scientific reports

OPEN

Energy efficient and robust node localization in WSNs using LSTM optimized DV hop framework to mitigate multihop localization errors

Amjad Rehman[1], Tariq Mahmood[1,2], Tahani Jaser Alahmadi[3]✉, Ahmed S. Almasoud[1] & Tanzila Saba[1]

Wireless Sensor Networks (WSNs) are distributed sensor nodes that sense data from their surroundings and relay it to a central network for processing and analysis. Sensor localization is a crucial technique in WSNs, enabling precise positions of target nodes based on environmental signal perception. However, achieving high accuracy in node localization remains a challenge. This study introduces an improved DV-Hop positioning algorithm that integrates Long Short-Term Memory (OLSTM-DVHop) networks to enhance node position predictions. The algorithm processes original data through filtering, analysis, and feature extraction to improve predicted node positions. The study analyzed errors using a standard DV-Hop algorithm and designed a robust architecture for WSN positioning. Simulation experiments validated the proposed improvements, aligning with the algorithm's accuracy requirements. The proposed error correction mechanism addresses uneven error distribution in the DV-Hop algorithm, adjusting the positions of nodes with significant deviations, reducing errors, and enhancing the positioning process's reliability and accuracy. The effectiveness of the proposed algorithm is evaluated by comparing it with other localization algorithms across different terrain types. The improved DV-Hop algorithm significantly reduces localization errors and offers superior accuracy, outperforming other algorithms in various experimental scenarios.

Keywords LSTM, DV-Hop, Positioning accuracy, Error correction

With the continuous development of computer networks, information technology has gradually become a bridge connecting the world. In this context, WSNs are crucial in linking humans and objects, forming the foundation for realizing the Internet of Things (IoT)[1]. WSNs composed of distributed sensor nodes are essential for collecting and transmitting environmental data for processing and analysis. Wireless sensors are now indispensable across various modalities, including measuring gas, pressure, temperature, humidity, liquid levels, wind speed, and flow rate. These sensors have become essential components in a broad spectrum of industrial and agricultural applications[2]. Consequently, the WSNs model has evolved significantly to meet the specific and complex requirements of these fields[3]. In WSNs, nodes can autonomously form networks without central control. Data is transmitted from one node to another using hop-by-hop communication, enabling efficient data transmission[4]. To ensure the effectiveness and timeliness of data transmission within the network, it is vital to monitor and report the location of events. Thus, designing practical localization algorithms is a key factor in maximizing the utility of WSNs. WSN localization algorithms have garnered significant attention and in-depth research from scholars[5]. The IoT represents the future direction of information technology development and has emerged as one of this century's most prominent and researched areas. WSNs are a fundamental component of IoT, characterized by spatially distributed autonomous sensor networks connected via electromagnetic waves[6]. WSNs are evolving towards low power consumption, low cost, multifunctionality, and high integration. WSNs offer monitoring, management, and control services through various technical means by collecting, processing, and transmitting

[1]Artificial Intelligence and Data Analytics (AIDA) Lab, CCIS, Prince Sultan University, Riyadh, 11586, Saudi Arabia. [2]Faculty of Information Sciences, University of Education, Vehari Campus, 61161, Vehari, Pakistan. [3]Department of Information Systems, College of Computer and Information Sciences, Princess Nourah bint Abdulrahman University, P.O. Box 84428, 11671 Riyadh, Saudi Arabia. ✉email: tjalahmadi@pnu.edu.sa

data[7]. Localization technology in WSNs is critical as it allows the detection of events to be meaningful by knowing their precise locations, thus supporting a wide range of applications such as environmental monitoring, intelligent transportation, and logistics management[8]. Consequently, WSN localization technology has become a hot topic in WSN research. WSN localization techniques are primarily categorized based on measured node location information and non-ranging methods[9]. The former requires nodes to be pre-aware of their location information. Common distance measurement techniques include Received Signal Strength Indicator (RSSI), Time of Arrival (ToA), Time Difference of Arrival (TDoA), and Angle of Arrival (AoA)[10]. On the other hand, non-ranging localization methods do not require additional hardware for distance measurement. Hence, non-ranging methods are more practical and have broader application prospects[11]. Scholars have proposed numerous localization algorithms for WSNs, often based on different distance measurement techniques and employing various node localization strategies and optimization methods. However, due to the complexity of wireless signals and environmental interference, positioning accuracy remains constrained[12]. Therefore, improving positioning accuracy and reducing localization errors remain critical research challenges in WSN localization technology. The DV-Hop localization algorithm is a WSN localization technique that utilizes the correlation between distance vectors and network node connectivity to determine node locations[13]. The DV-Hop algorithm is simple and effective, but its accuracy is influenced by multi-hop errors, node density, and the proportion of anchor nodes, necessitating further improvement[14]. In studying WSN localization algorithms, researchers have focused on several key issues, including the selection of anchor nodes, which can significantly enhance positioning accuracy[15]. The impact of factors such as node density, environmental interference, and sensor positioning on localization errors[16] and issues related to the real-time performance and energy consumption of the algorithm, as the DV-Hop algorithm requires multiple communications between nodes, leading to increased energy consumption and communication delays. Therefore, exploring ways to maintain accuracy while reducing energy consumption and communication delays is crucial. Despite the advantages of the DV-Hop algorithm, its practical application is limited by significant positioning errors and challenges related to real-time performance and energy consumption[17]. Addressing these issues is crucial to improving the utility of WSNs. In response to these challenges, this paper proposes an enhanced DV-Hop positioning algorithm for WSNs, incorporating LSTM networks to improve the accuracy of node position predictions. The proposed algorithm filters, analyzes, and extracts features from raw data to mitigate errors and enhance positioning precision. Additionally, a novel error correction mechanism is introduced within the OLSTM-DVHop framework to address the uneven distribution of error regions. This mechanism adjusts the positioning data of nodes with significant deviations, further reducing positioning errors and improving the reliability of node localization. The proposed method demonstrates superior accuracy and effectiveness through simulation experiments and comparative analysis with existing algorithms, particularly in complex and large-scale network environments.

Research challenges

The advancements in WSNs have offered substantial benefits in multiple sectors, from environmental monitoring to military operations. Achieving precise node localization within these networks is a challenging issue. Although the conventional DV-Hop localization algorithm is widely utilized for its simplicity and cost-effectiveness, it poses severe limitations that diminish its efficacy in real-world applications. One of the primary challenges stems from the algorithm's inherent susceptibility to localization errors due to its dependence on multi-hop distance estimations. Some of these errors are heightened by nodes' irregular distribution, density fluctuation, and environmental interference. Besides that, the algorithm struggles with energy efficiency and computational complexity issues, as it requires nodes to engage in multiple rounds of communication, leading to increased energy consumption and delayed processing times. One of the main issues is that error regions are not uniformly distributed over the network, squeezing some areas and dramatically worsening the localization accuracy. Consequently, coping with these critical issues is vital to improving the functionality of WSNs in various advanced and complex contexts in terms of accuracy, efficiency, and reliability. This research proposes a framework to optimize the DV-Hop algorithm by integrating LSTM networks. It aims to reduce localization errors and leads to energy efficiency, ensuring consistent accuracy across network conditions.

Novel contribution

The study improves the accuracy and reliability of sensor localization in WSNs, which is significant in environmental monitoring, smart cities, and security surveillance. The conventional localization algorithms, such as DV-Hop, suffer from substantial positioning errors due to uneven node distribution and inaccurate hop distance estimations, necessitating enhanced approaches to address these shortcomings.

- This research develops an enhanced localization algorithm that integrates LSTM networks with DV-Hop to filter, analyze, and extract features from raw data, significantly enhancing the precision of node position predictions.
- The suggested approach highlights the primary sources of error in traditional DV-Hop algorithms by designing a robust architecture tailored to WSN requirements, including error analysis and a correction mechanism that compensates for uneven error distribution.
- The study introduces a novel error correction strategy to improve positioning accuracy by adjusting node positions with significant deviations, considering the impact of node conditions on positioning errors.
- The proposed algorithm proved outstanding efficacy using various terrain types, demonstrating superior accuracy, stability, and average positioning error performance.
- The improved algorithm offers a more reliable and accurate solution for node localization in complex and dynamic WSN environments and significantly reduces positioning errors,

This research incorporates a new use of deep learning into WSN localization to supplement or replace some of the conventional algorithms. It provides the foundation for constructing a firmer and more accurate node positioning theory for practical use.

Paper organization

Section "Literature review" reviews WSNs, DV-Hop challenges, and localization techniques, highlighting gaps addressed by this study. Section "Methodology" outlines the methodology, including the improved DV-Hop algorithm with LSTM integration, covering architecture, pre-processing, and training. Section "Result" depicts the experimental setup, findings, and comparative analysis of the OLSTM-DVHop algorithm. Finally, section "Discussion and conclusion" summarizes key findings, limitations, and future research directions.

Literature review

The development of WSN localization algorithms has led to the emergence of several distance-based localization techniques, such as ultrasound, infrared, and GPS, which can directly measure distance or position between nodes. In the early 21st century, a team at Microsoft introduced the RADAR system[18], which utilized signal strength and propagation time to estimate node positions, achieving a certain degree of accuracy.

Thao et al.[19] designed a robust prediction model based on LSTM and RNN architecture to boost the cleanliness of school lavatories. The proposed model optimizes cleaning scheduling duties using real-time sanitation conditions obtained from a WSN, achieving RMSE values of 3.32 and 2.85 for LSTM and RNN, respectively. Kanwar et al.[20] optimized the average hop distance and upgraded auxiliary anchor nodes to improve localization coverage. However, calculating node positions using genetic algorithms led to excessive iteration counts, increasing the computational overhead for nodes. Penghong et al.[21] propose a DV-Hop localization method for WSN based on distance estimation using multinode (DEMN) and hop loss. DEMN calculates distance expectations between unknown nodes using information from the cross-domain, narrowing the search space. Hop loss mitigates the discrepancy between real and predicted hops. The Euclidean distance loss calculated by DEMN and hop loss are incorporated in a multi-objective optimization model. The empirical findings reveal that the suggested approach attains 86.11% location accuracy in a randomly distributed network, 6.05% higher than DEM-DV-Hop, with DEMN and hop loss contributing 2.46% and 3.41%, respectively. Zhou et al.[22] introduced the APIT (Approximate Point-In-Triangulation Test) model, which enhanced positioning accuracy by measuring distances and angles between nodes and using triangulation techniques. AbouRjeily et al.[23] proposed innovative sensor deployment strategies to reduce sensor number while maintaining high network quality and coverage. To address these challenges and improve the utility of WSNs, future research must focus on enhancing localization algorithms' accuracy, energy efficiency, and reliability. Such advancements will be crucial for their successful application in real-world production and daily life scenarios. Messous et al.[24] proposed an improved recursive DV-Hop positioning algorithm, refining the second and third steps for more accurate node position calculations. Latha et al.[25] proposed a modified APIT algorithm for node localization in WSNs by dividing the application area into overlapping and non-overlapping subregions. This approach addresses challenges such as small areas and narrow triangles. The APIT algorithm uses beacon signals from anchor nodes to determine target node positions. The experiment reveals that the proposed APIT with the Bat-SA algorithm performs well, achieving more even node distribution, reduced node positioning error, and lower latency than the conventional APIT. Phoemphon et al.[26] employed a hybrid genetic-particle swarm algorithm to boost localization precision. However, this extensive bioinspired computation significantly increased the energy consumption and localization time, making it potentially unsuitable for energy-constrained sensors.

Wang et al.[27] presents a 3D many-objective positioning model for wireless sensor location in the Internet of Things (IoT), considering distance error, node distribution, and computational cost. It proposes a data preprocessing and outlier removal strategy and uses a fashionable optimization algorithm. Experimental results show the model's accuracy and robustness surpass current single and multi-objective positioning models. Waqas et al.[28] developed an LSTM-DL model to optimize localization processes in WSN, minimize distance measurement errors, and have an improved evaluation function to reduce topological inaccuracies. The model outperforms traditional DV-Hop methods in accuracy and reliability, indicating that advanced LSTM-DL holds significant promise over conventional approaches in node localization within WSNs. Suphian et al.[29] designed a PSO-GRNN model to improve sensor node location and target tracking accuracy. This method uses RSSI values as initiation data for the GRNN algorithm, which uses the PSO method to determine the optimal GRNN spread constant value. The PSO method overcomes the insecure trial-and-error method of selecting the optimal GRNN spread constant, improving localization and target tracking accuracy. The hybrid PSO-GRNN outperforms the conventional LNSM technique and can achieve a significant gain of 87.58% compared to conventional RSSI. Chen et al.[12] enhanced the accuracy of DV-Hop by considering the quality of the links between nodes, refining the average hop count and unknown nodes, and modifying it with an error factor. This algorithm adjusts the hop distance calculation for each node based on signal strength by assigning weights to the links. Kim et al.[30] used an MLP with SGD among ANNs and RSSI data from a Zigbee sensor to estimate the indoor location. Four fixed nodes were placed at the corners of an unobstructed area, and mobile nodes captured position data and RSSI values. A data augmentation approach enhanced precision, and B-spline surface equations improved accuracy and computational speed. The method showed a decrease in the error margin below the sensor hardware tolerance as the segmentation steps increased. The research aims to develop practical indoor location-based services with minimal data. Han et al.[31] aimed to quantify the impact of anchor node mobility on the original non-ranging positioning algorithm in WSNs. They achieved better accuracy in various scenarios by combining the DV-Hop algorithm with other positioning techniques. Sharma et al.[32] proposed a method in which nodes collaborate to calculate their positions without relying on a central base station, reducing communication overhead and making the system more robust. Chen et al.[9] introduced an upgraded DV-Hop method using a spring model and reliable anchor node groups, abstracting the relationships between nodes in the network into a spring model to improve the localization capability of DV-Hop. Zhao et al.[33] developed an optimized algorithm that uses a range-based weighting factor to optimize the DV-Hop algorithm. This approach considers the transmission range of nodes and adjusts the hop distance accordingly to enhance the positioning accuracy of nodes. Kaur et al.[34] used two Grey Wolf algorithms to optimize the average hop distance, achieving some improvement in accuracy. Still, the hop count was not optimized, and the least squares method did not address minimizing weight errors. Wang et al.[35] suggested a distance estimation algorithm (DEM) to enhance the DV-Hop localization in WSNs. The DEM retrieves hop information, evaluates the maximum distance within an anchor node at distinct hops, averages the distance, and integrates the average distance with multiple objectives. Compared to other current approaches, the presented model achieves promising enhancements in terms of localization accuracy and required cost for measurement sensors of WSNs, where the model could be useful for enhancing the present sensor node localization techniques.

