors propose enhancing APIT with the Bat algorithm combined with Simulated Annealing (SA). The BAT algorithm utilizes echolocation for distance sensing and dynamically adjusts frequency pulses for effective exploration of the solution space. Simulated Annealing enhances efficiency by accepting worse solutions with decreasing probability, escaping local optima. The proposed algorithm optimizes paths and energy consumption, leading to enhanced energy efficiency and extended network lifetime. Simulations demonstrate that the proposed APIT with Bat-SA algorithm outperforms the conventional APIT, achieving more even node distribution, less node positioning error, and lower latency. Moreover, the algorithm's efficiency improves with increased iterations, and it demonstrates significant reduction in latency compared to the increase in communication radius. Overall, the proposed APIT with Bat-SA algorithm significantly optimizes WSN localization, enhancing system performance and scalability.

Yu et al. [31] present the Quantum Annealing Bat Algorithm (QABA) as an innovative solution for node localization in wireless sensor networks, integrating quantum evolutionary principles and annealing techniques into the bat algorithm framework to enhance both local and global search capabilities. QABA incorporates tournament and natural selection mechanisms to achieve a balance between search exploration and convergence towards optimal values. The paper introduces both 2D (QABA-2D) and 3D (QABA-3D) localization algorithms optimized with QABA, demonstrating significant improvements in convergence speed and solution accuracy compared to other heuristic algorithms. Simulation results highlight QABA's effectiveness across standard test functions, showcasing superior performance in convergent evolutionary strategies. Quantum evolutionary strategies guide variation using optimized values, accelerating convergence and improving accuracy, while simulated annealing enhances solution efficiency by directing populations towards optimal values. QABA's dynamic weighting and improved selection mechanisms ensure efficient convergence while preserving diversity, resulting in superior search performance. Simulations of 2D and 3D spatial localization confirm QABA's higher accuracy and efficiency, with resilience against noise interference and sustained performance across different noise levels, underscoring its robustness and real-world applicability. The paper concludes by highlighting QABA's excellence in node localization optimization for WSNs and proposes future research to extend its application to diverse optimization problems, positioning QABA as a significant advancement in WSN node localization.

Cao et al. [32] introduce a novel node localization method for wireless sensor networks based on the quantum annealing (QA) algorithm, aiming to overcome the limitations of classical approaches like SA and GA. Unlike traditional methods, which struggle with local optima, global optima attainment, and energy consumption issues, QA leverages the quantum tunneling effect to facilitate rapid traversal from local to global optima, simplifying calculations and accelerating computation speed. Simulation results demonstrate the efficacy of QA, showcasing enhanced precision and reduced energy consumption compared to GA and SA, thus offering promising prospects for broader application in WSN node localization. Edge computing, with its distributed nature, offers proximity to data sources, enabling real-time processing and reducing data transmission volume. By exploiting QA's quantum-inspired approach, this method significantly improves localization speed, accuracy, and energy efficiency, outperforming classical algorithms in various metrics. QA's ability to traverse energy barriers efficiently translates into lower energy consumption, prolonging node lifespan and reducing maintenance frequency. In conclusion, the proposed QA algorithm presents a quantum-inspired solution to optimize WSN performance, demonstrating superior precision and energy efficiency, and contributing to the advancement of edge computing applications in WSN node localization.

Rajakumar et al. introduced the Grey Wolf Optimization (GWO) algorithm, drawing inspiration from the social behavior of grey wolves, to tackle the node localization problem in wireless sensor networks [33]. The study implemented the GWO algorithm deploying nodes randomly within the network area and assessing its performance based on metrics such as computation time, percentage of localized nodes, and minimum localization error. The results showcased that GWO exhibited promising performance, demonstrating rapid convergence rates and high success rates compared to other metaheuristic algorithms like PSO and Modified Bat Algorithm. The GWO algorithm mirrors the hierarchical hunting strategy of grey wolves, employing alpha, beta, and delta wolves to guide the exploration and exploitation phases. Alpha wolves represent the best candidate solution, with beta and delta wolves assisting in decision-making and exploration. Strategies such as encircling prey, attacking prey, and searching for prey are employed to explore optimal solutions in both local and global search spaces. In addressing the WSN localization problem, the GWO algorithm initializes unknown and anchor nodes within the communication range, leveraging neighboring anchor nodes to estimate the locations of localized nodes. Environmental factors are accounted for by incorporating Gaussian noise, and optimization techniques are applied to minimize localization errors. Node positions are updated based on the locations of alpha, beta, and delta wolves, steering the algorithm towards optimal solutions while circumventing local optima. The study highlights the efficacy of the GWO algorithm in tackling the localization problem in WSNs, showcasing its superiority over other metaheuristic approaches. Future research avenues include testing the algorithm in dynamic node networks such as Mobile Ad hoc Networks (MANETs) and exploring hybrid algorithms that integrate GWO with other metaheuristic variants to enhance convergence and diversity in identifying unknown node positions.

A. DISCUSSION

The exploration of node localization techniques in wireless sensor networks WSNs is a multifaceted research area that addresses the critical challenge of determining the physical coordinates of sensor nodes within a network. As detailed in Table 2 and the provided summaries, these techniques employ a wide range of strategies, including probabilistic models, SVM, PSO, and various metaheuristic and hybrid algorithms, each with unique benefits and limitations. This analysis will delve into the comparative aspects of these techniques, drawing upon their deployment strategies, computational complexities, and performance metrics such as overhead, accuracy, and scalability.

The range-based methods, including those employing Quantum principles or Grey Wolf Optimization, typically offer more precise localization by using distance measurements between nodes and anchors. In contrast, range-free methods like SSA and BAT-SA rely on proximity and connectivity, providing cost-effective solutions in environments where distance measurement is challenging or impractical.

Hybrid approaches, which combine two or more optimization techniques, are gaining traction for their ability to address the limitations of individual methods. For instance, the Bat Algorithm with Simulated Annealing (BAT-SA) merges the explorative capabilities of the bat algorithm with the fine-tuning proficiency of simulated annealing, leading to improved energy consumption profiles and extended network lifetimes. Similarly, the Quantum Annealing Bat Algorithm (QABA) applies quantum evolutionary strategies to enhance solution quality and convergence speed, representing a significant advancement over traditional heuristic algorithms.

Techniques like Cuckoo Search and Grey Wolf Optimization are relatively newer entrants in the node localization domain, presenting innovative approaches inspired by natural processes. These algorithms show promise in minimizing localization errors and reducing computational requirements, although their performance in real-world scenarios may necessitate further empirical validation.

The deployment of nodes, whether random or uniform, significantly impacts the effectiveness of localization techniques. Random deployment, employed by algorithms like Fuzzy Logic and Neural Network PSO, is typically used in inaccessible terrains, while uniform deployment, seen in methods such as Trilateration PSO, is chosen for structured environments. Regarding mobility, most of the techniques analyzed focus on static nodes, suggesting a primary application in stable environments where nodes do not frequently change their positions.

The computational mode, whether distributed or centralized, affects the network's efficiency and scalability. Distributed techniques, which are prevalent among the reviewed algorithms, allow for parallel computations and thus can potentially scale better to larger networks. However, centralized methods, such as those using cosine rule optimization, can provide more accurate localizations at the cost of higher computational load on central nodes.

Pause time is a critical aspect of certain algorithms, such as the one that integrates Fuzzy Logic with Neural Networks, indicating a requirement for a period of stabilization before localization can proceed. The number of anchors—a term referring to nodes with known locations—varies greatly among the techniques, ranging from zero in SVM to over a hundred in some Fuzzy Logic applications. Anchors are pivotal for range-based localization methods, as they directly influence accuracy.

Accuracy, a paramount performance metric, varies across the techniques. While SVM and PSO exhibit high accuracy, other methods like those based on Voronoi diagrams provide medium accuracy. This metric often correlates with the overhead, the computational and communication resources required for localization. Notably, methods with high overhead, like Likelihood-based localization, can potentially offer better accuracy but may not be suitable for resource-constrained networks.

Scalability is another essential factor, especially for WSNs that may need to operate over large areas or integrate additional nodes over time. Techniques such as probabilistic models and SVM exhibit high scalability, making them well-suited for expanding networks. Meanwhile, some PSO-based methods show medium scalability, highlighting the need for a balance between localization performance and network growth.

Upon comparative evaluation, it is apparent that there is no one-fits-all solution for node localization in WSNs. Each algorithm presents a trade-off between various performance metrics. For example, probabilistic methods are scalable and have moderate accuracy but may incur long beacon utilization times, making them less desirable in time-sensitive applications. SVM stands out for its high accuracy and scalability with zero anchor requirement, suggesting its utility in densely deployed networks. PSO variants are versatile, being applicable in both centralized and distributed modes, but their performance is often contingent upon the specific variant and parameter tuning.

B. SUMMARY OF OUR LITERATURE REVIEW CONTRIBUTIONS

This review offers several distinct contributions to the existing body of work on node localization accuracy in wireless sensor networks (WSNs). First, it provides a comprehensive comparison of a wide range of localization techniques, as detailed in Table 2, highlighting key aspects such as computational mode, node deployment, mobility, the number of anchors, and accuracy. Unlike many existing reviews that focus on a limited set of techniques, this review encompasses both traditional methods like trilateration and advanced optimization-based approaches such as PSO, GA, and Harmony Search. Secondly, it systematically evaluates the scalability and overhead of each method, offering a nuanced understanding of their practical applicability in diverse WSN scenarios. Furthermore, the inclusion of both static and mobile node scenarios broadens the scope of the review, making it relevant to a wider range of applications. By integrating these various dimensions, this review not only identifies the strengths and limitations of individual methods but also provides a holistic view of the current state of node localization technologies, paving the way for future research and development in this critical area of WSNs.

C. COMPARISON WITH OTHER LITERATURE REVIEWS

Ahmad et al. [34] offer a thorough review of various optimization algorithms applied to node localization in WSNs. It emphasizes the significance of accurate node localization and examines a wide array of optimization techniques, including evolutionary algorithms, swarm intelligence, and metaheuristic approaches. The review evaluates these techniques based on key factors such as localization accuracy, scalability, computational complexity, and robustness. By identifying the strengths and limitations of each optimization approach, the review provides valuable insights into their applicability across different WSN deployment scenarios. Compared to the current review, Ahmad et al. provided an extensive analysis focused primarily on optimization algorithms, highlighting their effectiveness in improving localization accuracy and other performance metrics. While both reviews aim to enhance node localization in WSNs, the current review expands on this by integrating traditional methods like trilateration with advanced optimization techniques. It also explores the practical aspects of implementing these methods within a WSN localization simulator, offering

TABLE 2. Reviewed techniques comparison.

Technique	Used Methods	Nodes Deployment	Mobility	Computational Mode	Pause Time	Number of Anchors	Node Coordinates	Beacon Utilization	Overhead	Accuracy	Scalability
[4]	Probabilistic	Random	Static	Distributed	No	2-5	3D	Long	Medium	Medium	High
[5]	SVM	Uniform	Static	Distributed	No	0	2D	Medium	Low	High	High
[6]	PSO	Random	Static	Centralized	Yes	2	3D	Medium	Medium	Medium	Low
[7]	ML	Random	Static	Distributed	No	0-4	2D	High	Medium	Medium	Medium
[8]	FL & ANN	Random	Static	Distributed	Yes	121	2D	Medium	Medium	Medium	Medium
[9]	PSO	Random	Static	Distributed	No	4	2D	Long	Medium	High	Medium
[10]	BO	Random	Static	Distributed	No	0	2D	No	Medium	Medium	Medium
[11]	Fuzzy Logic	Random	Static	Distributed	No	110	2D	High	Low	High	Medium
[12]	Cuckoo Search	Random	Static	Distributed	No	3	2D	Medium	Medium	Medium	Medium
[13]	PSO BBO	Random	Static	Distributed	No	3+	2D	No	Low	High	High
[14]	Trilateration	Uniform	Static	Distributed	No	25	2D	Long	High	Medium	High
[15]	PSO	Random	Static	Distributed	No	100-250	3D	Medium	Low	High	High
[16]	SVM	Random	Static	Distributed	No	100	3D	Medium	Medium	High	Medium
[17]	CAT & PSO	Random	Static	Distributed	No	15-40	2D	No	Low	Medium	High
[18]	PSO	Random	Static	Distributed	No	5-40	2D	No	Medium	High	High
[19]	PSO	Uniform	Static	Centralized	No	10-25	2D	Long	Low	High	High
[20]	Cosine Rule	Uniform	Mobile	Distributed	Yes	1	2D	Medium	Medium	High	Medium
[21]	ML	Random	Static	Distributed	No	5-35	2D	No	Medium	High	High
[22]	GA	Random	Static	Distributed	No	3+	2D	No	Medium	High	Medium
[23]	Sparrow Search	Random	Static	Distributed	No	20-25+	2D	No	Medium	High	Medium
[24]	DMA	Random	Mobile	Distributed	No	4-20	2D	No	Medium	High	High
[25]	UAV	Random	Mobile	Distributed	No	4-36	2D	No	Medium	High	Medium
[26]	RNN	Random	Static	Distributed	No	72	2D	No	Medium	High	High
[27]	PSO	Random	Static	Distributed	No	20-400	3D	Medium	Medium	High	High
[28]	SSA	Random	Static	Distributed	No	10-35	2D	No	Medium	High	High
[29]	SA & BAT	Uniform	Static	Distributed	No	0	2D	No	Medium	High	High
[30]	SA & BAT	Uniform	Static	Distributed	No	25	2D	No	Medium	High	High
[31]	QA & BAT	Random	Static	Distributed	No	5-30	2D-3D	No	Medium	High	High
[32]	QA & EC	Random	Static	Distributed	No	30	2D	No	Medium	High	High
[33]	GWO	Random	Static	Distributed	No	10-100	2D	No	Medium	High	Medium

a hands-on evaluation of their performance under varied network conditions.