Methodology

The proposed study introduces the error correction mechanism of the LSTM-based DV-Hop localization algorithm. The algorithm's uneven error distribution can be used to correct errors, reduce localization errors, and improve accuracy[36]. The proposed methodology presents the robust LSTM-based DV-Hop localization algorithm, which includes an error correction mechanism, parameter determination, error function definition, and optimization methods.

WSN DV-Hop positioning algorithm

The DV-Hop algorithm for the location of nodes in WSN used for the monitoring, security, and health of the environment has been improved. However, the error location of the previous algorithms has not significantly influenced the errors detected. The proposed research revealed the strong idea that combining LSTM networks close to the DV-Hop algorithm would improve distance estimation, reducing localization errors. The method involves estimating the minimum hop count, the average hop distance, and the subsequent determination of the coordinates of the nodes whose locations are unknown to the system. The nodes were deployed within the WSN network, and the anchor nodes transmitted a packet containing position information, identification number, hop count, and address information to the neighboring nodes. The initial hop count is zero, and the hop count increases as the packet is transmitted. The nodes compare the packet's hop count to the corresponding hop count in their routing table, updating it if the received hop count is smaller. If the received hop count is larger, the packet is discarded, ensuring optimal information in the routing table. The minimum hop count from the unknown node to each anchor node is determined by repeatedly traversing all nodes in the WSN.

After determining the minimum hop count from the unknown nodes to the anchor nodes, the average hop distance $\mathrm{HopSize}_i$ required for node *i* to hop across the network is computed based on the coordinates and hop information of the anchor nodes. The average hop distance is computed using Eq. (1).

$$\mathrm{HopSize}_i = \frac{\sum_{i\neq j}\sqrt{(x_i-x_j)^2+(y_i-y_j)^2}}{\sum_{i\neq j} h_{ij}} \tag{1}$$

where *i* and *j* are the identifiers of two different anchor nodes, *m* represents the number of anchor nodes, (x_i, y_i) and (x_j, y_j) are the coordinates of anchor nodes *i* and *j*, respectively, and h_{ij} represents the hop count between anchor nodes *i* and *j*.

The average hop distance for anchor nodes is calculated and broadcasted within the WSN using the communication radius *R* to determine the coverage area. The unknown node *p* records the first calculated average hop distance and ignores subsequent received values to ensure the average hop distance is received from the nearest anchor node. The estimated distance between the unknown node *p* and anchor node *i* is calculated using Eq. (2).

$$d_{pi} = \mathrm{HopSize}_i \times h_{pi} \tag{2}$$

The distance between nodes *p* and anchor nodes *i* is estimated using d_{pi} and h_{pi}, respectively. Node *i* exchanges information with neighboring nodes to obtain the hop count h_{pi} from each node to the unknown node *p* and the average hop distance between the unknown node *p* and the neighboring nodes.

Once the estimated distances $d_1, d_2, \ldots, d_n$ between node *p* and the anchor nodes are obtained, the position of the unknown node can be calculated using the trilateration method[37]. Based on the measured distances between the unknown node and the surrounding anchor nodes, the node's position is determined using the geometric relationship between the unknown and anchor nodes as shown in Eq. (3).

$$\begin{aligned} &AX = B \\ &X = (A^T A)^{-1} A^T B \end{aligned} \tag{3}$$

The DV-Hop algorithm is effective for distributed node localization in WSNs, but uneven node distribution and significant network topology changes may impact its positioning accuracy.

Error analysis based DV-Hop algorithm

When applied to a real WSN, the DV-Hop localization algorithm often suffers from low positioning accuracy due to the relationship between distance vectors and hop counts. External factors can also introduce errors.

Inequal sensor node density during deployment can lead to significant discrepancies in the average hop distance calculated by the original algorithm. One hop is the hop count from the beacon nodes to the unknown node D. The average hop distance for beacon node A is calculated in Eq. (4).

$$\mathrm{HopSize}_a = \frac{d_1 + d_2}{2 + 2} \tag{4}$$

The distance from unknown node D to beacon nodes A and B is calculated using the formula $\mathrm{HopSize}_a$, where d_1 and d_2 represent the distances from anchor nodes A to B and C, respectively, and the hop count from unknown node D to beacon nodes A and B is one hop each is computed in Eq. (5).

$$\mathrm{HopSize}_a \times h_{ai} = \frac{1}{4} \times (d_1 + d_2) \times 1 \tag{5}$$

The hop count from anchor nodes A and B to unknown node D is h_{ai}, but the distance between unknown node D and beacon nodes A and B varies significantly. The DV-Hop algorithm computes distances using average hop distance and faces significant errors due to the distance difference between unknown and beacon nodes. These findings are based on computing the minimum hop count and the average single-hop distance between the two nodes. The DV-Hop algorithm employs a polyline distance to calculate the average hop distance, revealing a nonoptimal distance within the network. Besides that, the positioning accuracy is affected by cumulative errors from the least squares method, with communication possible within a certain range but not outside it.

LSTM architecture for positioning algorithm

The model consists of an input, hidden, and output layer. The input layer controls the inflow of new information, influencing updates to the cell state[38,39]. It comprises a six-dimensional array of coordinates predicted by the DV-Hop algorithm, the predicted coordinate distance from the network topology center, the quadrant of the indicated coordinates, the number of anchor nodes within the communication radius, and the average hop distance of the nodes. The hidden layers are the core of the LSTM structure, consisting of memory cells, input gates, forget gates, and output gates[40]. The LSTM consists of three layers, with 240 units in the first, 120 in the second, and 60 in the third. The last layer transforms the result from the hidden layers into the final prediction. The FC layer of the improved OLSTM-DVHop positioning algorithm yields the predicted position based on a specific task.

Proposed architecture of OLSTM-DVHop positioning algorithm

This study presented an improved WSN DV-Hop positioning algorithm based on LSTM networks (OLSTM-DVHop) to solve the DV-Hop algorithm's localization errors. The proposed OLSTM-DVHop is a new positioning method integrating deep learning and DV-Hop to improve localization accuracy and reduce noise interference. The general procedure of the algorithm, along with the specific functions, is enumerated below.

1. Determine the minimum hop count from each node to the anchor nodes. Each node broadcasts information to its surrounding nodes. Every node records the hop count to its neighboring nodes during this process. This data then identifies the minimum hop count from each node to the anchor nodes.
2. Calculate the average hop distance, $\mathrm{HopSize}$, epresents the mean distance a node traverses in a single hop across the network. It is calculated using anchor nodes' coordinates and hops information to ensure accurate distance estimation.
3. After obtaining the average hop distance of the beacon nodes, all anchor nodes broadcast the calculated average hop distances to the wireless sensor network. Unknown nodes record only the first received average hop distance and discard any subsequent ones. This strategy ensures that unknown nodes receive the average hop distance from the nearest beacon node, making the received average hop distance closer to the actual distance.
4. Based on the relationship between hop count and average hop distance between nodes, calculate the coordinates of the unknown nodes using trilateration.
5. The coordinate information calculated by the DV-Hop algorithm undergoes data preprocessing, which includes filtering and assessing the data to remove any invalid or abnormal values. The data is then normalized and processed for feature extraction. The input data for the LSTM model are divided into six dimensions: the position coordinates calculated by the DV-Hop algorithm, the distance between the predicted coordinates and the center of the network topology, the quadrant in which the predicted coordinates are located, the number of anchor nodes within the communication radius, and the average hop distance of the nodes. The input feature vectors include distance-related metrics (e.g., hop distances and distances from the network center) and location-related features (e.g., predicted position coordinates), while the outputs represent the predicted location coordinates. The LSTM network processes these inputs to output the predicted location coordinates of the unknown nodes in the WSN, leveraging its nonlinear predictive capability to enhance localization accuracy. This preprocessing ensures the quality and reliability of the data used for modeling.
6. Input the wireless sensor positioning data into the LSTM model for training. The input to the model consists of the six-dimensional feature values of the positioning data processed in Step 5. The output is the predicted coordinates of the unknown nodes. During the training phase, the true position coordinates of the nodes are compared with the model's output at each iteration to optimize the model's weight parameters.

7. The LSTM model is Optimized by performing a multi-layer grid search to fine-tune key parameters; a more accurate LSTM prediction model can be achieved, significantly improving the prediction performance for fault time series.

The OLSTM-DVHop positioning algorithm incorporates deep learning techniques with the traditional DV-Hop algorithm, offering a robust approach to enhance localization accuracy. The localization process begins with the DV-Hop algorithm estimating the positional coordinates of unknown nodes using hop counts and average hop distances. The data undergoes preprocessing to remove outliers and anomalies, followed by normalization and feature extraction to create six-dimensional feature vectors. These feature vectors, including distance and location metrics, are then processed by the LSTM network, which predicts the unknown nodes' location coordinates. The error correction mechanism further evaluates and iteratively adjusts the predictions, dynamically reducing errors through weighted entropy and adaptive threshold adjustments. This comprehensive approach ensures higher localization accuracy and robustness, validated through extensive simulation experiments. The proposed method's effectiveness is demonstrated through detailed experiments, as presented in section "Result".

In implementing the OLSTM-DVHop positioning algorithm, this study follows the principle of simplicity in designing wireless sensor localization algorithms. By integrating features of a multi-layer neuron autoencoder network, this paper constructs the overall framework of the LSTM prediction model, as shown in Fig. 1. The framework includes the input, hidden, and output layers, network training, and network prediction. The input layer preprocesses the DV-Hop localization data to meet the data input requirements of the deep learning network. The Hidden Layer uses LSTM cells to construct a single-layer recurrent neural network, while the Output Layer provides the prediction results. The model is optimized using the Adam algorithm.

OLSTM-DVHop algorithm training

In the model training phase, the original DV-Hop localization data are divided into a training set D_{tr} and a test set D_{te}. The training set $D_{\mathrm{tr}} = \{d_1, d_2, \ldots, d_m\}$ and the test set $D_{\mathrm{te}} = \{d_{m+1}, d_{m+2}, \ldots, d_n\}$ are standardized, resulting in new pre-training elements d'_i for the training set D'_{tr}, as shown in Eq. (6).

$$D'_{\mathrm{tr}} = \{d'_1, d'_2, \ldots, d'_m\} \tag{6}$$

The standardized data d'_i are computed using Eq. (7).

$$d'_i = \frac{d_i - \frac{1}{n}\sum_{i=1}^{n} d_i}{\sqrt{\frac{1}{n}\sum_{i=1}^{n}\left(d_i - \frac{1}{n}\sum_{i=1}^{n} d_i\right)^2}} \quad \text{for } 1 \leq i \leq m, t \in \mathbb{N} \tag{7}$$

To accommodate the input characteristics of deep learning models, the standardized training set D'_{tr} is segmented into length L segments, each representing a time step in the network input, as expressed in Eqs. (8) and (9).

$$X = \{X_1, X_2, \ldots, X_L\} \tag{8}$$

$$X_p = \{d'_p, d'_{p+1}, \ldots, d'_{m-L+p}\} \quad \text{for } 1 \leq p \leq L; p, L \in \mathbb{N} \tag{9}$$

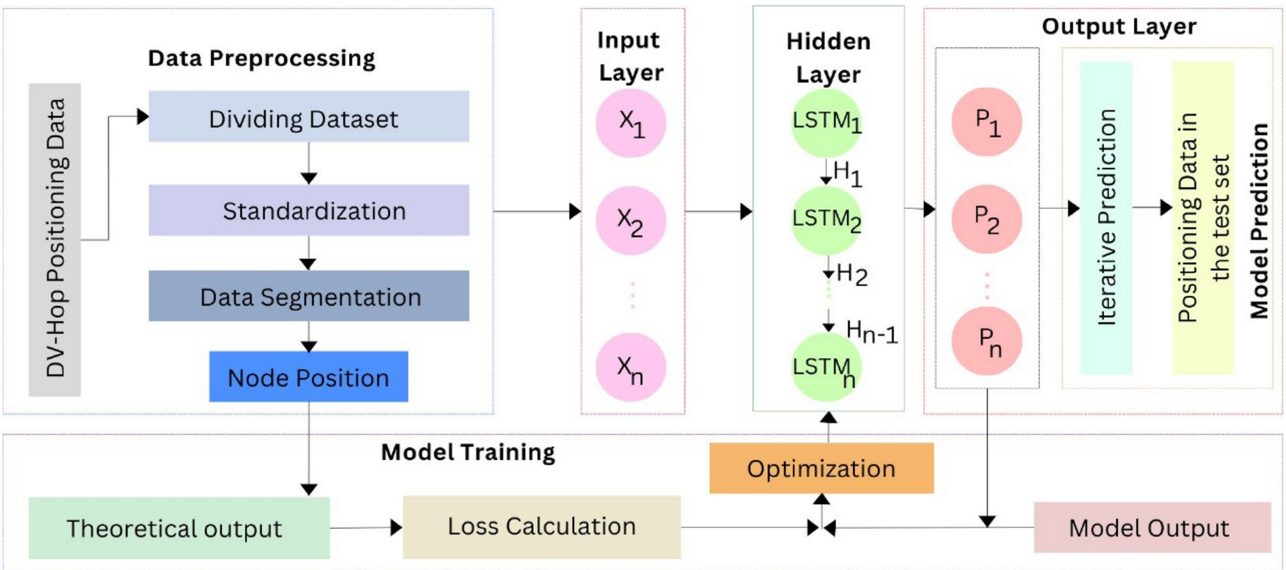

Fig. 1. Framework diagram of the OLSTM-DVHop localization algorithm.