Osamy et al. [35] focus on the integration of Artificial Intelligence (AI) methods to address challenges related to Coverage, Deployment, and Localization in WSNs. Spanning research from 2010 to 2021, it provides a detailed analysis of AI techniques such as swarm intelligence, nature-inspired algorithms, and evolutionary computation, evaluating their effectiveness in enhancing various WSN functions. The review systematically discusses the performance parameters, objectives, and deployment scenarios of different AI methods, guiding researchers toward understanding the latest applications and identifying suitable AI approaches for specific WSN challenges. Additionally, it outlines open research issues and suggests future research directions, thereby offering a roadmap for advancing WSN technologies using AI. Compared to the current review, the article has a broader focus on AI methods applied to multiple aspects of WSNs, including coverage and deployment, in addition to localization. While the current review is specifically concentrated on node localization accuracy, it stands out by integrating traditional geometric methods with optimization algorithms. Furthermore, the current review's in-depth analysis of localization errors and the impact of network parameters provides a targeted enhancement over the more general AI-focused analysis.

Annepu et al. [36] focus on node localization using unmanned aerial vehicles (UAVs) and compares various soft computing-based localization techniques, including traditional multilateration and advanced neural network architectures. It highlights the limitations of the conventional RSS-multilateration technique due to distortions in the propagation medium and underscores the advantages of optimization-based schemes and neural networks. Among the neural networks, the Extreme Learning Machine (ELM) is emphasized for its superior performance over other NN classifiers like Multilayer Perceptron (MLP) and Radial Basis Function (RBF) due to its fast and strong learning capabilities. The article provides a comprehensive review of localization techniques using both fixed terrestrial anchor nodes and aerial anchor nodes, presenting a detailed comparison of their performance, complexity, and suitability for different WSN scenarios. Compared to the current review, the article narrows its focus on the use of UAVs and soft computing techniques for node localization. While it provides a valuable comparison of traditional and advanced neural network approaches, its primary emphasis is on overcoming the limitations of RSS-based multilateration through the use of sophisticated learning algorithms. The current review, on the other hand, offers a broader evaluation perspective using a WSN localization simulator, which rigorously tests the hybrid method under varied network conditions, providing actionable insights into its real-world applicability. This hands-on approach, combined with the detailed analysis of localization errors and the impact of network parameters, sets the current review apart by offering a more comprehensive and practical examination of localization techniques across diverse WSN scenarios.

III. IMPLEMENTED WSN LOCALIZATION METHODS

A. TRILATERATION TECHNIQUE

Trilateration is a mathematical technique used in various fields such as geography, navigation, and mobile phone tracking, to determine the precise location of a point by measuring its distance from three distinct points [37]. This method is pivotal in localization technologies, especially in GPS, where it helps to pinpoint the exact position of a GPS receiver on Earth's surface. Understanding trilateration requires an exploration of its principles, applications, and how it distinguishes itself from similar methods like triangulation.

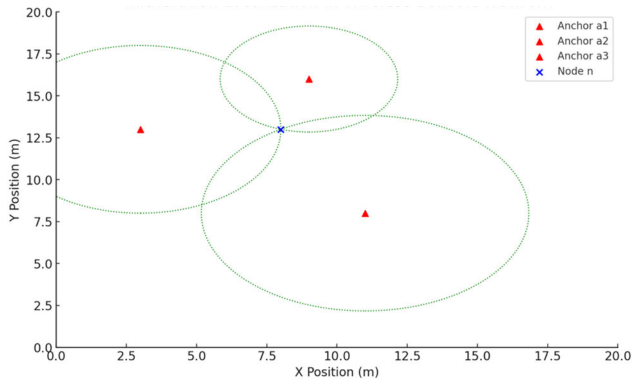


FIGURE 2. Trilateration using three anchor nodes.

At its core, trilateration is based on the concept of circles (Fig. 2), spheres, or hyperbolas, depending on whether the context is two-dimensional or three-dimensional. In a two-dimensional space, for instance, trilateration involves drawing circles around three known points. Each circle's radius corresponds to the distance from that point to the unknown location. The point where all three circles intersect is the location being determined. In three-dimensional space, such as with GPS, spheres are used instead of circles.

Given three known points $A(x_1, y_1)$, $B(x_2, y_2)$, and $C(x_3, y_3)$ and their distances to an unknown point (x, y) as d_1, d_2 and d_3, respectively, the distance formulas are:

$$(x - x_1)^2 + (y - y_1)^2 = d_1^2$$
$$(x - x_2)^2 + (y - y_2)^2 = d_2^2$$
$$(x - x_3)^2 + (y - y_3)^2 = d_3^2$$

These equations represent circles around each known point with radii equal to the distances d_1, d_2 and d_3. The solution to this system of equations is the coordinates of the unknown point where these circles intersect (Fig. 3).

When the receiver calculates its distance from at least three satellites, it effectively determines three spheres in space. The point where these spheres intersect is the receiver's location. Because of minor errors in time measurement, a fourth satellite is often used to correct any discrepancies, enhancing the accuracy of the location data.

Trilateration has a wide array of applications beyond GPS. It is used in cellular phone tracking, where the location of a phone is determined by its distance from cell towers. In indoor positioning systems, it helps navigate complex indoor spaces where GPS signals are unavailable. Even in robotics and drone navigation, trilateration plays a crucial role in enabling autonomous movement by helping these machines under-stand their positions relative to known points in their environment.

B. BOUNDING BOX

The bounding box technique for node localization in wireless sensor networks is a geometric method used to estimate the

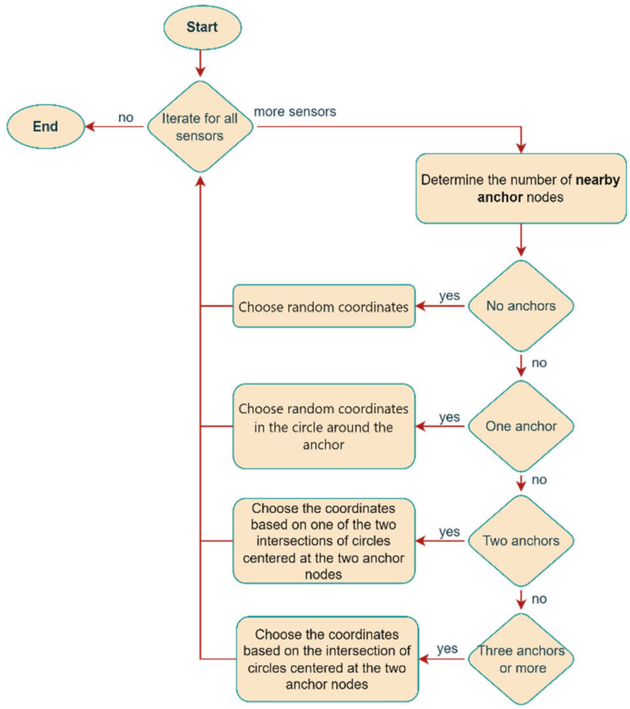


FIGURE 3. Flowchart of the Trilateration technique.

location of an unknown node based on the known locations of several anchor nodes (also known as beacon nodes) and the measured distances from these anchors to the unknown node [38]. This method is particularly useful when you need a simple and computationally inexpensive approach to localize nodes in a network (Fig. 4).

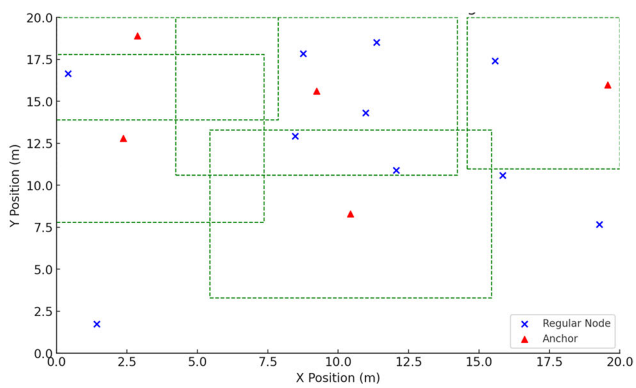


FIGURE 4. Bounding Box localization in WSN.

Assuming the distance d to the unknown node from at least three non-collinear anchor nodes is known, a bounding box can be formed around the unknown node's possible location. This box is defined by the maximum and minimum x and y coordinates that the unknown node can occupy based on the distances to the anchor nodes (Fig. 5).

For each anchor node A_i with coordinates (x_i, y_i) and distance d_i to the unknown node, the bounding box coordinates can be defined as:

$$x_{min} = \max(x_{min}, x_i - d_i)$$
$$y_{min} = \max(y_{min}, y_i - d_i)$$
$$x_{max} = \min(x_{max}, x_i + d_i)$$
$$y_{max} = \min(y_{max}, x_i + d_i)$$

where $x_{min}, x_{max}, y_{min}, y_{max}$ are the initial bounding box coordinates which are updated as each anchor node is processed.

The position of the unknown node can then be estimated as a random point inside the resulting bounding box:

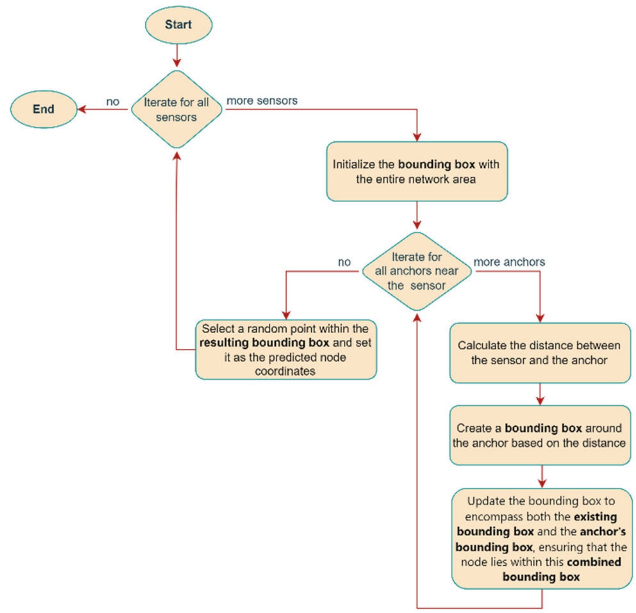


FIGURE 5. Flowchart of Bounding Box used for node localization.

In practice, further refinement steps may be necessary to improve the accuracy of the location estimate. These may include using more sophisticated models to estimate distance based on signal strength, accounting for obstacles, or using optimization techniques to minimize the error between the measured distances and the distances implied by the estimated position.

C. HARMONY SEARCH

Inspired by the musical improvisation process, the HS algorithm mimics the improvisation of musicians in searching for better harmony. Originally proposed by Geem et al. in [39], the HS algorithm has been successfully applied to various optimization problems, including node localization in WSNs. At its core, the HS algorithm maintains a population of candidate solutions called "harmonies." Each harmony represents a possible solution to the optimization problem, in this case, the coordinates of sensor nodes in the network. The algorithm iteratively improves these solutions by iteratively refining them based on the concept of harmony memory, pitch adjustment, and harmony updating.

In the context of node localization (Fig. 6), the HS algorithm aims to find the optimal spatial coordinates for sensor nodes that minimize the localization error [40]. The algorithm starts with an initial population of random solutions representing potential node locations. During each iteration, the algorithm evaluates the fitness of each har-

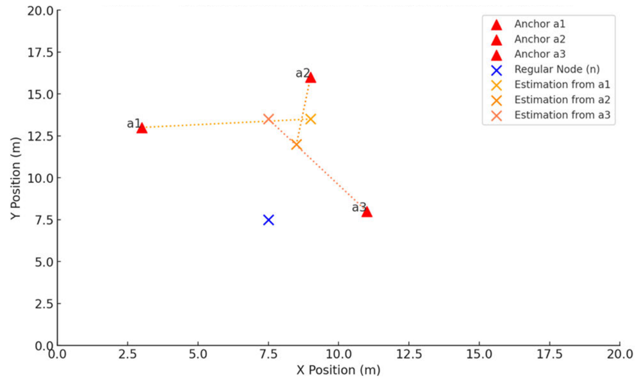


FIGURE 6. **Harmony Search localization in WSN.**

mony solution based on a localization error metric, such as Euclidean distance or received signal strength indicator (RSSI) measurements.

As shown in Fig. 7, the HS algorithm then iteratively refines these solutions by exploring the search space and adjusting the pitch of each harmony. This exploration-exploitation tradeoff is crucial for balancing between local and global search capabilities, ensuring that the algorithm converges to near-optimal solutions without getting trapped in local optima.

One of the key advantages of the HS algorithm is its ability to adaptively adjust the search process based on the harmony memory and pitch adjustment rate. By dynamically updating these parameters, the algorithm can effectively explore the search space and converge to high-quality solutions efficiently. Moreover, the HS algorithm is well-suited for node localization in WSNs due to its simplicity, robustness, and flexibility. Unlike traditional optimization techniques, such as gradient descent or simulated annealing, the HS algorithm does not require derivative information or complex parameter tuning. This makes it particularly attractive for resource-constrained sensor nodes with limited computational capabilities and energy resources. Furthermore, the HS algorithm can be easily parallelized and distributed, allowing for scalable implementation in large-scale WSNs. By leveraging parallel processing capabilities, the algorithm can expedite the localization process and accommodate dynamic changes in network topology or environmental conditions.

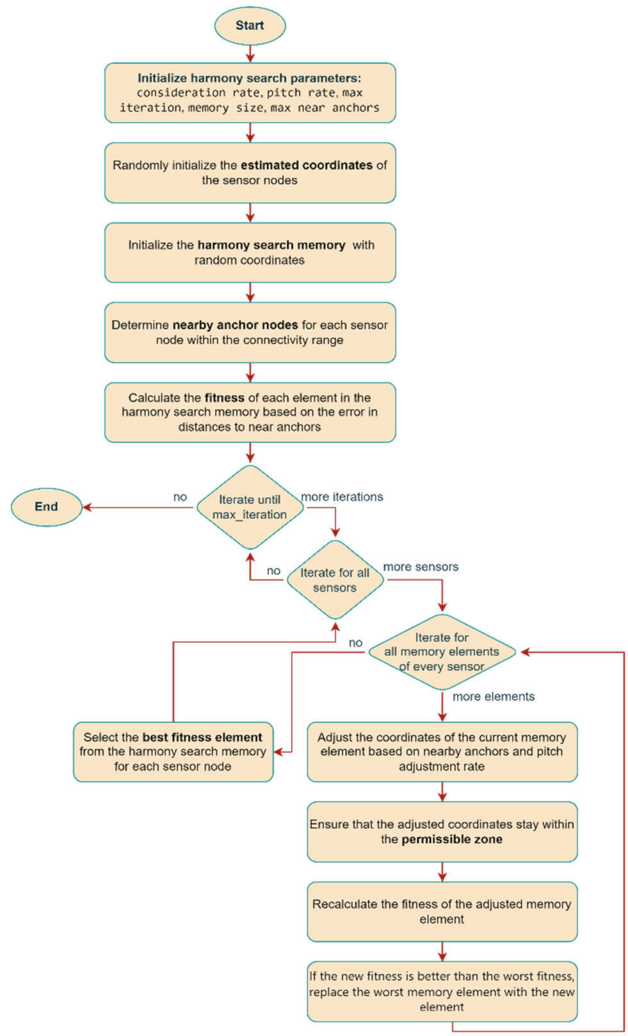


FIGURE 7. **Flowchart of Harmony Search used for node localization.**

D. PROPOSED METHODS (BBHS AND HSBB): HYBRIDIZATION OF BOUNDING BOX AND HARMONY SEARCH

The bounding box technique is a straightforward yet effective method for node localization. By encapsulating a node's possible positions within a bounding box based on nearby anchor nodes, it provides a simple yet robust solution. However, its deterministic nature may lead to suboptimal solutions, especially in complex environments with irregular shapes or obstacles. On the other hand, the harmony search technique presents a metaheuristic optimization approach inspired by musical improvisation. It iteratively refines candidate solutions within a search space, aiming to converge towards optimal solutions. Harmony search exhibits strong exploration and exploitation capabilities, enabling it to navigate complex solution spaces effectively. However, its computational complexity and reliance on parameter tuning may hinder its applicability in resource-constrained WSNs.

To address the limitations of both bounding box and harmony search techniques while leveraging their respective strengths, a hybridization approach is proposed. The hybrid technique (Fig. 8) combines the simplicity of bounding box localization with the search capabilities of harmony search to tailor the bounding box and refine the predicted node position within it.

The BBHS hybrid technique begins by applying the bounding box technique to determine the smallest bounding box enclosing a node's possible positions based on nearby anchor nodes. This initial bounding box serves as the search space for harmony search optimization. By constraining the search space to a smaller region of interest defined by the bounding box, the hybrid approach aims to improve the efficiency of

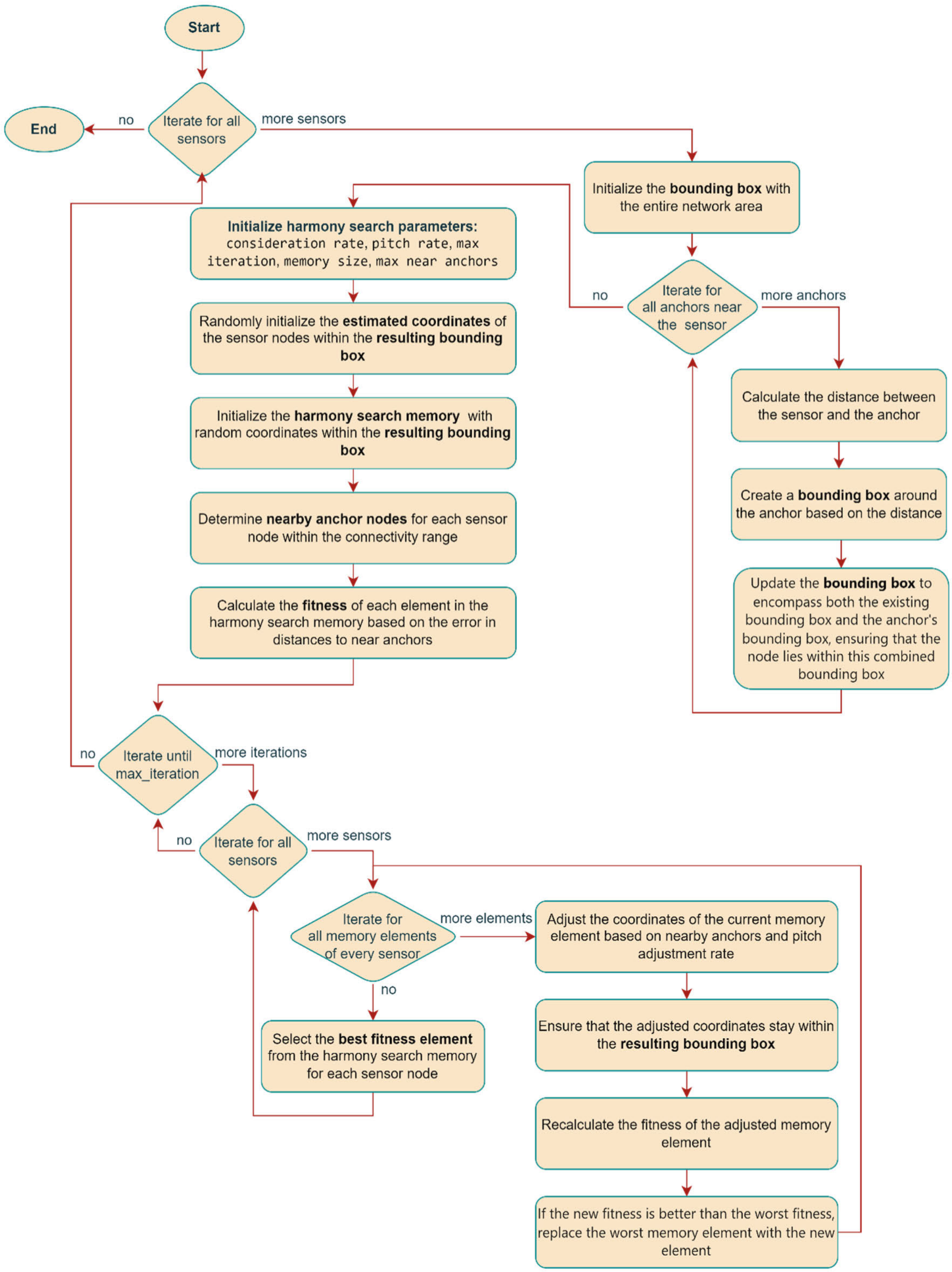


FIGURE 8. **Flowchart of BBHS (Bounding Box followed by Harmony Search).**

harmony search while preserving the robustness of bounding box localization.

Once the bounding box is determined, the harmony search technique is employed to search for the predicted node position within the bounding box (Fig. 9). Harmony search iteratively explores and refines candidate solutions within the constrained search space, aiming to minimize localization error and improve accuracy. The harmony search algorithm adjusts candidate solutions based on harmony memory, pitch adjustment, and

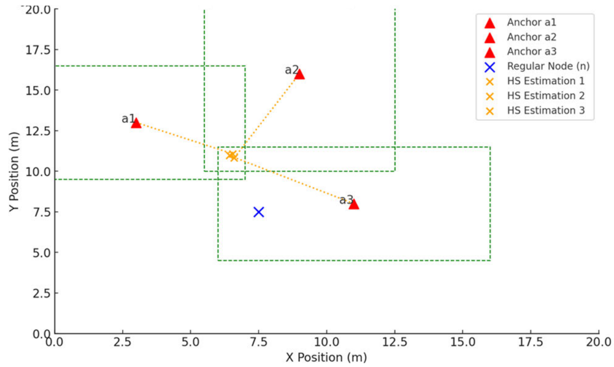


FIGURE 9. **BBHS localization in WSN.**

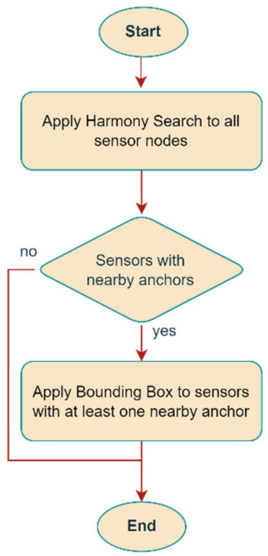


FIGURE 10. **Flowchart of HSBB (Harmony Search followed by Bounding Box).**

fitness evaluation, iteratively improving the predicted node position.

The second hybridization version is called HSBB. The hybrid protocol depicted in Fig. 10 employs a combination of Harmony Search and Bounding Box techniques to localize sensor nodes, optimizing their positions within a given space. The Harmony Search algorithm is applied to all sensor nodes in the network. Then, the protocol evaluates whether the sensors have nearby anchor nodes. If so, the Bounding Box method is applied. This hybrid approach harnesses the global optimization capabilities of Harmony Search with the precision-enhancing Bounding Box method to efficiently and accurately localize sensor nodes in a network.

IV. DEVELOPING A NETWORK SIMULATOR FOR NODE LOCALIZATION

In order to evaluate and optimize localization algorithms, researchers often rely on simulations to emulate real-world network scenarios in a controlled environment. Building a wireless network simulator tailored for node localization offers a powerful tool for testing algorithms, analyzing performance, and gaining insights into network behavior. We explore in this section the design and implementation of such a simulator, focusing on key parameters and considerations.

A. DESIGNING THE SIMULATOR

The development of a wireless network simulator for node localization involves several key components and considerations:

1) PARAMETERS DEFINITION

The simulator should allow users to specify essential parameters that characterize the simulated network environment. Four critical parameters include:

- *Number of Nodes:* Represents the total number of sensor nodes deployed in the network.
- *Number of Anchors:* Denotes the number of anchor nodes with known positions used for localization reference.
- *Connectivity Range:* Defines the maximum distance within which sensor nodes can communicate directly with each other or anchors.
- *Zone Width:* Specifies the width of the simulation area or zone where nodes are deployed.

The simulator generates the network topology based on the specified parameters. It distributes sensor nodes and anchor nodes randomly or following a predefined pattern within the simulation area. Additionally, it establishes communication links between nodes based on the connectivity range.

2) LOCALIZATION ALGORITHM INTEGRATION

The simulator incorporates various node localization algorithms for evaluation. It enables users to select and compare different algorithms, such as bounding box, trilateration, or hybrid approaches. The simulator applies the chosen algorithm to estimate the positions of sensor nodes based on received signal strength, time of arrival, or other localization metrics.

3) PERFORMANCE METRICS

To assess the effectiveness of localization algorithms, the simulator calculates and reports relevant performance metrics. These metrics may include localization accuracy, error distribution, convergence time, energy consumption, and scalability. By analyzing these metrics, researchers can evaluate the strengths and limitations of different algorithms under diverse network conditions.

4) VISUALIZATION AND ANALYSIS TOOLS

The simulator provides visualization tools to depict the simulated network topology, node positions, communication links, and localization results. It enables users to visualize algorithm behavior, identify localization errors, and gain insights into network dynamics. Additionally, the simulator may offer data analysis capabilities to generate plots, histograms, or statistical summaries of simulation results.

B. CONCEPTUAL DESIGN OF A WSN LOCALIZATION SIMULATOR

The development of a wireless network simulator for node localization involves several key components and considerations. We will use UML diagrams to describe the conceptual development of the simulator.