The corresponding model output Y and Y_p is given by Eqs. (10) and (11).

$$Y = \{Y_1, Y_2, \ldots, Y_L\} \tag{10}$$

$$Y_p = \{r'_p, r'_{p+1}, \ldots, r'_{m-L+p}\} \tag{11}$$

The model processes the input X, and the output is represented by Eq. (12).

$$P = \{P_1, P_2, \ldots, P_L\} \tag{12}$$

The forward propagation operation is defined in Eq. (13).

$$P_p = \mathrm{LSTM}_{\mathrm{forward}}(X_p, C_{p-1}, H_{p-1}) \tag{13}$$

The mean square error (MSE) is used as the loss function to evaluate the model, defined in Eq. (14).

$$\mathrm{loss} = \frac{1}{L(m-L)} \sum_{i=1}^{L(m-L)} (p_i - y_i)^2 \tag{14}$$

Here, P_i represents the predicted position coordinates in the ith iteration, and Y_i denotes the true position coordinates for the training sample ith. Additionally, C_{p-1} is the cell state vector of the previous step $(P-1)$, and H_{p-1} is the hidden state vector of the previous step $(P-1)$. These variables are key to the forward propagation process in the LSTM model, ensuring accurate localization predictions.

Input: Training data D_{tr}, test data D_{te}, parameters L, S_{state}, learning rate η
Output: Trained LSTM model, prediction accuracy

```
Standardize the training set D_tr to obtain D'_tr
Divide D'_tr into segments X and corresponding labels Y
Initialize LSTM cells with state size S_state
Connect LSTM cells to form the LSTM network
Initialize the LSTM network with a random seed
for each training step from 1 to max steps do
    Perform forward propagation to obtain predictions P
    Compute the loss using Mean Squared Error (MSE)
    Update LSTM network parameters using the Adam optimizer
end
Return the trained LSTM model
```

Algorithm 1. OLSTM-DVHop model training algorithm.

Algorithms 1 is a DV-Hop localization algorithm based on deep learning LSTM. The input parameters include F_0, m, L, S_{state}, seed, steps, and n, and the output is the prediction sequence and model accuracy. The algorithm first extracts D_{te} and D_{tr} from the dataset D_0. After standardizing D_{tr}, the feature matrix X and label vector Y are obtained, where X is of size $L \times m$ and Y is of size $m - L$. An LSTM cell is created with a state size S_{state}, connected to X, and initialized using the seed. Forward propagation is performed to obtain the prediction vector P, and the loss is calculated. The Adam optimizer is used to update the LSTM network. The trained LSTM network is applied to the test set, generating prediction vectors $P_f + j$. These are stored in P_0 and reverse standardized to produce P_{te}, followed by error calculation.

OLSTM-DVHop predication

After the OLSTM-DVHop network model is trained, predictions are made using the trained model. The model's output data are given by Eq. (15), where Y_f is a sequence of predicted values.

$$Y_f = \{d'_{m-L+1}, d'_{m-L+2}, \ldots, d'_m\} \tag{15}$$

The sequence Y_f is fed into the deep learning network, producing the output P_f as shown in Eq. (16).

$$P_f = \mathrm{LSTM}^*_{\mathrm{net}}(Y_f) = \{p_{m-L+2}, p_{m-L+3}, \ldots, p_{m+1}\} \tag{16}$$

The sequence $Y_f + 1$, which includes the new data point d'_{m+1}, is expressed in Eq. (17).

$$Y_{f+1} = \{d'_{m-L+2}, d'_{m-L+3}, \ldots, d'_{m+1}\} \tag{17}$$

The output sequence of the model, P_0, is represented in Eq. (18).

$$P_0 = \{p_{m+1}, p_{m+2}, \ldots, p_n\} \tag{18}$$

The final sequence corresponding to the test set D_{te} is decoded and denormalized, as shown in Eq. (19).

$$P_{\mathrm{te}} = \mathrm{de\_zscore}(P_0) = \{P^*_{m+1}, P^*_{m+2}, \ldots, P^*_n\} \tag{19}$$

Input: Trained LSTM model, test data D_{te}
Output: Predicted position sequences
for *each sequence* Y_f *in* D_{te} **do**
  Feed Y_f into the LSTM network to get P_f
  Combine $Y_f + 1$ with new data point d'_{m+1}
  Continue prediction to form the final sequence P_0
  Decode and denormalize P_0 to get P_{te}
end
Return final predicted sequences P_{te}

Algorithm 2. OLSTM-DVHop model prediction algorithm.

OLSTM-DVHop optimization

This paper proposes the OLSTM-DVHop localization algorithm to optimize the parameters of the LSTM prediction model. An objective function based on the mean absolute error (MAE) measures the difference between predicted and actual values. The algorithm calculates the absolute error between the predictions and the actual values, averaging them to assess performance. The optimization problem is defined as shown in Eq. (20):

$$\min \epsilon(P_{\mathrm{te}}, D_{\mathrm{te}}) \quad \text{s.t.} \quad \begin{array}{l} 2 \le L \le L_{\max} \le m, \mathrm{step}_L \\ 2 \le S_{\mathrm{state}} \le S_{\max}, \mathrm{step}_S \\ \eta \in \{\eta_1, \eta_2, \ldots, \eta_r\}, \mathrm{step}_\eta \end{array} \tag{20}$$

In the model optimization phase, Algorithm 3 finds the optimal combination of L, S_{state}, and η. The search space is three-dimensional, with $L_{\max}$ and $S_{\max}$ constrained to a smaller range to control model complexity. The random seed and training steps are fixed, while the value ranges for L, S_{state}, and η are iterated. The LSTM model is trained during the inner loop, and the results are sorted by prediction accuracy to determine the best parameter combination.

At the start, the input data F_0, training size m, time step L, state size S_{state}, random seed, steps, and learning rate η are predefined. For each η, values are iterated with step size step_η, and similarly for L and S_{state}. The function $\mathrm{LSTMpredict}(F_0, m, L, S_{\mathrm{state}}, \mathrm{seed}, \mathrm{steps}, \eta)$ trains the LSTM model and returns the prediction results. The optimal parameter set is returned, ensuring enhanced model performance and minimizing overfitting.

Input: Training data D_{tr}, test data D_{te}, parameter ranges for $L, S_{\mathrm{state}}, \eta$
Output: Optimized model parameters
for *each* η *in* $\{\eta_1, \eta_2, \ldots, \eta_r\}$ **do**
  for *each* L *from 2 to* L_{max} **do**
    for *each* S_{state} *from 2 to* S_{max} **do**
      Train LSTM model on D_{tr}
      Predict on D_{te}
      Calculate error $\varepsilon(P_{\mathrm{te}}, D_{\mathrm{te}})$
      if *error is less than the previous minimum* **then**
        Update optimal parameters
      end
    end
  end
end
Return optimized parameters

Algorithm 3. OLSTM-DVHop model optimization algorithm.

Error correction mechanism based on OLSTM-DVHop algorithm

The study introduces a localization error correction mechanism to address localization errors in the DV-Hop algorithm due to uneven regional distribution. The mechanism consists of actions taken under certain states, with the strategy for determining error corrections involving various decision-making methods. The choice of error correction significantly impacts the cumulative rewards of the mechanism, and selecting an incorrect action may lead to failure or correction failure. The network topology environment represents the scene where nodes are located, including states, actions, rewards, and state transition rules. The mechanism aims to find an optimal policy that maximizes long-term rewards. Good state selection simplifies the problem, reduces decision-

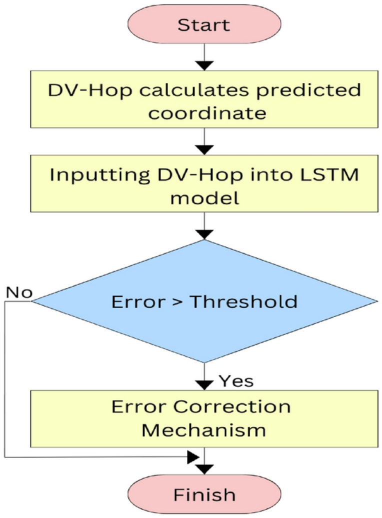

Fig. 2. Flowchart of the localization error correction mechanism based on OLSTM-DVHop.

making, and improves efficiency and accuracy. The policy choice is critical to the mechanism's performance and effectiveness in completing tasks.

The paper proposes a localization error correction mechanism based on the OLSTM-DVHop algorithm to enhance the accuracy of localization algorithms and reduce errors. The mechanism involves network initialization, where all nodes broadcast their IDs and record the IDs they sense, as depicted Fig. 2. Neighboring nodes exchange ID information packets and perform continuous processing of hop counts. The DV-Hop algorithm calculates the predicted coordinates of location nodes, with the minimum hop count from unknown nodes to anchor nodes determined. Anchor nodes compute the average hop distance of anchor nodes using the coordinates and hop count information of beacon nodes. Unknown nodes record only the first received average hop distance and discard any subsequent average hop distances. The coordinates calculated by the DV-Hop algorithm are preprocessed and input into the LSTM model for training. The error in the predicted coordinates is checked to determine if it exceeds a predefined threshold. The localization error correction mechanism is applied to refine the predicted coordinates, achieving higher localization accuracy. This comprehensive process ensures improved accuracy and reliability in diverse wireless sensor network deployments. The LSTM-based localization framework uses preprocessing and dynamic correction mechanisms to enhance model robustness and localization accuracy.

Algorithm 4 illustrating the steps for minimizing localization errors by adjusting the correction rate and updating state values iteratively.

Input: Initial state S_0, correction directions K, trials T, correction rate c
Output : Minimized localization error

for *each correction direction k from* 1 *to* K **do**
  for *each trial t from* 1 *to* T **do**
    Execute correction action
    Observe the resulting state and error
    Update state value function
    if *correction successful* **then**
      Adjust correction rate c
    end
  end
end
Return optimized state value function and minimized error

Algorithm 4. Error correction mechanism algorithm.

The DV-Hop algorithm has significant errors, which are addressed by proposing an EC mechanism to reduce these errors. The mechanism estimates state value functions and adjusts based on cumulative errors, starting with exploring the environment, estimating state value functions, and refining them through sampling, as depicted in Fig. 3.