1) USE CASE DIAGRAM

The case diagram (Fig. 11) outlines the interactions a user has with the WSN Localization Simulator. The user is the primary actor who can perform six main use cases within the system. These include ''Configure Simulation Parameters'' to set up initial conditions; ''Select Localization Method'' to determine how nodes are localized; ''Set Simulation Confidence'' to define the robustness of the simulation; ''Run Simulation'' to execute the simulation with the given parameters; ''View Results'' to inspect the outcomes, and ''Visualize Network'' to see a graphical representation of the network based on the simulation data. The use cases are depicted as processes that the user can initiate or control in the simulator environment.

2) ACTIVITY DIAGRAM

The activity diagram (Fig. 12) depicts the workflow of a WSN localization simulator from start to finish. The process begins at the ''Start'' node and proceeds linearly. First, the user ''Set Simulation Parameters'' for the WSN environment. Next, they ''Select Localization Method'' to be used for the nodes. The user then ''Set Simulation Confidence,'' which likely determines the number of iterations or the accuracy of the simulation. The simulation is then ''Run,'' followed by the system ''Calculate Localization Error'' to evaluate the performance. Subsequently, the ''Generate Results'' activity creates output data or visualizations, which leads to ''Visualize Network,'' where the network's graphical representation is displayed.

3) DEPLOYMENT DIAGRAM

The deployment diagram (Fig. 13) depicts the system architecture for a simulation application, divided across three main nodes: User Workstation, Simulation Server, and Database Server. The User Workstation hosts the UI Component, through which users interact with the system. The Simulation Server is the computational heart, housing the Simulation Engine and the Result Processor, which performs simulations and processes the outcomes, respectively. The Database Server contains the Database, storing simulation data and results. Communication flows are indicated, such as ''User to UI'' for interactions, ''UI to Simulation Engine'' for simulation commands, and ''Simulation Engine to Database'' for data persistence. The Result Processor also communicates with the UI Component, feeding results back for display.

4) CLASS DIAGRAM

The class diagram (Fig. 14) represents the structure of the WSN Localization Simulator. The main class, Simulator, has associations with several other classes: SimulationZone, SimulationParameters, LocalizationMethod, SimulationResult, Node, and Graph. SimulationZone holds attributes for the dimensions of the simulation area (width, height). SimulationParameters contains various settings like nodeRange, anchorRange, connectivityRange, localizationMethod, and numberOfRuns, and offers methods to validate and summarize these parameters. The LocalizationMethod class has a method localizeNodes() to determine the positions of the nodes based on the anchor nodes. SimulationResult class manages the results of the simulation, including various error metrics and methods for graph generation, saving, and summarizing results. Node class represents the sensors in the network, with attributes like coordinates, nearbyAnchors, and methods for distance calculation, position prediction, and anchor identification. Anchor class is a subclass of Node, adding anchorID. The class Graph encapsulates the data and methods needed to create and manipulate graphical representations (plot(), update(), export()), with attributes to describe its structure.

5) SEQUENCE DIAGRAM

The sequence diagram (Fig. 15) illustrates the interactions between the user, various system components, and objects within a WSN localization simulator. The diagram is a visual representation of how different parts of the system communicate in order to complete a simulation run and display the results to the user.

At the beginning, the user interacts with the ''Simulation Parameters Pane,'' where they set the parameters for the simulation. This action is denoted by the ''Set parameters'' message. Once the parameters are set, the user sends a ''Start Simulation'' message to the ''Simulator'' object, initiating the simulation process. The ''Simulator'' object then sends ''Initialize Simulation'' messages to itself, preparing the simulation environment. This involves two key actions: ''Create Nodes'' and ''Create Anchors,'' which are sent as messages from the ''Simulator'' to the ''Node'' and ''Anchor'' objects, respectively. This depicts the creation of nodes and anchors within the simulation space according to the parameters defined by the user.

Following the initialization, each ''Node'' object requests distance information from each ''Anchor'' object by sending a ''Request Distance'' message. The ''Anchor'' objects respond with ''Respond with Distance'' messages, providing the necessary distance data back to the ''Nodes.'' Once all distance information has been gathered, the ''Simulator'' moves forward with the localization process. A ''Predict Positions'' message is sent to the ''LocalizationMethod'' object, which computes the predicted positions of the nodes. The results are then communicated back to the ''Simulator'' with a ''Set Predicted Position'' message for each node.

The ''Simulator'' continues to compile the simulation results and sends a ''Compile Results'' message to the ''SimulationResult'' object. The process of result compilation is likely to involve computing localization errors and

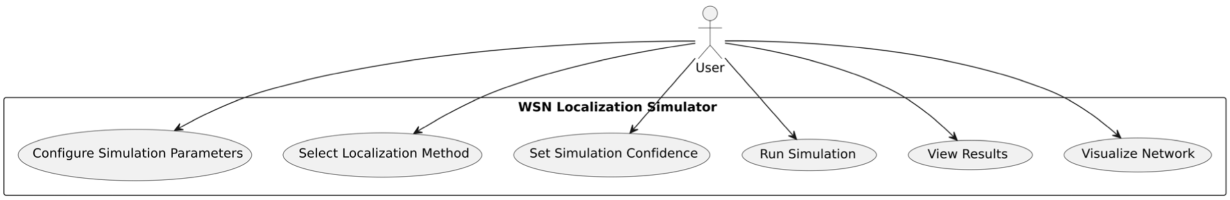


FIGURE 11. Use case diagram for WSN localization simulator.

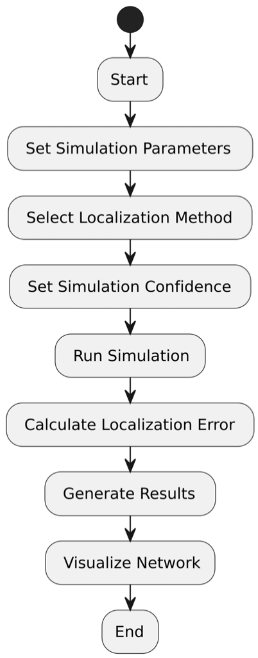


FIGURE 12. Activity diagram for WSN localization simulator.

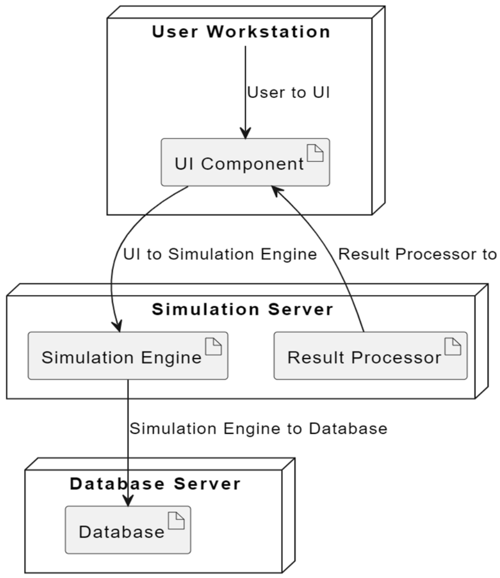


FIGURE 13. Deployment diagram for WSN localization simulator.

preparing the data for visualization. After the results are compiled, the "Simulator" sends a "Generate Graphs" message to the "Simulation Results Pane." This action generates the necessary graphs that depict the localization errors in various forms, as specified by the user's parameters.

Simultaneously, an "Update Visualization" message is sent to the "Network Visualization Pane." This pane updates the visual representation of the network, showcasing the nodes, anchors, and possibly their connectivity and predicted positions based on the simulation data. The last two interactions in the diagram represent the user's experience of viewing the outcomes of the simulation. The "Display Graphs" and "Display Network" messages likely correspond to the system presenting the visual data on the screen, allowing the user to analyze the results and the visualized network after the simulation run is complete.

6) STATE DIAGRAM

In Fig. 16, the state diagram represents the various states of a simulation process and the transitions between them. The process starts in the "Idle" state, transitions to "Configuring_Parameters" when the user is inputting settings, and moves to "Ready" once all parameters are set. The "Running_Simulation" state follows the start of the simulation, which then progresses to "Generating_Graphs" upon completion. After graphs are generated, the system enters "Displaying_Results" and finally reaches "Simulation_Complete." There's an "Error_State" that can be reached from any state if a failure occurs, with transitions back to either "Idle" or "Running_Simulation" after the error is fixed, allowing for error handling and recovery.

7) COMPONENT DIAGRAM

The component diagram (Fig. 17) illustrates the architecture of a WSN Localization Simulator. It shows a Simulation Engine at the core, interacting with a User Interface that includes a Simulation Control Panel. The Parameter Store holds simulation parameters which are updated by the engine and sent from the control panel. Data Management is a central component responsible for orchestrating the flow of data between the engine, Visualization for graphical representation, Reporting for output generation, Persistence for data storage, and a Results Store. The Results Display Panel queries the Results Store to present outcomes to the user, completing the feedback loop of the simulation cycle.

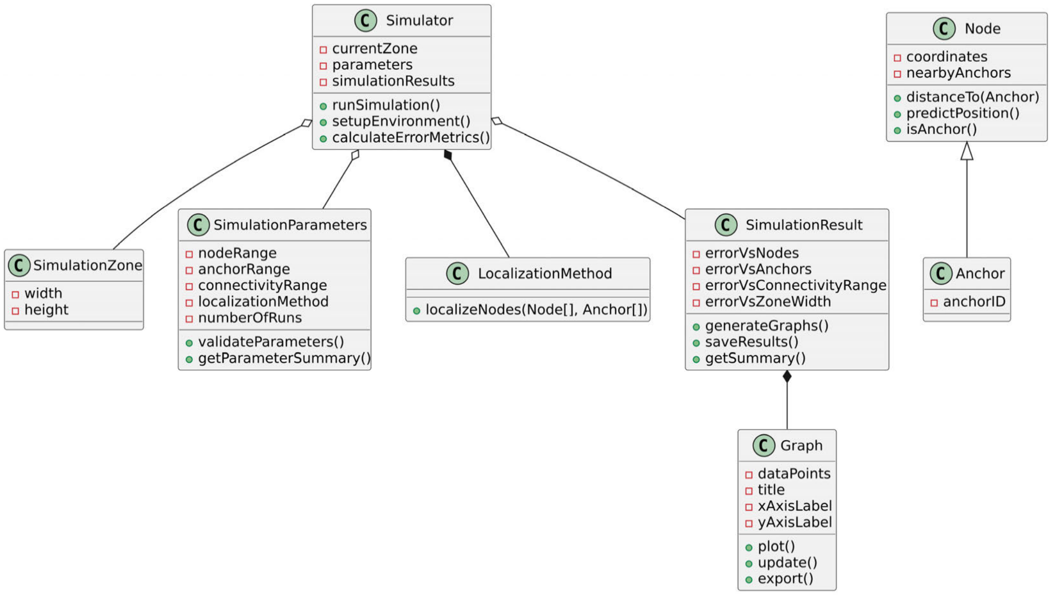


FIGURE 14. Class diagram for WSN localization simulator.

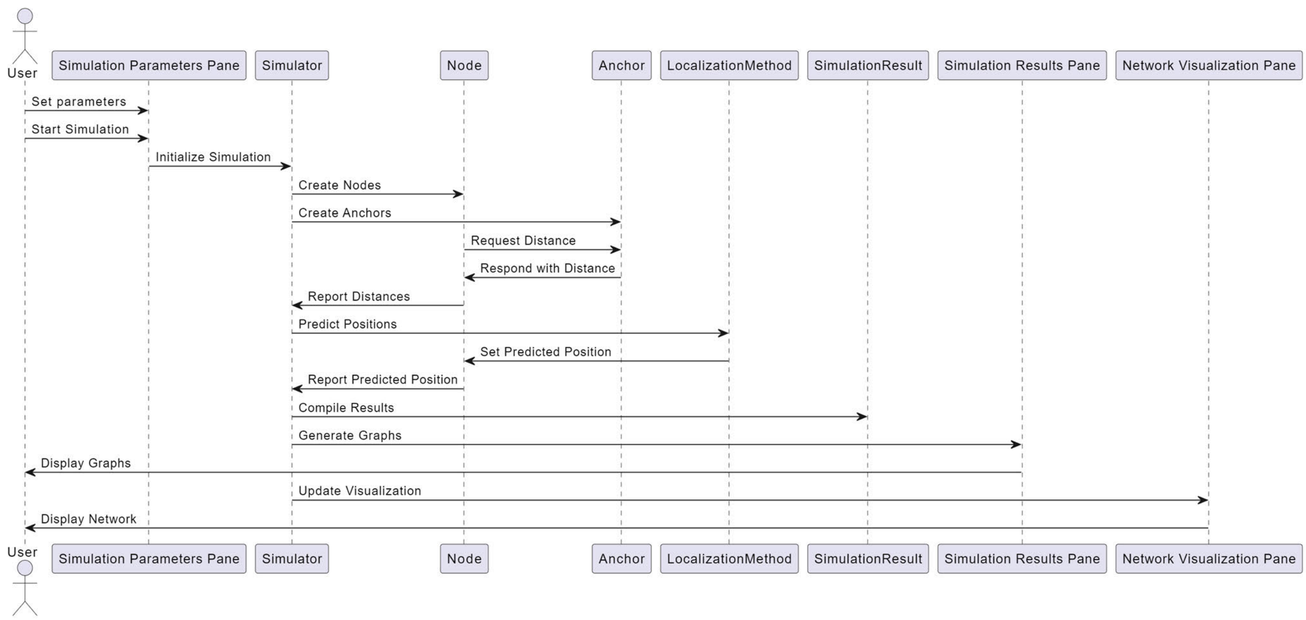


FIGURE 15. Sequence diagram for WSN localization simulator.

C. SIMULATOR INTERFACE

The used WSN localization simulator, a specialized tool designed to model and analyze the localization of nodes within a wireless sensor network. This simulator is developed by Mohammed Omari at the American University of Ras Al Khaimah (AURAK) and funded by Mohammed Bin Rashed Smart Learning Program (MBRSLP) in the year 2023. The interface (Fig. 18) is user-friendly and divided into several distinct sections, each dedicated to a specific aspect of the simulation process.

1) PARAMETERS PANE

- *Simulation Zone:* At the top-left corner, users can set the dimensions of the simulation area, with options to adjust the width and height. There are also checkboxes to toggle the visibility of connectivity and prediction lines within the simulated network.
- *Node Configuration:* The "Nodes" section allows users to define the range of nodes to be included in the simulation by setting a minimum and maximum value. Additionally, users can define a step size for

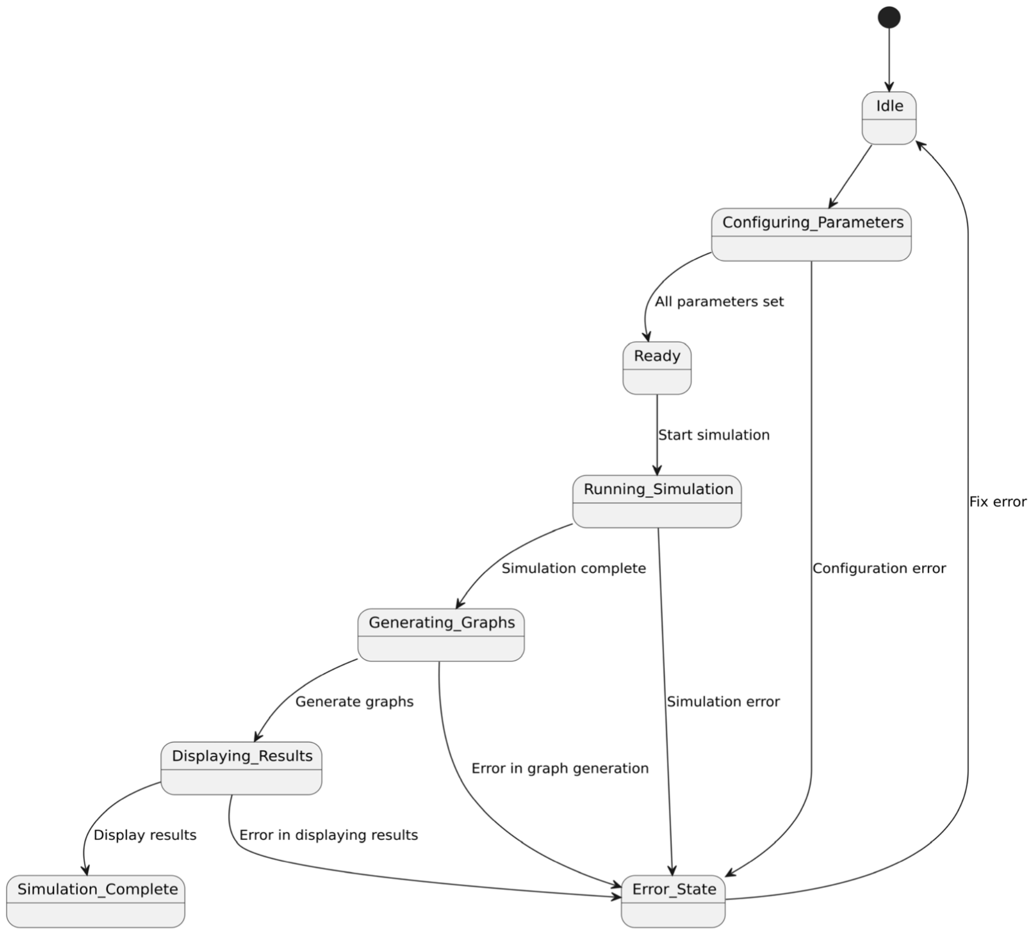


FIGURE 16. State diagram for WSN localization simulator.

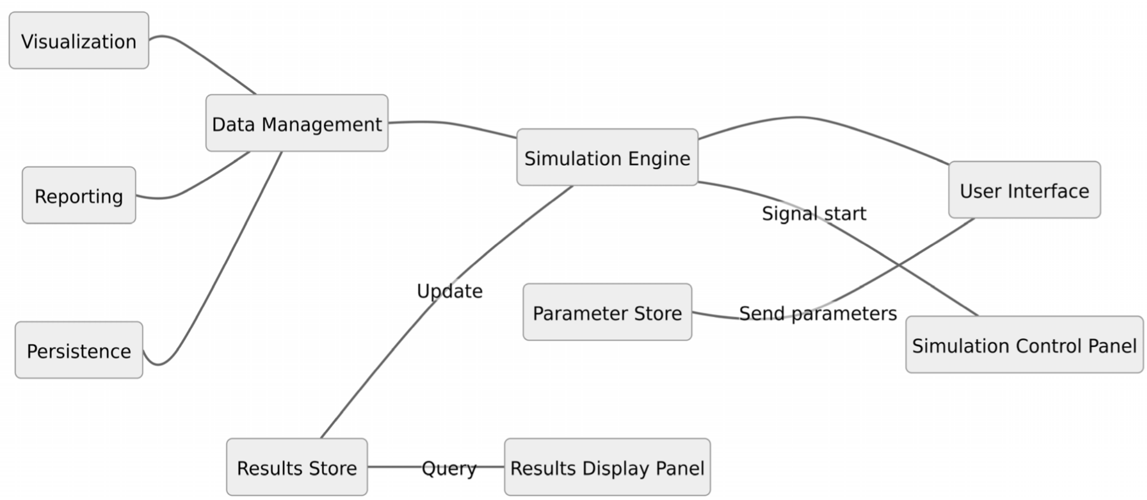


FIGURE 17. Component diagram for WSN localization simulator.

the range and have the option to fix the number of nodes.

- *Anchor Configuration:* Similarly, the "Anchors" section enables the configuration of anchors within the network. Users can set the range of anchors, providing a minimum and maximum count, a step size, and an option to use a fixed value only.
- *Connectivity Range:* This parameter is crucial as it defines the range within which nodes can communicate. Users can specify a minimum and maximum

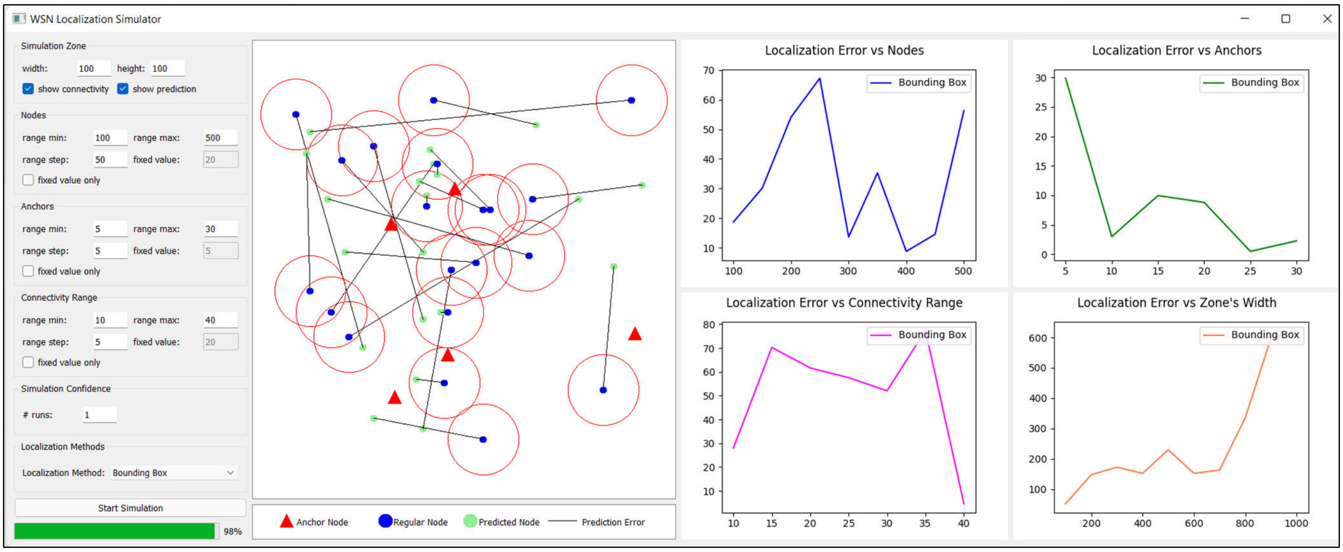


FIGURE 18. Interface of WSN localization simulator.

connectivity range and a step size for incremental adjustments. A fixed value can also be selected.

- *Simulation Confidence:* Users can influence the robustness of the simulation results by specifying the number of runs. This parameter likely affects the statistical confidence of the results and helps in assessing the reliability of the localization method being tested.
- *Localization Methods:* A dropdown menu titled "Localization Methods" allows the user to select the algorithm used for node localization. In the given example, the selected method is "Bounding Box."
- *Simulation Controls:* A green "Start Simulation" button prominently initiates the simulation process. Below this, a progress bar gives visual feedback on the simulation's progress, indicating what percentage of the simulation has been completed.

2) NETWORK VISUALIZATION PANE

The central area of the simulator displays the network graphically. Nodes are illustrated as dots, with anchor nodes in one color and regular nodes in another. Lines between nodes indicate connectivity, and prediction errors are represented by lines that connect actual node positions with their predicted positions based on the localization algorithm.

3) RESULTS PANE

The right side of the interface features four graphs that present the localization error metrics:

- *Localization Error vs Nodes:* This graph plots the error against the number of nodes, providing insight into how node density affects localization accuracy.
- *Localization Error vs Anchors:* It shows the impact of the number of anchor nodes on localization error.
- *Localization Error vs Connectivity Range:* This graph depicts how varying the range of connectivity influences the error, crucial for understanding the effect of node isolation or network density.
- *Localization Error vs Zone's Width:* Lastly, this graph examines the correlation between the physical size of the simulated area (zone's width) and the localization error, which could be vital for scaling the network in real-world applications.

Each graph bears a legend indicating that the plotted data points correspond to the localization method, implying the interface can compare multiple methods.

V. EXPERIMENTAL RESULTS AND ANALYSIS

The network simulator was used in the experimental phase to evaluate and illustrate the performance of different localization methods and then compare them with our hybrid method. The simulation is based on different parameters such as the number of anchors and the connectivity range (Table 3) in addition to the harmony search hyperparameters.

A. COMPARISON OF LOCALIZATION ERROR VS. NUMBER OF NODES

Fig. 19 presents a comparison of five different techniques for localization. The accuracy of this process is critical, and it is often measured using the Root Mean Square Error (RMSE). The lower the RMSE, the more accurate the localization. The graph plots RMSE against the number of nodes used in the network, with the number of nodes ranging from 100 to 500.

The techniques compared are trilateration, bounding box, harmony search, HSBB and BBHS. As the graph illustrates, trilateration does not perform as well as the other methods, showing a relatively high RMSE across all node numbers. This suggests that while trilateration may be simple, it may

TABLE 3. Network simulation parameters.

Parameters	Values/Range	Characteristics
Network Zone	100×100 m^2	Fixed
Number of Nodes	100 – 500	Variable
Number of Anchors	5 – 30	Variable
Connectivity Range	10 – 40 m	Variable
Simulation Runs	100	Fixed
HS Maximum Iterations	10	Fixed
HS Memory Size	10	Fixed
HS Consideration Rate	0.95	Fixed
HS Pitch Adjustment Rate	0.07	Fixed
HS Number of Nearby Anchors	5	Variable

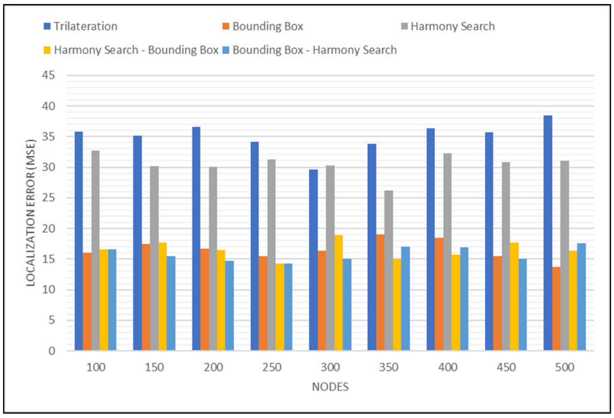


FIGURE 19. Peer techniques comparison of localization error vs number of nodes.

not be the most reliable for networks of the sizes tested. The bounding box technique encloses an area where the nodes are likely to be located and iteratively refines this area to improve accuracy. It is evident that the bounding box method outperforms trilateration, especially as the number of nodes increases. This implies that bounding box techniques may be more suited for larger networks. The harmony search algorithm showed a performance that surpasses trilateration. However, bounding box method consistently surpasses trilateration and harmony search across all numbers of nodes, indicating a more robust and adaptable approach to localization.