The proposed error correction mechanism addresses localization errors in the DV-Hop algorithm by leveraging the uneven distribution of errors. The mechanism leverages the uneven distribution of errors to correct them, reducing errors and enhancing accuracy partially. The error function is a weighted average error function based on entropy used to compare the differences between the model output and the true target values-the error function in Eq. (21) aims to minimize the squared error between the target true values and the model

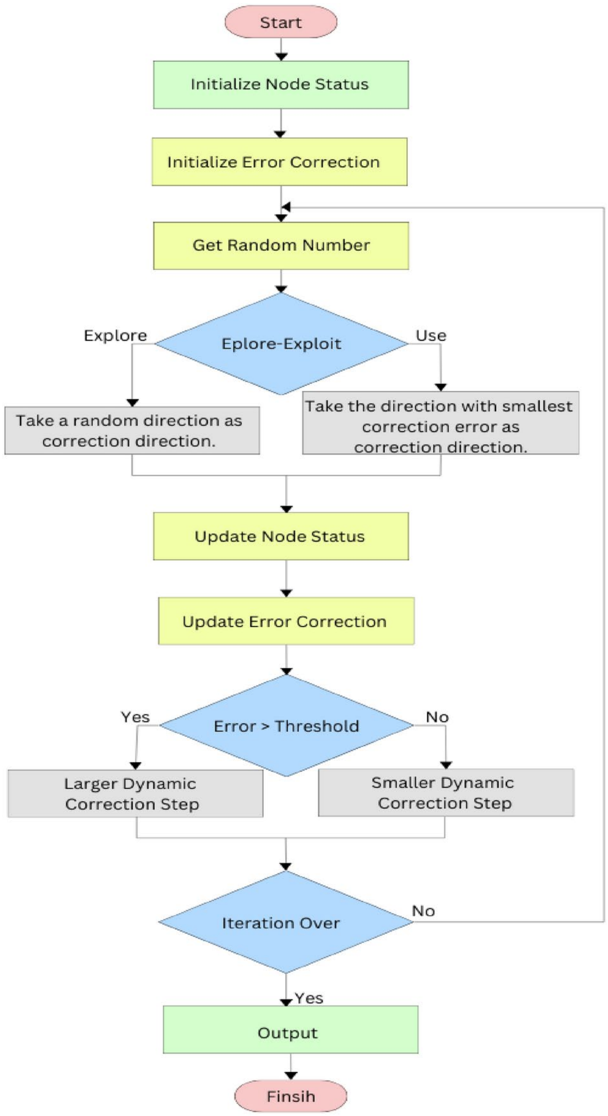

Fig. 3. Flowchart of the localization error correction mechanism.

output values, considering the uncertainty and diversity of probability distributions to improve the model's performance and generalization ability. The error correction mechanism is an iterative optimization mechanism designed to dynamically adjust parameters to minimize the error function and improve localization accuracy. The implementation error correction mechanism is depicted in Algorithm 5

Input: K Number of offset directions, R error function, T number of attempts, c correction rate, and λ dynamic correction threshold
Output : Generates the cumulative error r

Initialize: $\forall i = 1, 2, \ldots, K : Q(i) = 0, \text{count}(i) = 0, r = 0$
for $t = 1$ **to** T **do**
    if $rand(t) < c$ **then**
        $k =$ randomly select from $\{1, 2, \ldots, K\}$
    end
    else
        $k = \text{argmax}(Q(i))$
    end
    $v = R(k)$
    $r = r + v$
    $Q(k) = \frac{Q(k) \times \text{count}(k) + v}{\text{count}(k) + 1}$
    $\text{count}(k) = \text{count}(k) + 1$
end
Calculate the dynamic correction threshold λ
Initialize $c = 1, \lambda = \frac{1}{2}c$
for $t = 1$ **to** T **do**
    if $\lambda < c$ **then**
        $c = \frac{1}{\sqrt{t}}$
    end
    else
        $c = \frac{\lambda}{2(1+\lambda^3) - t}$
    end
    if $c == 0$ **then**
        $c = \frac{\lambda}{2}$
        $\lambda = c$
    end
end

Algorithm 5. Error correction algorithm.

$$R(k) = \frac{e^{Q(k)/c}(Y^2 - Y_p^2)}{\sum_{i=1}^{K} e^{Q(i)/c}} \tag{21}$$

K is number of offset direction, R is the error function, T is the number of attempts, c is the correction rate, λ is the dynamic correction threshold and r is the output cumulated error. Cumulative error r is introduced and set to zero initially, $Q(i)$ and $\text{count}(i)$ for the offset direction i are set to zero as well. In each iteration, if the random number generated is smaller than the correction rate c, a random offset k is selected. Otherwise, the maximum $Q(i)$ offset direction is selected. The partial error v is obtained using error function R in the selected direction k and accumulated to the total error r. This process continues to update the model, which includes a dynamic control of some of the parameters to reach the control and stability of the designed algorithm. The determination of the dynamic characteristic of the correction threshold λ is made for the next computations cycle, and the correction rate c is also reflected in these changes. The approach minimizes the error function and improves the model's make through parameter optimization.

Evaluation mechanism
The localization error correction mechanism is a crucial concept that aims to address specific goals and challenges through interactions with its environment. Rewards, as feedback signals, are closely tied to correction

and guide the mechanism's actions. The primary objective is to maximize cumulative rewards and learn optimal decisions through iterative refinement. The design of reward signals directly influences the behavior learned by the mechanism, encouraging it to adopt strategies that align with the correction goals and desired behaviors. An effective mechanism must balance exploration and exploitation, known as the Exploration-Exploitation Balance. The $\varepsilon - greedy$ strategy is a common method to achieve this balance, allowing the mechanism to refine strategies efficiently, avoid stagnation in suboptimal solutions, and improve localization accuracy by leveraging the strengths of both approaches.

The WSN topology environment provides a crucial feedback signal to the error correction mechanism, evaluating the quality of its actions and encouraging or penalizing specific behaviors. **Rewards** are the primary learning signal for the mechanism, guiding its actions in different states to maximize rewards. The goal is to maximize cumulative rewards, the total rewards obtained over all time steps. The mechanism uses value functions to evaluate future rewards for given states and actions, helping it make more informed decisions. Rewards are closely linked to corrections and serve as guiding signals for the localization error correction mechanism. The design of reward signals is critical to the success of corrections, as poor signals can lead to incorrect strategies and correction failures. A well-designed reward system ensures the mechanism learns effective strategies for successful error correction and improved localization accuracy, expressed as in Eq. (22).

$$E\left[\frac{1}{T}\sum_{t=1}^{T} r_t\right] \tag{22}$$

The EC model of the TE is a decision-making process within the environment *E*, consisting of state space *X*, action space *P*, reward function *R*, and state transition probability function. The goal is to find an optimal policy that maximizes long-term rewards. State selection simplifies the problem, reduces decision counts, and improves efficiency and accuracy. The policy defines the EC mechanism's action choices in various states, forming *P* where each policy maps the current state to the chosen action. The EC mechanism aims to interact with the TE to maximize future rewards, and policy choice is crucial to the mechanism's performance. Rewards are feedback signals from the TE that guide the actions of the EC mechanism. The reward *Q*(*k*) is the average reward for correction action *a*, calculated as Eq. (23).

$$Q(k) = \frac{1}{n}\sum_{i=1}^{n} v_i \tag{23}$$

Initial rewards are set to zero, and with each trial, the reward is updated as Eq. (24).

$$Q_n(k) = \frac{1}{n}((n-1) \times Q_{n-1}(k) + v_n) \tag{24}$$

Besides this, the **error value function** is a mathematical tool that evaluates the quality of actions an error correction mechanism takes in a given state. It helps the mechanism decide which action to take to maximize rewards. The function represents the long-term average reward obtained by the mechanism when following the current policy from the given state, expressed as the discounted sum of expected future rewards. This function is crucial for making informed decisions to optimize error correction performance.

Result

Simulation experiments and analysis

The research analyzes terrain limitations, which impact the positioning accuracy of WSNs, and evaluates the efficacy of error correction algorithms in node positioning. Multiple WS nodes are deployed within a $100m \times 100m$ simulation area by developing four-node distribution scenarios. The study compares eight positioning algorithms' robustness and positioning accuracy under various parameters, aiming to propose a suitable method for WSNs with terrain limitations. The experiment was conducted with a total number of wireless sensor nodes ranging from 100 to 600, increasing by 50 each time. The anchor node ratio ranged from 10% to 35%, with increments of 2.5% each time. The experimental results were obtained from 60 trials. The LE is used as an evaluation criterion to assess the performance of eight different algorithms for positioning unknown nodes in wireless sensor networks. The evaluated algorithms are DV-Hop, DEDV-Hop[41], WOADV-Hop[42], ABCDV-Hop[43], 3D DEH DV-Hop[44], RNN-LSTM[45], VCR IDV-Hop[46] and the OLSTM-DVHop positioning algorithm proposed in this study.

Figure 4 depicts the DV-Hop algorithm's positioning error when positioning an unknown node. The circle represents the true coordinates of the unknown node, while the rectangle represents the predicted position. The rectangular block's distance from the true coordinates indicates the algorithm's error. The error size is related to the unknown node's position and direction in the network topology. The error is smaller at the center and larger at the edge. The star represents the node coordinates after position correction, which are very close to the true coordinates. This indicates that the positioning error correction mechanism effectively corrects the unknown node's positioning error. As depicted in Fig. 4, the corrected positions align very closely with the true positions, indicating the effectiveness of the OLSTM-DVHop algorithm in minimizing localization errors. This alignment results from the controlled conditions of the simulation, where environmental noise, irregular node distributions, and signal interference are absent. These controlled conditions were chosen to evaluate the fundamental capabilities of the proposed model in an idealized environment. Although these results demonstrate

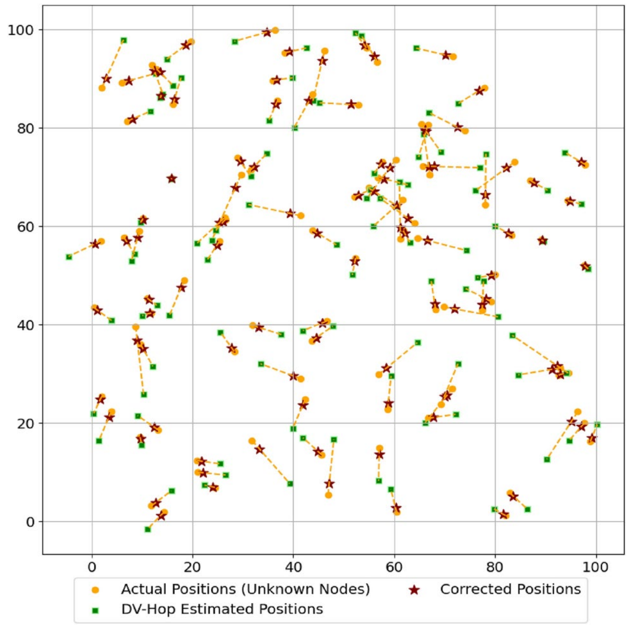

Fig. 4. Comparison chart of positioning error correction mechanism and DV-Hop positioning.

Parameter	Value
WSN dimension	100 m × 100 m
Communication radius	25–60 m
Anchor ratio	5–35%
Number of nodes	100–600
Offset directions	180
Number of attempts	120
Correction rate	0.3
Max epochs	100
Gradient threshold	1
Initial learning rate	0.001
Hidden units	60–210
LSTM layers	3

Table 1. Environment configuration and optimized model parameters.

the potential of the OLSTM-DVHop algorithm under idealized conditions, we recognize the need to validate its performance under realistic scenarios.

The configuration information of the simulation environment is shown in Table 1. The parameters include Max Epochs, Gradient Threshold, Initial Learn Rate, Learn Rate Drop Period, Learn Rate Drop Factor, Hidden Units, and LSTM Layer, each contributing to the model's performance. Localization accuracy is evaluated using the Average Localization Error (ALE), calculated as the Euclidean distance between predicted and actual positions, as shown in Eq. (25).

$$\mathrm{ALE} = \frac{\sum_{i=1}^{n} \sqrt{(x_i - x_i')^2 + (y_i - y_i')^2}}{n \times R} \tag{25}$$

where *n* represents nodes, x_i and y_i are actual coordinates, x_i' and y_i' are predicted coordinates, and *R* is the communication radius. A smaller ALE indicates higher localization accuracy. The effect of terrain constraints on localization accuracy is evaluated using four network typologies: rectangular random distribution, C-shaped, S-shaped, and O-shaped, with sensor nodes randomly deployed in the same $100m \times 100m$ area.

Performance analysis of localization accuracy

This study used the ALE (Average Localization Error) metric to measure the localization accuracy of the proposed algorithm in relation to the total number of nodes. The performance of the proposed algorithm is compared to existing algorithms, including DV-Hop[41], WOADV-Hop[42], ABCDV-Hop[43], 3D DEH DV-Hop[44], VCR IDV-

Hop[46], RNN-LSTM[45]. The simulation experiments were conducted in a $100m \times 100m$ rectangular simulation area, with the beacon node ratio set at 25% and the communication radius fixed at $35m$. To examine the impact of the total number of nodes, the node count was gradually increased from 100 to 600 in increments of 50, with 50 experiments conducted at each increment to obtain average results. The ability of the eight algorithms to localize unknown nodes was evaluated through the average localization error. Figure 5a–d illustrates the effect of varying the number of nodes on ALE in four different network topologies.

In the rectangular random topology, as shown in Fig. 5a, compared to the traditional DV-Hop algorithm, the DEDV-Hop algorithm reduced ALE by 30.77%. Similarly, WOADV-Hop reduced ALE by 32.31%, 3D DEH DV-Hop by 35.38%, ABCDV-Hop by 41.54%, VCR IDV-Hop by 46.15%, and RNN-LSTM by 52.31%. Additionally, the proposed OLSTM-DVHop algorithm outperformed all methods, reducing ALE by 58.46% compared to the traditional DV-Hop algorithm.

Figure 5b depicts the C-shaped random topology. Compared to DV-Hop, DEDV-Hop reduced ALE by 13.33%, WOADV-Hop achieved a 24% reduction, ABCDV-Hop achieved a 29.33% reduction, 3D DEH DV-Hop achieved a 33.33% reduction, VCR IDV-Hop achieved a 40% reduction, RNN-LSTM achieved a 46.67% reduction, and OLSTM-DVHop reduced ALE by 54.67%. OLSTM-DVHop outperformed the other algorithms across all four network topologies, resulting in a significant reduction in ALE. Specifically, OLSTM-DVHop performs better than DV-Hop, DEDV-Hop, WOADV-Hop, ABCDV-Hop, 3D DEH DV-Hop, VCR IDV-Hop, and RNN-LSTM by 54.67%, 47.69%, 40.35%, 35.85%, 32.00%, 24.44%, and 15.00%, respectively.