The HSBB seems to combine the strengths of both the bounding box and harmony search methods, leading to even lower RMSE values than when these methods are applied independently. Surprisingly, while one might expect method to deliver the lowest RMSE, it does not outperform the BBHS in most cases. It is, however, consistently more effective than either trilateration or the bounding box method alone.

Comparatively, it is clear that hybrid methods involving harmony search tend to yield better localization accuracy. The data strongly suggests that incorporating the adaptive, heuristic search properties of harmony search with the structured approach of a bounding box significantly reduces localization errors, particularly in larger networks.

An interesting trend observed is the impact of the number of nodes on the RMSE for different methods. While all methods show some increase in error as the number of nodes grows, the rate of increase is not uniform. For example, the trilateration method's error rate grows faster with the number of nodes than the other methods. This might be due to the geometric complexity and the increased probability of error propagation in larger networks when using trilateration.

Moreover, the graph illustrates that the harmony search and its hybrid with the bounding box are more scalable, as the increase in RMSE is less steep as the network size grows. This indicates that these methods are more stable and reliable for extensive networks, where precise localization is crucial.

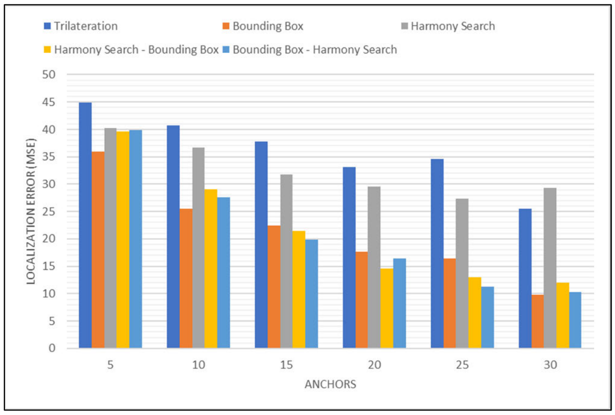


FIGURE 20. Peer techniques comparison of localization error vs number of anchors.

B. COMPARISON OF LOCALIZATION ERROR VS. NUMBER OF ANCHORS

Fig. 20 offers a comparative visualization of the peer techniques as they relate to the number of anchor nodes in a network ranging from 5 to 30. The trilateration method performs less effectively as the number of anchors increases, with a relatively high RMSE that does not improve significantly with more anchors. This suggests that beyond a certain point, simply increasing the number of anchor nodes does not necessarily enhance trilateration's accuracy, possibly due to cumulative distance measurement errors or geometric dilution of precision. While starting with high error rates at lower numbers of anchors, the bounding box method shows a marked improvement as the number of anchors increases. The initial high RMSE could be indicative of the bounding box method's sensitivity to having a sufficient number of anchor nodes to define an accurate initial boundary. As the number of anchors grows, the method can more reliably pinpoint the location within a progressively refined bounding area. Harmony search demonstrates a consistent RMSE across varying numbers of anchors. This suggests a robustness inherent in the harmony search algorithm, which does not rely as heavily on the number of anchor nodes but rather on the iterative optimization process it employs to refine localization estimates. It performs comparably well even with fewer anchor nodes. HSBB initially starts with a high error like the bounding box method, but its RMSE decreases more dramatically as the number of anchors increases. This indicates that the combination of both methods leverages the strengths of each: the structure provided by the bounding box and the adaptive optimization of harmony search. Lastly, BHSS appears to be less consistent across the range of anchor nodes. It begins with a high RMSE that reduces as more anchors are introduced, suggesting a dependency on having a certain threshold number of anchors to optimize localization accuracy effectively.

Upon comparing the techniques, it is clear that harmony search, whether standalone or combined with bounding box constraints, generally yields lower RMSEs, highlighting its effectiveness in localization tasks. In contrast, the trilateration method does not significantly benefit from additional anchor nodes beyond a certain number, indicating a potential limitation of this approach in environments where deploying a large number of anchor nodes is feasible.

A trend observable across the methods is that increasing the number of anchors up to a certain point improves localization accuracy, but this improvement plateaus or becomes inconsistent after surpassing a certain number of anchors. This could be due to overfitting, interference, or the complexity of the network topology exceeding the algorithms' ability to compensate. Fig. 20 indicates that deploying a moderate to a high number of anchor nodes, employing harmony search in conjunction with bounding box methods can significantly enhance localization accuracy. It also indicates that harmony search alone is a robust method that performs well regardless of the number of anchor nodes.

C. COMPARISON OF LOCALIZATION ERROR VS. CONNECTIVITY RANGE

Fig. 21 shows the impact of connectivity range on the localization error across peer techniques. The connectivity range, varying from 10 to 40 units, indicates the maximum distance over which nodes can communicate with each other or with anchor nodes.

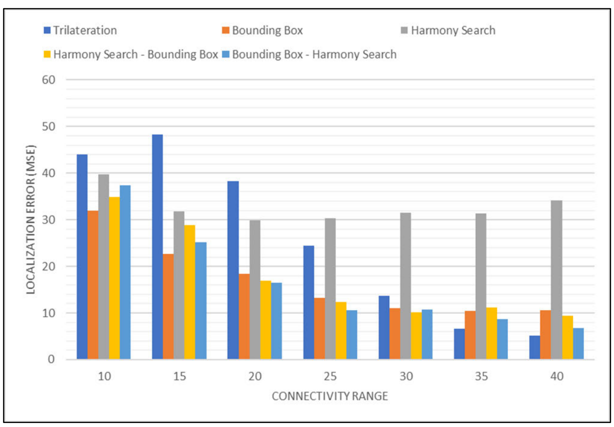


FIGURE 21. Peer techniques comparison of localization error vs connectivity range.

Starting with trilateration, there is a noticeable trend where the RMSE is initially very high at lower connectivity ranges and then sees a reduction as the range increases to 20 units. Beyond this, the error begins to climb again. This could be interpreted as trilateration benefiting from a moderate connectivity range, where nodes can adequately measure distances without excessive noise or errors. However, as connectivity increases further, the potential for error accumulation or inaccuracies in distance measurements might increase, leading to a higher RMSE. The bounding box method also starts with a high RMSE at low connectivity ranges, but it significantly improves as the range increases, reaching its lowest RMSE at a connectivity range of 25 units. Beyond this, the error slightly increases but remains relatively stable. This suggests that the bounding box method relies on a certain density of network connectivity to define an accurate initial area for localization, but after a certain point, further increases in range do not contribute to accuracy and may even introduce more complexity. The harmony search method, shows a consistent pattern of reducing RMSE as the connectivity range increases. The gradual descent in RMSE across all connectivity ranges implies that harmony search benefits from increased connectivity, likely because it can utilize more information to refine its optimization process for localization. HSBB shows an initial decrease in RMSE with increasing connectivity range, followed by a leveling off and even a slight increase in error at the highest connectivity range. The initial improvement may be due to the harmony search algorithm efficiently optimizing within the constraints provided by the bounding box method. However, the eventual increase in error at high connectivity ranges suggests there may be a limit to how much connectivity contributes to the accuracy of this combined approach. Lastly, BBHS starts with a very high RMSE at low connectivity ranges. As the connectivity increases, there is a substantial drop in error, suggesting that this hybrid technique requires a certain threshold of connectivity to function effectively. The error rate for this method is the lowest at intermediate ranges and begins to rise again at higher connectivity ranges, albeit not as sharply as some of the other methods.

The overall trend from Fig. 21 indicates that all techniques benefit from an increase in connectivity range up to a point. Beyond that optimal point, however, further increases in connectivity range do not necessarily result in better localization accuracy and may, in fact, lead to higher errors for certain methods. This could be due to various factors such as signal interference, multipath propagation, or computational complexity.

From an application perspective, this graph suggests that when deploying a network, there is an optimal connectivity range that maximizes localization accuracy. This optimal range seems to be around 20 to 30 units for most methods, with significant variations depending on the specific localization technique used.

D. COMPARISON OF LOCALIZATION ERROR VS. ZONE'S WIDTH

Fig. 22 presents an analysis of the effect of zone width on localization error across peer localization techniques. Beginning with trilateration, interesting pattern is shown: as the

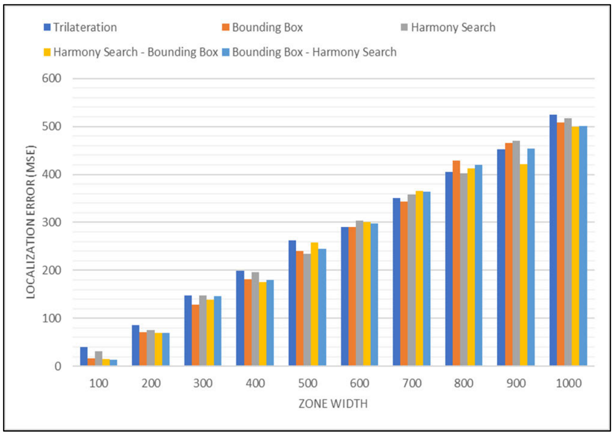


FIGURE 22. Peer techniques comparison of localization error vs zone' width.

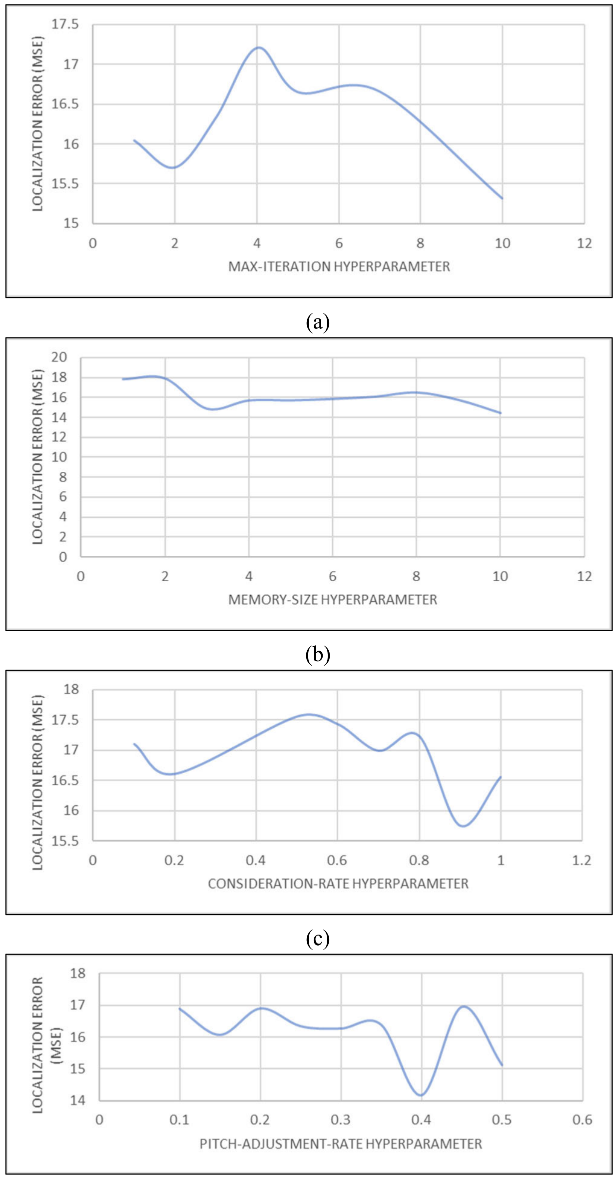
zone width increases, the RMSE initially decreases, suggesting improved accuracy, but after a certain point, the RMSE begins to rise again. This indicates that trilateration may be more effective in medium-sized zones, where the distances between nodes are neither too small to cause signal overlap nor too large to introduce significant errors in distance estimation. The bounding box method exhibits a steady increase in RMSE as the zone width expands. This method appears to work better in smaller zones, where defining a bounding box is more manageable and less likely to include large error margins. Harmony search shows a different trend: the RMSE remains relatively stable as the zone width increases. This stability suggests that the harmony search algorithm is less sensitive to zone size changes, maintaining its performance over a wide range of conditions. HSBB demonstrate a gradual increase in RMSE with larger zones, similar to the bounding box method but with generally lower error values. This suggests that while the bounding box provides a structural framework for the search, the harmony search's optimization capabilities can mitigate but not entirely overcome the challenges of increasing zone size. Finally, BBHS seems to offer a middle ground in terms of RMSE progression. The increase in RMSE is less pronounced than with the bounding box method alone but is more noticeable than with harmony search alone. This combination may offer a balance between structure and flexibility when it comes to handling varying zone widths.

Comparing the techniques, it is evident that no single method is universally superior across all zone widths. Each has its strengths and weaknesses that become more or less pronounced depending on the size of the zone. For smaller zones, the bounding box method appears adequate, but as zones increase in size, harmony search exhibits a distinct advantage in maintaining accuracy.