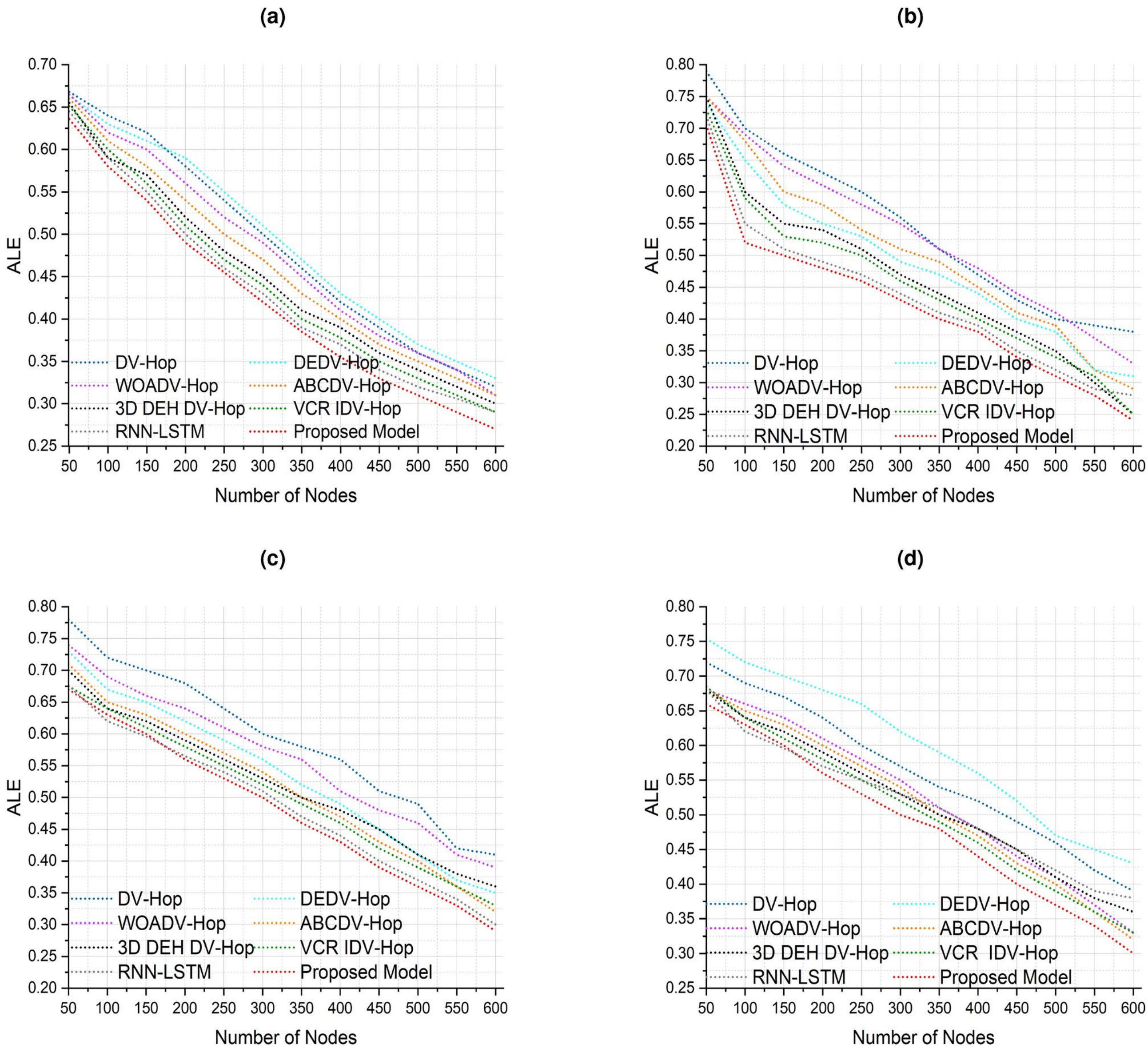

Fig. 5. (**a**) Positioning errors and unlocatable nodes in a square random topology. (**b**) Positioning error trends and anomaly distribution in a C-shaped grid topology. (**c**) Positioning errors and unlocatable nodes in an S-shaped random topology. (**d**) Positioning errors and unlocatable nodes in an O-shaped grid topology.

Figure 5c depict the S-shaped random topology, where DEDV-Hop reduced ALE by 28.84% compared to DV-Hop, WOADV-Hop achieved a 33.13% reduction, ABCDV-Hop achieved a 36.12% reduction, 3D DEH DV-Hop achieved a 40.11% reduction, VCR IDV-Hop achieved a 41.21% reduction, RNN-LSTM achieved a 42.91% reduction, and O-LSTM-DV-Hop reduced ALE by 60.41%. Specifically, OLSTM-DVHop performs better than DV-Hop, DEDV-Hop, WOADV-Hop, ABCDV-Hop, 3D DEH DV-Hop, VCR IDV-Hop, and RNN-LSTM by 54.67%, 47.69%, 40.35%, 35.85%, 32.00%, 24.44%, and 15.00%, respectively. Figure 5c depicts the S-shaped random topology, where DEDV-Hop reduced ALE by 28.57% compared to DV-Hop, WOADV-Hop achieved a 32.86% reduction, ABCDV-Hop achieved a 35.71% reduction, 3D DEH DV-Hop achieved a 40% reduction, VCR IDV-Hop achieved a 41.43% reduction, RNN-LSTM achieved a 42.86% reduction, and OLSTM-DVHop reduced ALE by 60%. OLSTM-DVHop performs better than DV-Hop, DEDV-Hop, WOADV-Hop, ABCDV-Hop, 3D DEH DV-Hop, VCR IDV-Hop, and RNN-LSTM by 52.78%, 45.16%, 41.38%, 38.18%, 34.62%, 32.00%, and 29.17%, respectively.

Figure 5d depict the O-shaped random topology, DEDV-Hop reduced ALE by 23.87% compared to DV-Hop, WOADV-Hop achieved a 30.16% reduction, ABCDV-Hop achieved a 35.29% reduction, 3D DEH DV-Hop achieved a 41.66% reduction, VCR IDV-Hop achieved a 43.53% reduction, RNN-LSTM achieved a 45.62% reduction, and O-LSTM-DV-Hop reduced ALE by 65.41%. OLSTM- DVHop consistently outperformed other algorithms across all network topologies, leading to significant reductions in ALE. The average localization accuracy of the different localization algorithms across various network topologies, with the total number of nodes used as the experimental variable, was analyzed. In the rectangular random topology, the O-LSTM-DV-Hop algorithm reduced ALE by 59.38% compared to the traditional DV-Hop algorithm. In the C-shaped random topology, ALE was reduced by 54.34%. In the S-shaped random topology, ALE was reduced by 60.41%, and in the O-shaped random topology, ALE decreased by 65.41%. O-LSTM-DV-Hop also showed various degrees of improvement over the three optimized DV-Hop algorithms in each wireless sensor network topology. In the O-shaped random topology, compared to DV-Hop, DEDV-Hop reduced ALE by 23.87%, WOADV-Hop achieved a 30.16% reduction, ABCDV-Hop reduced ALE by 35.29%, and O-LSTM-DV-Hop achieved a 65.41% reduction in ALE. O-LSTM-DV-Hop outperformed the other algorithms across all four network topologies, resulting in a significant decrease in ALE.

Performance analysis based on anchor node proportion in localization accuracy

This research evaluates the performance of eight different algorithms, including DV-Hop, WOADV-Hop, ABCDV-Hop, 3D DEH DV-Hop, VCR IDV-Hop and RNN-LSTM for localizing unknown nodes in wireless sensor networks, using ALE as the performance metric. The simulation experiments were conducted in a rectangular area of 100m × 100m, with the total number of wireless sensor nodes fixed at 600 and the communication radius set to 35 meters. The anchor node proportion was varied from 10% to 35%, with an incremental increase of 2.5%. The results were averaged over 50 trials to assess the accuracy of each algorithm in localizing unknown nodes.

Figure 6a–d presents the ALE for different anchor node proportions across various network topologies. The average ALE of the proposed OLSTM- DVHop is compared state-of-the-art (SOTA), using anchor node proportion as the variable. The comparison highlights the localization performance of each algorithm in four network topologies, demonstrating the proposed algorithm's superior performance and practical potential across different environments.

In the random rectangular topology Fig. 6a, DEDV-Hop reduces the average localization error (ALE) by 26.47% compared to DV-Hop, WOADV-Hop reduces the error by 44.12%, ABCDV-Hop by 50%, 3D DEH DV-Hop reduces the error by 33.82%, VCR IDV-Hop reduces the error by 39.71%, RNN-LSTM reduces the error by 47.06%, and the proposed OLSTM-DVHop achieves the highest reduction of 70.59%. Specifically, OLSTM-DVHop performs better than DV-Hop, DEDV-Hop, WOADV-Hop, ABCDV-Hop, 3D DEH DV-Hop, VCR IDV-Hop, and RNN-LSTM by 70.59%, 60%, 47.06%, 41.18%, 55.56%, 51.22%, and 44.44%, respectively. Figure 6b illustrates the findings from the C-shaped random topology. The reduction in the ALE achieved by 24%, WOADV-Hop by 40%, ABCDV-Hop by 52%, 3D DEH DV-Hop by 33.33%, VCR IDV-Hop by 42.67%, RNN-LSTM by 45.33%, and the proposed OLSTM-DVHop achieves the highest reduction of 80%, outperforming all other models. Specifically, OLSTM-DVHop performs better than DV-Hop, DEDV-Hop, WOADV-Hop, ABCDV-Hop, 3D DEH DV-Hop, VCR IDV-Hop, and RNN-LSTM by 80%, 73.68%, 66.67%, 58.33%, 70%, 65.12%, and 63.41%, respectively.

In the S-shaped random topology Fig. 6c, DEDV-Hop reduces ALE by 22.37%, WOADV-Hop by 31.58%, ABCDV-Hop by 42.11%, 3D DEH DV-Hop by 34.21%, VCR IDV-Hop by 40.79%, RNN-LSTM by 43.42%, and the proposed OLSTM-DVHop achieves a reduction of 59.21%, outperforming all other models. Specifically, OLSTM-DVHop performs better than DV-Hop, DEDV-Hop, WOADV-Hop, ABCDV-Hop, 3D DEH DV-Hop, VCR IDV-Hop, and RNN-LSTM by 59.21%, 48.14%, 40.38%, 31.03%, 38.33%, 36.27%, and 27.91%, respectively. Lastly, in the O-shaped random topology Fig. 6d, DEDV-Hop achieves an ALE reduction of 18.42%, WOADV-Hop by 40.79%, ABCDV-Hop by 44.74%, 3D DEH DV-Hop by 34.21%, VCR IDV-Hop by 39.47%, RNN-LSTM by 42.11%, and the proposed OLSTM-DVHop achieves a reduction of 59.21%, outperforming all other models. Specifically, OLSTM-DVHop performs better than DV-Hop, DEDV-Hop, WOADV-Hop, ABCDV-Hop, 3D DEH DV-Hop, VCR IDV-Hop, and RNN-LSTM by 59.21%, 48.39%, 40.89%, 31.03%, 38.33%, 36.27%, and 27.91%, respectively.

The study demonstrates the effectiveness of a proposed algorithm in accurately localizing WSNs using anchor node proportion as the experimental variable. The results show that when the proportion of anchor nodes is small, all algorithms exhibit large errors. However, as the communication radius increases, the localization accuracy of each algorithm improves. The proposed algorithm outperformed the other algorithms in terms of reducing ALE. Figure 6a–d shows the superior performance of the different localization algorithms across four

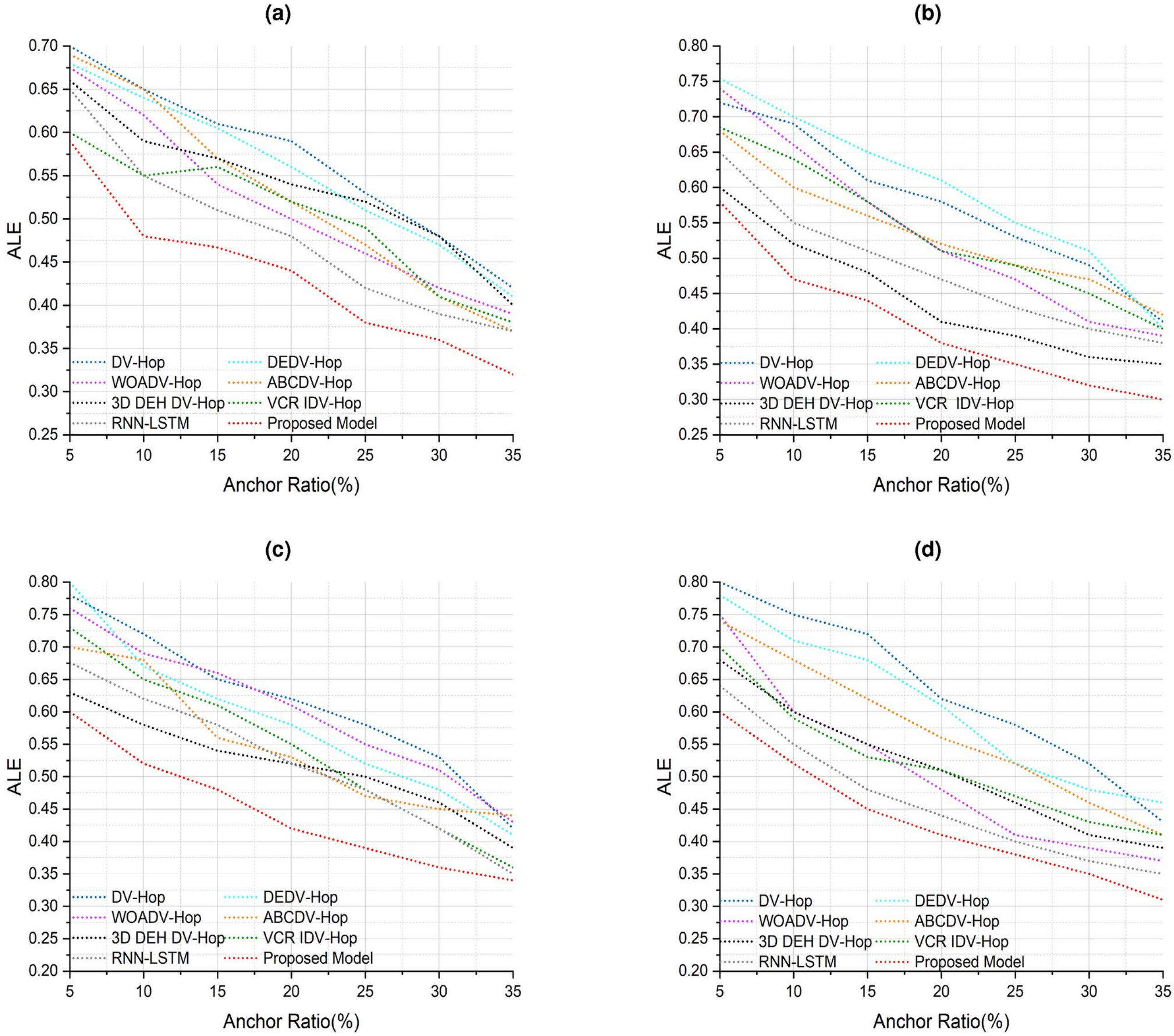

Fig. 6. Evaluation of ALE with different anchor node proportions across four network topologies. (**a**) Rectangular random distribution scenario, (**b**) C-shaped topology scenario, (**c**) S-shaped topology scenario, (**d**) O-shaped topology scenario.

different network topologies. The OLSTM-DVHop method significantly reduced ALE compared to the other algorithms in all typologies.