E. FINE-TUNING HARMONY SEARCH HYPERPARAMETERS

Fine-tuning hyperparameters for Harmony Search involves systematically exploring the hyperparameter space to optimize parameters like maximum iterations, consideration rate, pitch adjustment rate, and memory size. This process begins with an understanding of each parameter's role, typically starting with default or common values. Sensitivity analysis identifies influential parameters, guiding subsequent exploration.

This iterative exploration and refinement process, informed by empirical observations and domain expertise, ensures the effective optimization of HS across diverse tasks. By continually exploring and refining the hyperparameter space, practitioners enhance HS's adaptability and performance in various optimization scenarios.



(a) (b) (c) (d)

FIGURE 23. Fine tuning Harmony Search Hyperparameters: (a) Maximum Iterations (b) Memory Size (c) Consideration Rate (d) Pitch Adjustment Rate.

Fig. 23 (a) presents the relationship between the max-iteration hyperparameter and the localization error (RMSE). Initially, as the max-iteration value increases from 0 to 2, there's a noticeable decrease in the error, suggesting early iterations significantly improve localization accuracy. However, between 2 to 6 iterations, the error fluctuates, reaching a peak around iteration 4. This indicates that simply adding iterations does not monotonically improve performance and may introduce overfitting or instability. Post iteration 6, the error gradually decreases, stabilizing and reaching its minimum at iteration 10. This suggests that fine-tuning the max-iteration parameter to 10 yields the most accurate localization results, balancing computational effort against precision. The uptick in error at 10 hints at diminishing returns thereafter. Therefore, setting the max-iteration to 10 optimizes the balance between accuracy and computational efficiency.

Fig. 23 (b) displays how the memory-size hyperparameter affects the localization error. Initially, we observe a slight decrease in error as memory size increases from 0 to 2, suggesting more memory allocation up to a certain point contributes to better localization. From 2 to around 6, the error rate plateaus, indicating that within this range, changes in memory size have a negligible effect on error reduction. As the memory size approaches 10, there is a subtle yet consistent decline in error, which infers that augmenting memory size up to 10 improves the model's performance. This trend suggests that a memory size of 10 is optimal for minimizing localization error while also implying efficient resource utilization. Beyond this point, the error rate levels off, indicating that further increases in memory size do not translate into significant improvements in accuracy and could be redundant or inefficient.

Fig. 23 (c) depicts the influence of the HS consideration rate hyperparameter on the localization error. It begins with a sharp decline in error as the rate increases from 0 to 0.2, indicating that a small amount of consideration leads to substantial improvements in localization accuracy. Between 0.2 and 0.6, the error oscillates, suggesting a complex relationship where adjustments to the consideration rate can either improve or degrade performance. Beyond 0.6, the trend again shows a general decline in error, with a noticeable dip at 0.9. This marks the point at which the consideration rate is optimally fine-tuned, achieving the lowest error and indicating the most balanced and effective setting for this hyperparameter. Post 0.9, the error slightly increases, hinting that further increments might lead to overcompensation and thus reduce performance.

Fig. 23 (d) shows the impacts of pitch adjustment rate hyperparameter on the localization error. The error begins relatively high and demonstrates a gentle decrease as the pitch adjustment rate is increased from 0 to 0.2. The error then increases slightly up to a rate of 0.3, implying that a higher rate initially does not contribute to better localization. However, the most significant observation is the marked reduction in error at the 0.4 rate, indicating that this particular value of the hyperparameter substantially enhances the accuracy of localization. Beyond this rate, the error ascends sharply, suggesting that a rate higher than 0.4 leads to poorer performance, likely due to over-adjustment. Hence, a pitch adjustment rate of 0.4 appears to be the optimal setting, where the error is minimized, and the system achieves its best balance of adjustment for accurate localization.

F. COMPARISON AFTER FINE TUNING HARMONY SEARCH HYPERPARAMETERS

Fine-tuning hyperparameters for Harmony Search involves systematically exploring the hyperparameter space to optimize parameters like maximum iterations, consideration rate, pitch adjustment rate, and memory size. This process begins with an understanding of each parameter's role, typically starting with default or common values. Sensitivity analysis identifies influential parameters, guiding subsequent exploration.

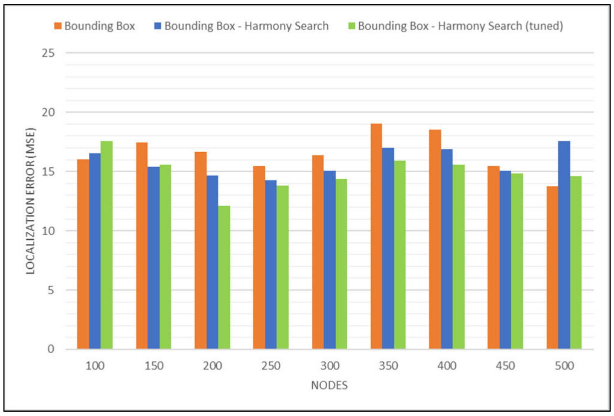


FIGURE 24. Effect of fine tuning over localization error vs number of nodes.

Fig. 24 showed that that the fine-tuned BBHS method consistently outperforms the other two methods across the range of nodes tested. Notably, as the number of nodes increases, the fine-tuned method maintains a relatively lower error rate, showcasing its scalability and robustness. This suggests that the fine-tuning process has successfully optimized the algorithm to handle larger networks more efficiently, where maintaining low localization error is crucial for tasks such as sensor network deployment and coordination.

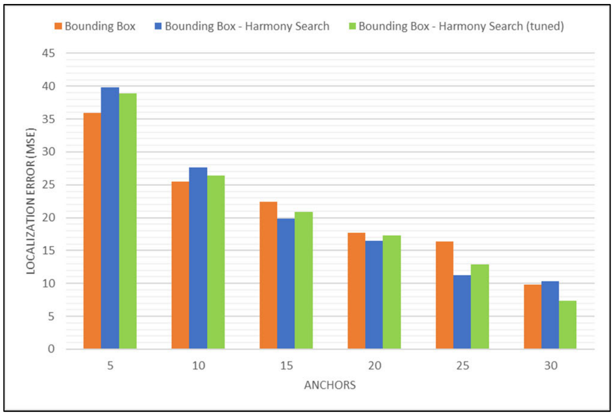


FIGURE 25. Effect of fine tuning over localization error vs number of anchors.

Fig. 25, demonstrates superior performance of tuned BBHS, particularly as the number of anchors increases. This trend is significant because it indicates that the fine-tuned method can leverage additional reference points more effectively than the non-tuned methods. The capacity to reduce error with more anchors is a valuable quality in real-world applications where anchors can be viewed as known positions in a network used to improve the accuracy of determining unknown node positions.

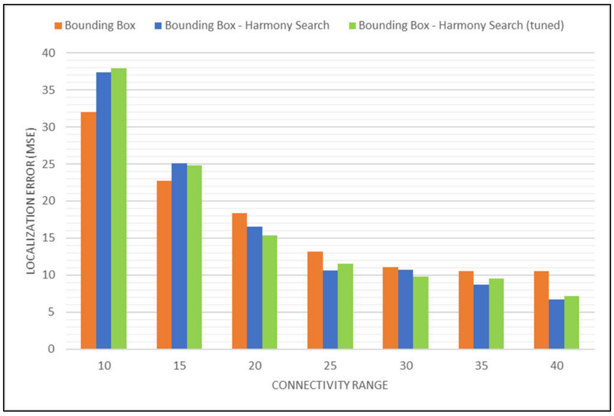


FIGURE 26. Effect of fine tuning over localization error vs connectivity range.

Fig. 26 displays how the localization error changes with the connectivity range. It is interesting to note that the fine-tuned method does not always have the lowest error; however, the error decreases notably as the connectivity range increases to 40. This implies that the fine-tuned method could be optimized for scenarios where a higher connectivity range is available. It underscores the importance of considering the connectivity range when deploying a localization system as it can greatly impact the performance of the localization technique.

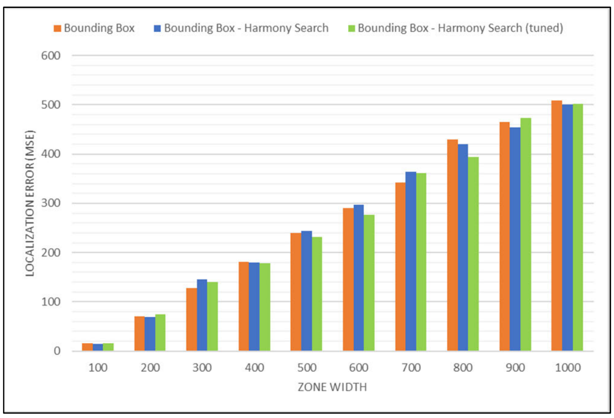


FIGURE 27. Effect of fine tuning over localization error vs zone's width.

In Fig. 27, a dramatic increase in error is observed for all methods as zone width increases, with the fine-tuned method showing a relatively gradual rise. This indicates that while all methods are affected by increased zone width, fine-tuning has provided a degree of resilience. This resilience is crucial in environments with wide zones where precise localization is challenging but essential.

G. ANALYSIS OF FINE-TUNING ENHANCEMENTS

Across Figures 24-27, we observe a common theme: the fine-tuned BBHS method offers a significant reduction in localization error when compared to the other two methods. This is a clear indicator that the fine-tuning process has yielded a more effective hybrid localization method. Fine-tuning has likely involved adjusting hyperparameters, enhancing the harmony search algorithm components, or improving the bounding box algorithm's sensitivity and precision. The fine-tuning of the hybrid technique seems to prepare the method for practical deployment better than its counterparts. In real-world scenarios, where variables such as the number of nodes, anchors, connectivity range, and zone width can drastically vary and be unpredictable, a method that can maintain a lower localization error across these variations is invaluable.

VI. CONCLUSION

The importance of node localization in wireless sensor networks cannot be overstated, as it underpins the effectiveness and utility of these networks across a broad spectrum of applications. From environmental monitoring and disaster management to smart cities and military surveillance, the accurate positioning of sensor nodes is crucial for data integrity, decision-making processes, and the execution of tasks dependent on spatial information. The drive for precision in localization has spurred a variety of methods, each striving to refine the accuracy and reliability of the positioning under different network conditions.

In this paper, we explained the development a WSN localization simulator to serve as a tool for network designers, allowing them to simulate and visualize different localization algorithms' performance in a controlled environment. Through such simulation, one can understand the behavior of WSNs under various conditions, optimize network parameters for improved accuracy and efficiency, and potentially identify the best-suited localization methods for specific applications or environments. The interface was designed for ease of use, enabling both novice and expert users to set up a simulation quickly, visualize complex data intuitively, and obtain actionable insights into the dynamics of WSN localization.

We implemented some traditional methods like trilateration which has provided foundational techniques, leveraging geometric principles to deduce node positions. The accuracy of these methods, however, is often compromised in real-world scenarios by factors such as signal attenuation, environmental obstructions, and noise, which can significantly skew distance measurements and, consequently, localization accuracy. Bounding Box emerged as an alternative, aiming to mitigate some of the vulnerabilities of distance-dependent methods by estimating node positions within the confines of a virtual geometric space defined by anchor nodes. Despite its innovative approach, the Bounding

Box method still encounters challenges, particularly in environments with sparse node distributions or when scaling up to larger areas, which can lead to less precise localization.

The hybridization of the Bounding Box method with the Harmony Search algorithm, resulting in BBHS and HSBB, has marked a significant step forward, especially with BBHS. These methods blend the strengths of geometric containment with an optimization process inspired by the improvisation of musical ensembles, where the algorithm iteratively searches for a harmonious (optimal) state—here, the accurate position of nodes. BBHS demonstrated the potential of this hybrid approach, showing notable improvements over both trilateration and the standalone Bounding Box method. It leveraged the global optimization prowess of the harmony search to refine the initial estimates, consistently reducing localization errors across a range of network configurations. BBHS introduced enhancements that further reduced localization errors and stabilized performance across varying numbers of nodes, anchor densities, and connectivity ranges. The analyses of BBHS performance, after fine-tuning HS hyperparameters, illustrated its impressive adaptability and precision, with consistent results that suggest an algorithm refined to the point of being relatively unaffected by changes in the network environment—a significant achievement for any localization method.

The conclusion drawn from the literature review and the experimental work is clear: hybrid methods represent the cutting edge of localization in WSNs. They not only address the limitations of earlier methods but also open the door to more robust and reliable network operations across diverse applications. By effectively harnessing the power of optimization algorithms, these hybrid techniques have demonstrated that they can provide high accuracy while being less susceptible to environmental variables that traditionally hinder node localization.

Future iterations of these methods might incorporate additional layers of sophistication, possibly integrating machine learning to adapt and learn from the network environment dynamically. As WSNs continue to grow in complexity and scale, the drive for improved localization methods will undoubtedly persist, with the ultimate goal of achieving autonomous, self-organizing networks that can operate with minimal human intervention and maximum efficiency.

REFERENCES

[1] A. Paul and T. Sato, "Localization in wireless sensor networks: A survey on algorithms, measurement techniques, applications and challenges," *J. Sensor Actuator Netw.*, vol. 6, no. 4, p. 24, Oct. 2017.