Performance analysis based on radius on localization accuracy

This paper presents simulation experiments using ALE to evaluate the performance of the LSTM-based DV-Hop positioning algorithm. It compares it with existing algorithms using communication radius as a variable, including DV-Hop, WOADV-Hop, ABCDV-Hop, 3D DEH DV-Hop, VCR IDV-Hop, and RNN-LSTM. To examine the impact of communication radius on localization accuracy, the communication radius was gradually increased from 10m to 60m in increments of 5m, and each experiment was repeated 60 times to obtain the results. The accuracy of each algorithm in localizing unknown nodes was compared, and Fig. 7a–d presents the effect of varying communication radii on ALE across four different network typologies.

In the random rectangular topology Fig. 7a, obstacles between unknown nodes and anchor nodes weaken the communication signal, increasing localization error. However, as the number of anchor nodes and total nodes increases, the localization error for all algorithms decreases, primarily due to the increased communication radius, which allows unknown nodes to communicate with more anchor nodes, expanding the range of information dissemination. In the random rectangular topology, the DEDV-Hop algorithm achieves a performance improvement of 6.89% compared to DV-Hop, WOADV-Hop reduces ALE by 34.01%, and ABCDV-Hop reduces ALE by 41.27%. The 3D DEH DV-Hop reduces ALE by 50.32%, VCR IDV-Hop reduces ALE by 47.19%, RNN-LSTM reduces ALE by 47.93%, and the proposed OLSTM-DVHop achieves the highest ALE

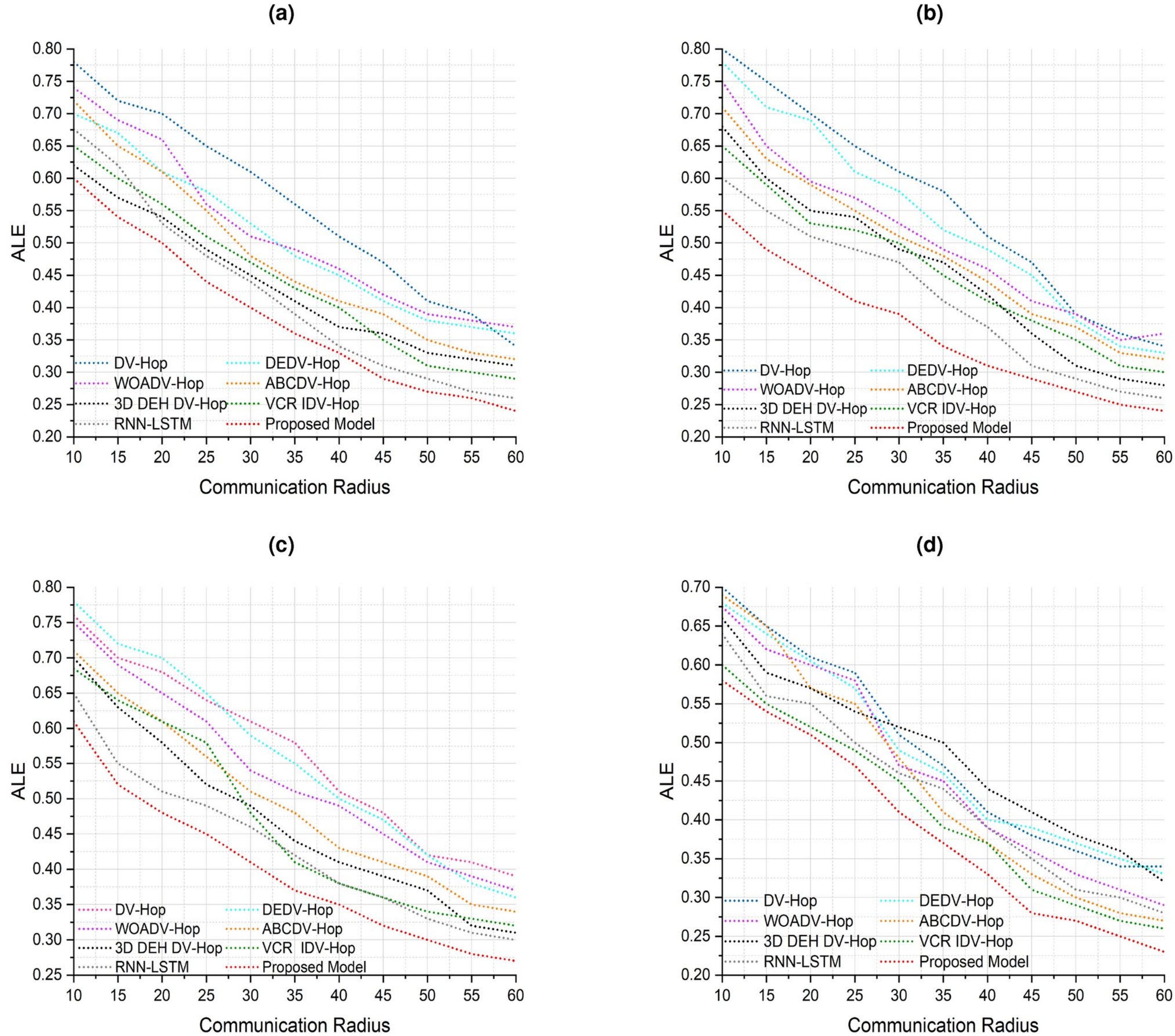

Fig. 7. Evaluation of the impact of different communication radii on ALE in four different network topologies. (**a**) ALE in rectangular random distribution scenario. (**b**) C-shaped topology scenario. (**c**) S-shaped topology scenario. (**d**) O-shaped topology scenario.

reduction of 70.25%. Specifically, OLSTM-DVHop performs better than DV-Hop, DEDV-Hop, WOADV-Hop, ABCDV-Hop, 3D DEH DV-Hop, VCR IDV-Hop, and RNN-LSTM by 62.82%, 52.94%, 45.50%, 51.85%, 48.91%, 49.00%, and 49.26%, respectively. In the C-shaped network topology Fig. 7b, obstacles between unknown nodes and anchor nodes weaken the communication signal, leading to an increase in localization error. However, as the number of anchor nodes and total nodes increases, the localization error for all four algorithms decreases, primarily due to the increased communication radius, which allows unknown nodes to communicate with more anchor nodes, expanding the range of information dissemination. In the C-shaped network topology, the DEDV-Hop algorithm achieves a performance improvement of 7.03% compared to DV-Hop, WOADV-Hop reduces ALE by 34.64%, and ABCDV-Hop reduces ALE by 40.37%. The 3D DEH DV-Hop reduces ALE by 50.60%, VCR IDV-Hop reduces ALE by 47.05%, RNN-LSTM reduces ALE by 47.80%, and the proposed OLSTM-DVHop achieves the highest ALE reduction of 70.25%. Specifically, OLSTM-DVHop performs better than DV-Hop, DEDV-Hop, WOADV-Hop, ABCDV-Hop, 3D DEH DV-Hop, VCR IDV-Hop, and RNN-LSTM by 62.82%, 52.91%, 45.61%, 51.82%, 48.92%, 49.02%, and 49.20%, respectively.

In the S-shaped network topology, the net localization accuracy is poor, and the reduction in localization error with increasing communication radius is slow. This is due to numerous winding paths, significantly improving the hop count between unknown and the farthest anchor nodes. This causes a significant deviation between the estimated and actual distances, ultimately leading to poor overall localization accuracy. In the S-shaped network topology Fig. 7c, the DEDV-Hop algorithm achieves a performance improvement of 7.72% compared to DV-Hop, WOADV-Hop reduces ALE by 33.54%, and ABCDV-Hop reduces ALE by 39.04%. The

3D DEH DV-Hop reduces ALE by 50.89%, VCR IDV-Hop reduces ALE by 47.21%, RNN-LSTM reduces ALE by 48.15%, and the proposed OLSTM-DVHop achieves the highest ALE reduction of 70.25%. Specifically, OLSTM-DVHop performs better than DV-Hop, DEDV-Hop, WOADV-Hop, ABCDV-Hop, 3D DEH DV-Hop, VCR IDV-Hop, and RNN-LSTM by 62.82%, 52.53%, 45.28%, 51.21%, 47.93%, 48.54%, and 49.13%, respectively. In the O-shaped network topology, with fewer winding paths, the estimation error is more minor, resulting in lower localization error. As the number of nodes increases and the distance between nodes decreases, the localization error becomes less pronounced. When the communication radius reaches 30m, the helpful information obtained by unknown nodes saturates, and further increases in communication radius result in diminishing returns. In the O-shaped network topology Fig. 7d, the DEDV-Hop algorithm achieves a performance improvement of 6.78% compared to DV-Hop, WOADV-Hop reduces ALE by 34.52%, and ABCDV-Hop reduces ALE by 39.14%. The 3D DEH DV-Hop reduces ALE by 49.95%, VCR IDV-Hop reduces ALE by 46.05%, RNN-LSTM reduces ALE by 47.18%, and the proposed OLSTM-DVHop achieves the highest ALE reduction of 70.25%. Specifically, OLSTM-DVHop performs better than DV-Hop, DEDV-Hop, WOADV-Hop, ABCDV-Hop, 3D DEH DV-Hop, VCR IDV-Hop, and RNN-LSTM by 62.96%, 53.07%, 47.33%, 51.11%, 48.64%, and 49.27%, respectively.

The experimental results show that when the communication radius is constrained, each algorithm exhibits relatively high localization errors, and the proportion of beacon nodes significantly affects the accuracy of the localization algorithms. However, as the communication range expands, localization accuracy becomes more stable. These observations underscore the importance of considering communication radius and beacon node proportion when developing and evaluating localization algorithms. The results indicate that increasing the communication range improves the accuracy of position prediction for wireless sensor nodes.

Positioning error trend and anomaly distribution

In this simulation experiment, the anchor node ratio is set to 35%, and the number of nodes is 600. The communication radius is 65 meters. LE ≥ 0.7 is considered an isolated node, regarded as an unlocatable node. The boxplot displays every unknown node's minimum and maximum positioning errors in the four network topology algorithms, the median error, and the 25th and 75th percentiles of LE results. The proposed OLSTM-DVHop algorithm is relatively stable, with positioning errors lower than the original DV-Hop and others.

In a square random topology, as shown in Fig. 8a, the Proposed Model achieves the lowest ALE, with only four nodes having errors higher than average, demonstrating superior accuracy. In contrast, DV-Hop has the highest ALE, with 45 nodes exhibiting higher errors. WOADV-Hop shows improvement with 30 nodes, while ABCDV-Hop and DEDV-Hop have 27 and 35 nodes, respectively. The 3D DEH DV-Hop and VCR IDV-Hop algorithms perform moderately, with 33 and 31 nodes having higher errors. RNN-LSTM performs slightly better, with 28 nodes showing higher errors. In the C-shape grid topology, positioning errors vary significantly across the algorithms, as depicted in Fig. 8b. The DV-Hop algorithm exhibits the highest ALE, with 41 nodes having positioning errors above average. DEDV-Hop slightly improves, with 37 nodes showing errors. WOADV-Hop and ABCDV-Hop reduce the number of nodes with higher errors to 29 and 26, respectively. The 3D DEH DV-Hop and VCR IDV-Hop algorithms have 34 and 32 nodes with higher errors, showing moderate improvement. The Proposed OLSTM-DVHo model performs best, with only three unlocatable nodes and 36 with higher errors, demonstrating the highest accuracy and stability. RNN-LSTM also improves but doesn't match the Proposed Model.

In the S-shape topology, as depicted in Fig. 8c, the performance of each algorithm varies. The DV-Hop algorithm exhibits the highest ALE, with 41 nodes having errors better than the average. The DEDV-Hop algorithm follows, with 37 nodes having positioning errors above average. The WOADV-Hop algorithm sees 29 nodes with errors exceeding the average, while the ABCDV-Hop algorithm has 26 such nodes. The Proposed Model (OLSTM-DVHop) significantly outperforms the others, with only three nodes unlocatable and 36 nodes having errors better than the average. In the O-shape grid topology, as shown in Fig. 8d, the DEDV-Hop algorithm has three unlocatable nodes and 34 nodes with positioning errors better than the average. The WOADV-Hop algorithm shows 31 nodes with positioning errors above the average. The ABCDV-Hop algorithm has three unlocatable nodes and 26 nodes with higher errors. Four nodes are unlocatable in the 3D DEH DV-Hop algorithm, and 31 show positioning errors exceeding the average. The Proposed Model significantly outperforms the others, with fewer nodes having positioning errors. The proposed positioning algorithm with an error correction mechanism demonstrates superior stability and positioning accuracy compared to the other algorithms tested in various scenarios. The experimental results prove the proposed algorithm's effectiveness and robustness in different WSN positioning scenarios. Our simulation results demonstrate the robustness of the proposed OLSTM-DVHop algorithm under conditions that include irregular node distributions and obstacles, which partially simulate noise effects. The LSTM-based error correction mechanism inherently mitigates data uncertainties, addressing multi-hop inaccuracies and uneven node distributions. This is evident in the significantly improved ALE metrics across diverse network scenarios, showcasing the algorithm's resilience against noise-like variations.

Comparison based on complexity and computation time

Computation time is a crucial measure of an algorithm's performance and complexity, reflecting the relationship between its running time and the size of the problem it solves. It helps predict and estimate program execution time and space usage, optimizing program design and planning. Comparing algorithm complexity allows selecting the most optimal solution for a given problem. In many cases, multiple algorithms can be applied to solve the same problem, making it essential to analyze their complexities to determine the most suitable option. This experiment uses computation time as a criterion to assess the impact of node count on algorithmic performance, shown in Table 2.

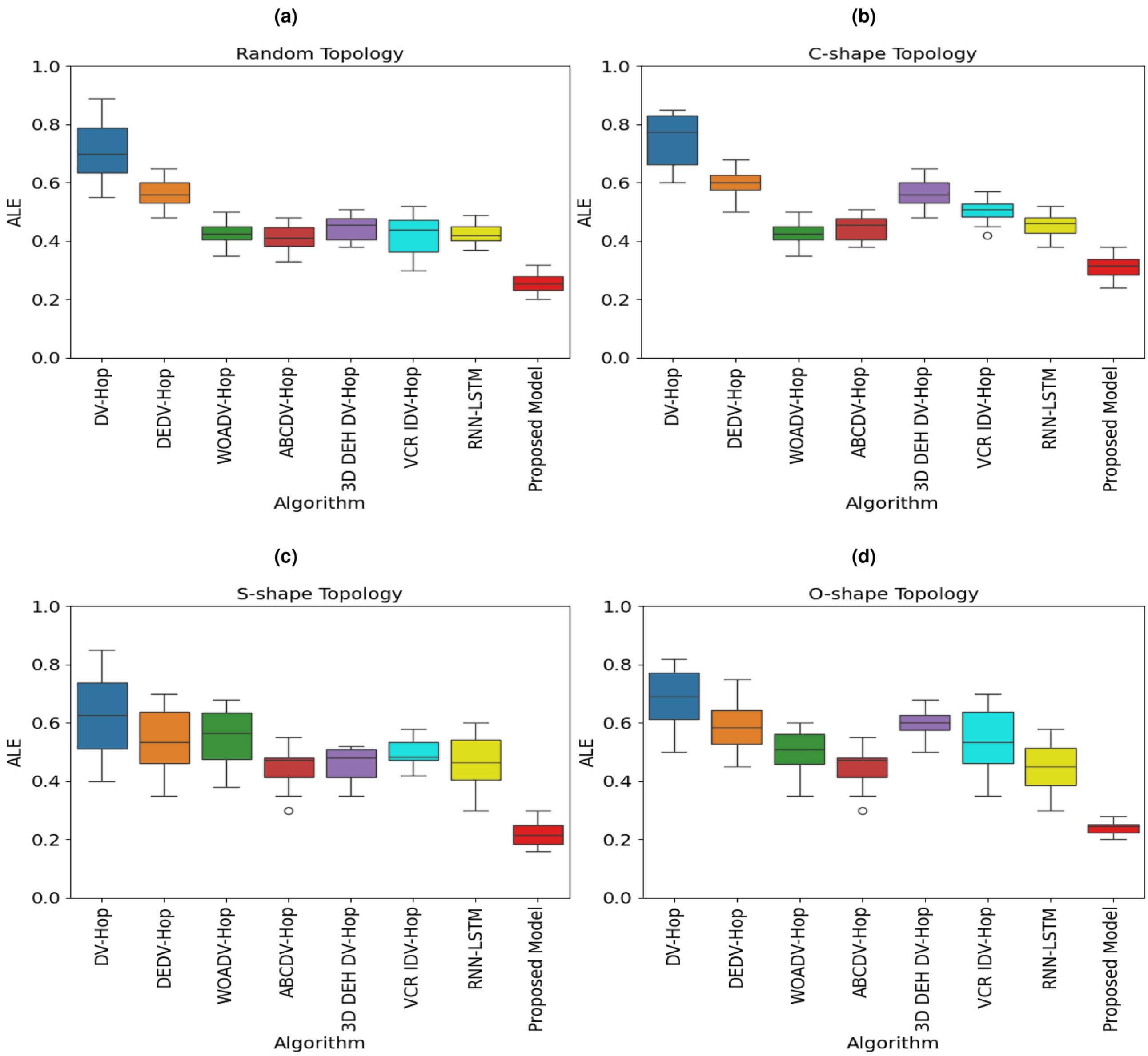

Fig. 8. Performance of the localization error correction mechanism in node localization. (**a**) Rectangular random distribution scenario. (**b**) C-shaped topology scenario. (**c**) S-shaped topology scenario. (**d**) O-shaped topology scenario.

Algorithm	Number of nodes					
	100	200	300	400	500	600
DV-Hop[14]	0.028	0.044	0.063	0.076	0.212	0.533
DEDV-Hop[41]	2.247	3.618	3.993	5.9421	7.962	4.67
WOADV-Hop[42]	0.056	0.079	0.099	0.099	0.242	0.324
ABCDV-Hop[43]	0.097	0.136	0.179	0.189	0.292	0.2934
3D DEH DV-Hop[44]	0.086	0.147	0.180	0.203	0.302	0.8345
VCR IDV-Hop[46]	0.057	0.126	0.148	0.177	0.182	0.1934
RNN-LSTM[45]	0.046	0.083	0.087	0.075	0.321	0.424
OLSTM-DVHop	0.030	0.046	0.065	0.067	0.115	0.092

Table 2. Computation time of positioning algorithms with different node counts (s).

The experimental results (Table 2) reveal that the computation time of OLSTM-DVHop algorithms varies with different node counts. Larger nodes increase the computation time, leading to more hops and distances, increasing the computational load on positioning algorithms. DV-Hop, despite large positioning errors, has the shortest computation time across all node counts, with an average time of 0.1593 s from 100 to 600 nodes. However, DEDV-Hop, which uses optimizations to improve positioning accuracy, shows significantly higher computation times, with an average time of 4.4066 s from 100 to 600 nodes. WOADV-Hop, ABCDV-Hop, 3D DEH DV-Hop, VCR IDV-Hop, RNN-LSTM, and OLSTM-DV-Hop are algorithms that improve the accuracy and efficiency of DV-Hop. WOADV-Hop introduces weighting mechanisms, reducing positioning errors but increasing computation times. ABCDV-Hop has a slightly higher average computation time of 0.1498 s, while 3D DEH DV-Hop has a moderate computation time of 0.302 s. RNN-LSTM has an increased computation time with increasing node count, while OLSTM-DVHop reduces positioning errors by 56.8% with an average computation time of 0.092 s.

The computation time of the evaluated algorithms under different anchor node ratios is shown in Table 3, with an increase in anchor node ratio generally resulting in increased computational overhead for calculating average hop distance.

The DV-Hop algorithm has relatively large positioning errors but the shortest computation time in all anchor node ratios. In the 5% to 35% anchor node ratios range, the average computation time for DV-Hop is 0.071 s. WOADV-Hop improves upon the traditional DV-Hop algorithm, reducing positioning errors at the cost of slightly increased computation time. The average computation time for WOADV-Hop across anchor node ratios is 0.0966 s. The DEDV-Hop algorithm has the longest computation time, averaging 6.5241 s, due to the introduction of genetic algorithm-based optimizations. ABCDV-Hop exhibits about twice the computation time of DV-Hop but delivers higher positioning accuracy under the same conditions, with an average computation time of 0.1916 s. The 3D DEH DV-Hop algorithm shows relatively stable computation times, averaging 0.193 s. VCR IDV-Hop also consistently performs with an average computation time of 0.1944 s. The RNN-LSTM model shows increased computation time at higher anchor node ratios, with an average of 0.524 s. Finally, OLSTM-DVHop offers improved positioning accuracy, reducing positioning errors by 74.6% compared to DV-Hop while maintaining reasonable computation times, with an average of 0.074 s across anchor node ratios.

Performance of error correction mechanism in node positioning

The paper uses a random uniform distribution scenario to explore the impact of node randomness on positioning error under uniform node deployment. It also introduces a C-shape network topology for simulating sensor nodes along rugged terrain bending contours with non-uniform attenuation, an S-shape network topology for complex scenarios with multiple obstacles, and a C-shape network topology for uniform obstacles along the node deployment contour.

The research evaluates the proposed algorithms' robustness, accuracy, and comprehensive localization performance under various parameters, total node counts, and beacon node ratios.

Figure 9a presented the efficacy of the DV-Hop model, where the ALE ranges between 0.71 and 0.79, indicating higher errors in lower anchor node ratios (5–10%). Figure 9b shows the reliability of the WOADV-Hop model, where the ALE decreases from 0.72 to 0.64 as the number of anchor nodes increases, demonstrating improved localization accuracy. Figure 9c depicts the findings of the ABCDV hop algorithm, where the ALE decreases from 0.62 to 0.54 as the number of anchor nodes gets higher, demonstrating improved localization accuracy. Additionally, more significant numbers of nodes contribute to further improvement in performance. Figure 9d reveals the efficacy of the 3D DEDV-Hop algorithm, where the ALE decreases from 0.625 to 0.45 as the number of anchor nodes increases, indicating an improved localization accuracy. In addition, raising the number of nodes further enhances performance by reducing localization error. Figure 9e shows the evaluation of the 3D DEH DV-Hop algorithm, where the ALE decreases from 0.54 to 0.44 as the number of anchor nodes increases, indicating improved localization accuracy with higher anchor node ratios. Figure 9f illustrates the result of the VCR IDV-Hop algorithm, where the ALE decreases from 0.58 to 0.44 as the number of anchor nodes boosts, indicating improved localization accuracy as the anchor node ratio rises. Figure 9g illustrates the reliability of the RNN-LSTM algorithm, where the ALE decreases from 0.52 to 0.40 as the number of anchor nodes gets higher, indicating a significant improvement in localization accuracy. Additionally, higher node densities contribute to further reductions in error, suggesting that more nodes enhance the algorithm's accuracy. Figure 9h highlights

Algorithm	Anchor node ratio						
	5%	10%	15%	20%	25%	30%	35%
DV-Hop[14]	0.061	0.064	0.065	0.063	0.081	0.082	0.085
DEDV-Hop[41]	5.9047	5.9931	6.858	6.917	6.812	6.239	5.345
WOADV-Hop[42]	0.099	0.097	0.090	0.097	0.098	0.096	0.099
ABCDV-Hop[43]	0.177	0.186	0.198	0.199	0.195	0.198	0.1845
3D DEH DV-Hop[44]	0.196	0.193	0.197	0.201	0.198	0.203	0.193
VCR IDV-Hop[46]	0.157	0.159	0.188	0.196	0.172	0.1944	5.334
RNN-LSTM[45]	0.146	0.183	0.187	0.095	0.381	0.524	2.813
OLSTM-DVHop	0.073	0.074	0.076	0.067	0.073	0.089	0.074

Table 3. Computation time of positioning algorithms with different anchor node ratios (s).