[2] N. A. Alrajeh, M. Bashir, and B. Shams, "Localization techniques in wireless sensor networks," *Int. J. Distrib. Sensor Netw.*, vol. 9, no. 6, Jun. 2013, Art. no. 304628.

[3] X. Yan, Q. Luo, Y. Yang, S. Liu, H. Li, and C. Hu, "ITL-MEPOSA: Improved trilateration localization with minimum uncertainty propagation and optimized selection of anchor nodes for wireless sensor networks," *IEEE Access*, vol. 7, pp. 53136–53146, 2019.

[4] V. Ramadurai and M. L. Sichitiu, "Localization in wireless sensor networks: A probabilistic approach," in *Proc. Int. Conf. Wireless Netw.*, 2003, pp. 275–281.

[5] D. A. Tran and T. Nguyen, "Localization in wireless sensor networks based on support vector machines," *IEEE Trans. Parallel Distrib. Syst.*, vol. 19, no. 7, pp. 981–994, Jul. 2008.

[6] A. Gopakumar and L. Jacob, "Localization in wireless sensor networks using particle swarm optimization," in *Proc. IET Conf. Wireless, Mobile Multimedia Netw.* Edison, NJ, USA: IET, 2008, pp. 227–230.

[7] R. L. Moses, D. Krishnamurthy, and R. M. Patterson, "A self-localization method for wireless sensor networks," *EURASIP J. Adv. Signal Process.*, vol. 2003, no. 4, pp. 1–11, Mar. 2003.

[8] S. Yun, J. Lee, W. Chung, E. Kim, and S. Kim, "A soft computing approach to localization in wireless sensor networks," *Expert Syst. Appl.*, vol. 36, no. 4, pp. 7552–7561, May 2009.

[9] X. Chen and B. Zhang, "Improved DV-hop node localization algorithm in wireless sensor networks," *Int. J. Distrib. Sensor Netw.*, vol. 8, no. 8, Aug. 2012, Art. no. 213980.

[10] S. Arora and S. Singh, "Node localization in wireless sensor networks using butterfly optimization algorithm," *Arabian J. Sci. Eng.*, vol. 42, no. 8, pp. 3325–3335, Aug. 2017.

[11] S. Amri, F. Khelifi, A. Bradai, A. Rachedi, M. L. Kaddachi, and M. Atri, "A new fuzzy logic based node localization mechanism for wireless sensor networks," *Future Gener. Comput. Syst.*, vol. 93, pp. 799–813, Apr. 2019.

[12] 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.

[13] A. Kumar, A. Khosla, J. S. Saini, and S. Singh, "Meta-heuristic range based node localization algorithm for wireless sensor networks," in *Proc. Int. Conf. Localization GNSS*, Jun. 2012, pp. 1–7.

[14] Y. Chen, S. Lu, J. Chen, and T. Ren, "Node localization algorithm of wireless sensor networks with mobile beacon node," *Peer-to-Peer Netw. Appl.*, vol. 10, no. 3, pp. 795–807, May 2017.

[15] H. Wu, J. Liu, Z. Dong, and Y. Liu, "A hybrid mobile node localization algorithm based on adaptive MCB-PSO approach in wireless sensor networks," *Wireless Commun. Mobile Comput.*, vol. 2020, pp. 1–17, Jun. 2020.

[16] Z. Hao, J. Dang, Y. Yan, and X. Wang, "A node localization algorithm based on Voronoi diagram and support vector machine for wireless sensor networks," *Int. J. Distrib. Sensor Netw.*, vol. 17, no. 2, Feb. 2021, Art. no. 155014772199341.

[17] J. Li, M. Gao, J.-S. Pan, and S.-C. Chu, "A parallel compact cat swarm optimization and its application in DV-hop node localization for wireless sensor network," *Wireless Netw.*, vol. 27, no. 3, pp. 2081–2101, Apr. 2021.

[18] N. Liu, J.-S. Pan, J. Wang, and T.-T. Nguyen, "An adaptation multi-group quasi-affine transformation evolutionary algorithm for global optimization and its application in node localization in wireless sensor networks," *Sensors*, vol. 19, no. 19, p. 4112, Sep. 2019.

[19] S. Phoemphon, C. So-In, and N. Leelathakul, "A hybrid localization model using node segmentation and improved particle swarm optimization with obstacle-awareness for wireless sensor networks," *Expert Syst. Appl.*, vol. 143, no. 2020, Apr. 2020, Art. no. 113044.

[20] K. Sabale and S. Mini, "Localization in wireless sensor networks with mobile anchor node path planning mechanism," *Inf. Sci.*, vol. 579, pp. 648–666, Nov. 2021.

[21] L. Wang, M. J. Er, and S. Zhang, "A kernel extreme learning machines algorithm for node localization in wireless sensor networks," *IEEE Commun. Lett.*, vol. 24, no. 7, pp. 1433–1436, Jul. 2020.

[22] S. K. Rout, P. K. Mohapatra, A. K. Rath, and B. Sahu, "Node localization in wireless sensor networks using dynamic genetic algorithm," *J. Appl. Res. Technol.*, vol. 20, no. 5, pp. 520–528, Oct. 2022.

[23] H. Zhang, J. Yang, T. Qin, Y. Fan, Z. Li, and W. Wei, "A multi-strategy improved sparrow search algorithm for solving the node localization problem in heterogeneous wireless sensor networks," *Appl. Sci.*, vol. 12, no. 10, p. 5080, May 2022.

[24] R. Tan, Y. Li, Y. Shao, and W. Si, "Distance mapping algorithm for sensor node localization in WSNs," *Int. J. Wireless Inf. Netw.*, vol. 27, no. 2, pp. 261–270, Jun. 2020.

[25] V. Annepu and R. Anbazhagan, "Implementation of an efficient extreme learning machine for node localization in unmanned aerial vehicle assisted wireless sensor networks," *Int. J. Commun. Syst.*, vol. 33, no. 10, p. e4173, Jul. 2020.

[26] W. Ingabire, H. Larijani, R. M. Gibson, and A.-U.-H. Qureshi, "Outdoor node localization using random neural networks for large-scale urban IoT LoRa networks," *Algorithms*, vol. 14, no. 11, p. 307, Oct. 2021.

[27] A. Kumar, A. Khosla, J. S. Saini, and S. S. Sidhu, ''Range-free 3D node localization in anisotropic wireless sensor networks,'' *Appl. Soft Comput.*, vol. 34, pp. 438–448, Sep. 2015.

[28] H. M. Kanoosh, E. H. Houssein, and M. M. Selim, ''Salp swarm algorithm for node localization in wireless sensor networks,'' *J. Comput. Netw. Commun.*, vol. 2019, pp. 1–12, Feb. 2019.

[29] M. T. S. Latha and K. B. Rekha, ''A hybrid metaheuristic bat algorithm with simulated annealing for node localization in wireless sensor network,'' *Math. Statistician Eng. Appl.*, vol. 71, no. 3, pp. 845–861, 2022.

[30] T. S. Latha, K. B. Rekha, and A. W. Ferede, ''Lightweight APIT with bat optimization with simulated annealing localization for resource-constrained sensor networks,'' *Wireless Commun. Mobile Comput.*, vol. 2023, pp. 1–11, May 2023.

[31] S. Yu, J. Zhu, and C. Lv, ''A quantum annealing bat algorithm for node localization in wireless sensor networks,'' *Sensors*, vol. 23, no. 2, p. 782, Jan. 2023.

[32] Y. Cao, Y. Zhao, and F. Dai, ''Node localization in wireless sensor networks based on quantum annealing algorithm and edge computing,'' in *Proc. Int. Conf. Internet Things (iThings) IEEE Green Comput. Commun. (GreenCom) IEEE Cyber, Phys. Social Comput. (CPSCom) IEEE Smart Data (SmartData)*, Jul. 2019, pp. 564–568.

[33] R. Rajakumar, J. Amudhavel, P. Dhavachelvan, and T. Vengattaraman, ''GWO-LPWSN: Grey wolf optimization algorithm for node localization problem in wireless sensor networks,'' *J. Comput. Netw. Commun.*, vol. 2017, no. 1, 2017, Art. no. 7348141.

[34] R. Ahmad, W. Alhasan, R. Wazirali, and N. Aleisa, ''Optimization algorithms for wireless sensor networks node localization: An overview,'' *IEEE Access*, vol. 12, pp. 50459–50488, 2024.

[35] W. Osamy, A. M. Khedr, A. Salim, A. I. A. Ali, and A. A. El-Sawy, ''Coverage, deployment and localization challenges in wireless sensor networks based on artificial intelligence techniques: A review,'' *IEEE Access*, vol. 10, pp. 30232–30257, 2022.

[36] V. Annepu, D. R. Sona, C. V. Ravikumar, K. Bagadi, M. Alibakhshikenari, A. A. Althuwayb, B. Alali, B. S. Virdee, G. Pau, I. Dayoub, C. H. See, and F. Falcone, ''Review on unmanned aerial vehicle assisted sensor node localization in wireless networks: Soft computing approaches,'' *IEEE Access*, vol. 10, pp. 132875–132894, 2022.

[37] L. A. M. Hernández, S. P. Arteaga, G. S. Pérez, A. L. S. Orozco, and L. J. G. Villalba, ''Outdoor location of mobile devices using trilateration algorithms for emergency services,'' *IEEE Access*, vol. 7, pp. 52052–52059, 2019.

[38] W. Ma, K. Li, and G. Wang, ''Location-aware box reasoning for anchor-based single-shot object detection,'' *IEEE Access*, vol. 8, pp. 129300–129309, 2020.

[39] Z. W. Geem, J. H. Kim, and G. V. Loganathan, ''A new heuristic optimization algorithm: Harmony search,'' *Simulation*, vol. 76, no. 2, pp. 60–68, Feb. 2001.

[40] D. Manjarres, J. D. Ser, S. Gil-Lopez, M. Vecchio, I. Landa-Torres, and R. Lopez-Valcarce, ''On the application of a hybrid harmony search algorithm to node localization in anchor-based wireless sensor networks,'' in *Proc. 11th Int. Conf. Intell. Syst. Design Appl.*, Nov. 2011, pp. 1014–1019.

MOHAMMED OMARI (Member, IEEE) received the bachelor's degree from the University of Oran, Algeria, and the master's and Ph.D. degrees in computer science from New Mexico Institute of Mining and Technology, USA. In January 2022, he joined the Department of Computer Science and Engineering, American University of Ras Al Khaimah, where he is currently a Full Professor. Before his current position, he was a Faculty Member with the Ahmed Draia University of Adrar, Algeria, where he was the Dean of the Faculty of Sciences and Engineering Sciences. His research interests include wireless networks, adhoc networks, clustering of sensor networks, routing protocols, image and video compression, security, and cryptography. He actively collaborates with international research laboratories to develop wireless sensor network protocols using heuristic methods.

MOHAMMED KADDI received the Magister degree in industrial and automatic computing and the Ph.D. degree in electrical engineering from the University of Bechar, Algeria, in 2014 and 2019, respectively. He is currently an Associate Professor with the Ahmed Draia University of Adrar, Algeria. His work focuses on developing innovative solutions to enhance the efficiency and effectiveness of wireless sensor networks and related technologies. His research interests include energy conservation in wireless sensor networks, wireless localization techniques, networking, mobile communication, artificial intelligence, machine learning, and data grid management.

KHOULOUD SALAMEH received the Ph.D. degree in computer science and control engineering, automation, and robotics from the University of the Basque Country, Spain, in 2017. She is currently an Associate Professor with the Computer Science and Engineering Department, American University of Ras Al Khaimah (AURAK), United Arab Emirates. Before joining AURAK, she was engaged as a Postdoctoral Researcher in computer science with the University of Pau and Pays de l'Adour, France. Her teaching portfolio includes courses in software design and engineering, algorithms analysis and design, and human–computer interaction. Her research interests include data representation and modeling, optimization of digital ecosystems, and the development of ontologies.

ALI ALNOMAN received the Ph.D. degree in electrical and computer engineering from Toronto Metropolitan University (formerly Ryerson University), Toronto, Canada, in 2019. He joined the Wireless Networks and Communications Research Laboratory, Toronto Metropolitan University, in 2015. From 2017 to 2020, he was a Sessional Professor with the Faculty of Applied Science and Technology, Sheridan College, Canada. He is currently an Assistant Professor with the Department of Computer Science and Engineering, American University of Ras Al Khaimah, United Arab Emirates. His research interests include machine learning, the IoT, energy efficiency, smart communication systems, and cloud-edge computing. During the Ph.D. studies, he was awarded the Graduate Research Excellence Award for two consecutive years and the Doctoral Completion Award by Toronto Metropolitan University.

KHADRA ELFATMI received the B.S. and M.S. degrees in computer science from the Ahmed Draia University of Adrar, Algeria, in 2022. Her research interests include networking, clustering protocols, and node localization algorithms.

FADILA BAARAB received the B.S. and M.S. degrees in computer science from the Ahmed Draia University of Adrar, Algeria, in 2022. Her research interests include networking, clustering protocols, and node localization algorithms.