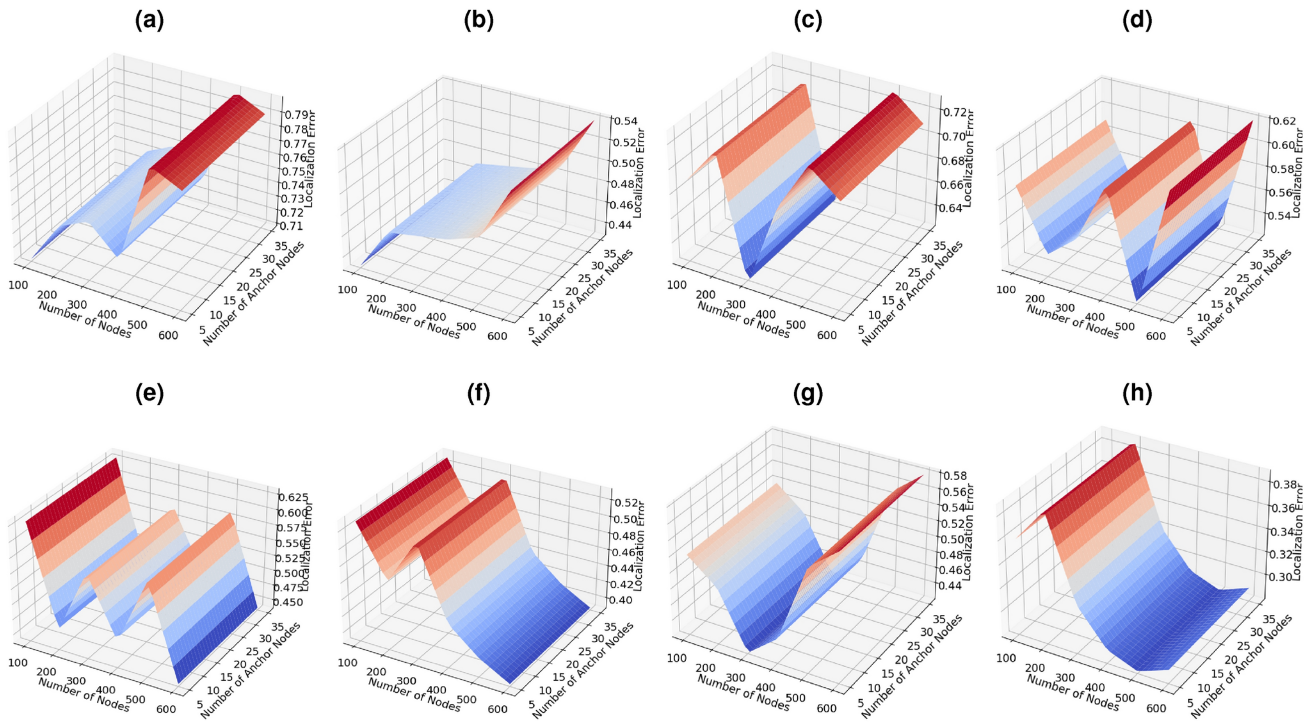

Fig. 9. Comprehensive localization performance of algorithms under different total node counts and beacon node ratios. (**a**) DV-Hop algorithm, (**b**) WOADV-Hop, (**c**) ABCDV-Hop, (**d**) DEDV-Hop, (**e**) 3D DEH DV-Hop, (**f**) VCR IDV-Hop, (**g**) RNN-LSTM, (**h**) proposed model.

the best efficiency of the proposed model at higher anchor node ratios and node counts. The ALE decreases as low as 0.32, demonstrating the superior precision and stability of the suggested OLSTM-DV-Hop algorithm in scenarios with higher node counts (up to 600) and larger anchor node ratios.

Discussion and conclusion

The study proposed an improved DV-Hop (OLSTM-DVHop) localization algorithm for WSNs based on LSTM. The proposed algorithm addresses critical limitations of the traditional DV-Hop algorithm by incorporating an LSTM model to enhance the accuracy of node positioning, particularly in scenarios with uneven error distribution. The LSTM model effectively captures the nonlinear relationships between hop counts and actual distances, significantly improving positioning accuracy. The simulation results confirmed the superiority of the OLSTM-DVHop algorithm over several other DV-Hop-based algorithms. The algorithm consistently demonstrated lower positioning errors and higher accuracy across various network topologies and node densities. Additionally, introducing an error correction mechanism further reduced the Localization Error, making the algorithm robust against the common inaccuracies encountered in WSNs. However, the increased computational complexity associated with the LSTM model and the need for extensive training data highlight areas for future improvement. Despite these challenges, the OLSTM-DVHop algorithm is a significant step in developing more accurate and reliable WSN localization methods.

Although the proposed OLSTM-DVHop localization algorithm improved positioning accuracy and error correction within WSNs, it has limitations. One of the primary shortcomings of the approach is its reliance on a substantial amount of training data to allow the LSTM model to perform optimally. In scenarios where labeled data are scarce or difficult to obtain, the algorithm's performance may degrade, reducing its effectiveness in real-world applications. Another challenge is the increased computational complexity and processing time caused by the fusion of LSTM and the traditional DV-Hop algorithm, which may not be suitable for WSNs deployed in environments with critical energy efficiency and low power consumption, such as remote locations. The algorithm's efficacy against various noise and environmental factors in obstacles interferes with signal propagation and demands more testing. The simulation results are promising, but the real-world deployment may reveal significant challenges not considered in the controlled simulation environment. The OLSTM-DVHop algorithm has been optimized to reduce computational requirements, making it suitable for energy-constrained WSNs. Future research will fuse the ML models that require less data or computational power. Moreover, more experimental findings in diverse real-world environments are needed to validate its applicability and adaptability. Hybrid models combining LSTM with other lightweight algorithms could provide a balanced approach. Environmental noise, varying node densities, and signal attenuation will also be incorporated to assess the proposed algorithm's robustness. Such enhancements will provide a comprehensive understanding of its applicability in real-world deployments.

Data Availability

All data generated or analyzed during this study are included in this published article.

Received: 19 October 2024; Accepted: 10 March 2025
Published online: 28 April 2025

References

1. Mohapatra, H., Rath, A. K., Lenka, R. K., Nayak, R. K. & Tripathy, R. Topological localization approach for efficient energy management of wsn. *Evolutionary intelligence* **17**, 717–727 (2024).
2. Soori, M., Arezoo, B. & Dastres, R. Artificial intelligence, machine learning and deep learning in advanced robotics, a review. *Cognitive Robotics* **3**, 54–70, 10/gs4cw6 (2023).
3. Devi. Long range-based low-power wireless sensor node. *ETRI Journal* (2023). Accessed: Mar. 07, 2024.
4. Bensaid, R., Mnaouer, A. B. & Boujemaa, H. Energy efficient adaptive sensing framework for wsn-assisted iot applications. *IEEE Access* (2024).
5. Fawad, M. et al. Enhancing localization efficiency and accuracy in wireless sensor networks. *Sensors* **23**, 2796 (2023).
6. Bakar, K. B. A., Zuhra, F. T., Isyaku, B. & Sulaiman, S. B. A review on the immediate advancement of the internet of things in wireless telecommunications. *IEEE Access* **11**, 21020–21048 (2023).
7. Hussain, S. *et al.* An adaptive intrusion detection system for wsn using reinforcement learning and deep classification. *Arabian Journal for Science and Engineering* 1–15 (2024).
8. Yuvarasu, M., Balaram, A., Chandramohan, S. & Sharma, D. K. A performance analysis of an enhanced graded precision localization algorithm for wireless sensor networks. *Cybernetics and Systems* 1–16 (2023).
9. Chen, T., Hou, S. & Sun, L. An enhanced dv-hop positioning scheme based on spring model and reliable beacon node set. *Computer Networks* **209**, 1089–1096 (2022).
10. Rathnayake, R., Maduranga, M. W. P., Tilwari, V. & Dissanayake, M. B. Rssi and machine learning-based indoor localization systems for smart cities. *Eng* **4**, 1468–1494 (2023).
11. Zhang, D., Zhang, X. & Xie, F. Research on location algorithm based on beacon filtering combining dv-hop and multidimensional support vector regression. *Sensors (Basel)* **21**, 451–458 (2021).
12. Chen, Y., Li, X. & Ding, Y. An improved dv-hop localization algorithm for wireless sensor networks. In *Proceedings of the 2018 13th IEEE Conference on Industrial Electronics and Applications (ICIEA)*, 1831–1836 (F 31 May-2 June 2018, 2018).
13. Han, D., Yu, Y., Li, K.-C. & de Mello, R. F. Enhancing the sensor node localization algorithm based on improved dv-hop and de algorithms in wireless sensor networks. *Sensors* **20**, 343 (2020).
14. Cui, Z., Sun, B. & Wang, G. A novel oriented cuckoo search algorithm to improve dv-hop performance for cyber-"physical systems. *Journal of Parallel and Distributed Computing* **103**, 42–52 (2017).
15. Gul, O. M., Erkmen, A. M. & Kantarci, B. Ntn-aided quality and energy-aware data collection in time-critical robotic wireless sensor networks. *IEEE Internet of Things Magazine* **7**, 114–120 (2024).
16. Survey and recommendations. Oliveira, L. L. d., Eisenkraemer, G. H., Carara, E. A., Martins, J. B. & Monteiro, J. Mobile localization techniques for wireless sensor networks. *ACM Transactions on Sensor Networks* **19**, 1–39 (2023).
17. Lakshmi, Y. V., Singh, P., Mahajan, S., Nayyar, A. & Abouhawwash, M. Accurate range-free localization with hybrid dv-hop algorithms based on pso for uwb wireless sensor networks. *Arabian Journal for Science and Engineering* **49**, 4157–4178 (2024).
18. Bahl, P. & Padmanabhan, V. N. Radar: An in-building rf-based user location and tracking system. In *Proceedings IEEE INFOCOM 2000 Conference on computer communications Nineteenth annual joint conference of the IEEE computer and communications societies (Cat No 00CH37064)*, 775–784 (IEEE, 2000).
19. Thao, L. *et al.* Optimize cleaning school's restroom by wsn and lstm approach. *Journal of Intelligent & Fuzzy Systems: Applications in Engineering and Technology* **45**, 1057–1065, 10/gtkx8k (2023).
20. Kanwar, V. & Kumar, A. Dv-hop based localization methods for additionally deployed nodes in wireless sensor network using genetic algorithm. *Journal of Ambient Intelligence and Humanized Computing* **11**, 5513–5531 (2020).
21. Wang, P., Wang, X., Li, W., Fan, X. & Zhao, D. Dv-hop localization based on distance estimation using multi-node and hop loss in iot. *IEEE Internet of Things Journal* (2024).
22. Zhou, Y., Ao, X. & Xia, S. An improved apit node self-localization algorithm in wsn. In *Proceedings of the 2008 7th World Congress on Intelligent Control and Automation*, 7582–7586 (IEEE, 2008).
23. Abou-Rjeily, C. & El-Zahr, S. Deep reinforcement learning based relaying for buffer-aided cooperative communications. *Physical Communication* **59**, 102086, 10/gtkxxk (2023).
24. Messous, S., Liouane, H. & Liouane, N. Improvement of dv-hop localization algorithm for randomly deployed wireless sensor networks. *Telecommunication Systems* **73**, 75–86 (2019).
25. Latha, T. S., Rekha, K. B. & Ferede, A. W. Lightweight apit with bat optimization with simulated annealing localization for resource-constrained sensor networks. *Wireless Communications and Mobile Computing* **2023**, 7982038 (2023).
26. Phoemphon, S., So-in, C. & Leelathakul, N. Optimized hop angle relativity for dv-hop localization in wireless sensor networks. *IEEE Access* **6**, 78149–78172 (2018).
27. Wang, P., Li, H. & Cai, X. 3d many-objective dv-hop localization model with nsga3. *Soft Computing* **28**, 6171–6183 (2024).
28. Huang, B. Lstm-dl: A counting optimization approach for lstm-based dv-hop localization model in wireless sensor networks. *Highlights in Science, Engineering and Technology* **98**, 498–506 (2024).
29. Tariq, S. M. & Al-Mejibli, I. S. Wsn localization method based on hybrid pso-grnn approach. *International Journal of Intelligent Engineering & Systems* **16** (2023).
30. Kim, K. R., Lim, A. & Cho, J. H. Artificial neural network for indoor localization based on progressive subdivided quadrant method. *Applied Sciences* **13**, 8545 (2023).
31. Han, G., Chao, J. & Zhang, C. The impacts of mobility models on dv-hop based localization in mobile wireless sensor networks. *Journal of Network and Computer Applications* **42**, 70–79 (2014).
32. Sharma, G. & Kumar, A. Modified energy-efficient range-free localization using teaching-learning-based optimization for wireless sensor networks. *IETE Journal of Research* **64**, 124–138 (2017).
33. Zhao, W., Su, S. & Shao, F. Improved dv-hop algorithm using locally weighted linear regression in anisotropic wireless sensor networks. *Wireless Personal Communications* **98**, 3335–3353 (2017).
34. Kaur, A., Kumar, P. & Gupta, G. P. Nature inspired algorithm-based improved variants of dv-hop algorithm for randomly deployed 2d and 3d wireless sensor networks. *Wireless Personal Communications* **101**, 567–582 (2018).
35. Wang, P., Zhou, R., Fan, X. & Zhao, D. A distance estimation model for dv-hop localization in wsns. *IEEE Transactions on Vehicular Technology* **72**, 5290–5299 (2022).
36. Hamza, M. A. et al. Optimal bidirectional lstm for modulation signal classification in communication systems. *Cmc-Comput. Mater. Contin* **72**, 3055–3071 (2022).
37. Etzlinger, B., Ganhör, A. & Karoliny, J. Wsn implementation of cooperative localization. In *Proceedings of the 2020 IEEE MTT-S International Conference on Microwaves for Intelligent Mobility (ICMIM)*, 1–4 (IEEE, 2020).

38. J., H., T., X. & Y., W. A reinforcement learning handover parameter adaptation method based on lstm-aided digital twin for udn. *Sensors* **23**, 2191–2204 (2023).
39. S.-H., P., B.-Y., L. & M.-J., K. Development of a soil moisture prediction model based on recurrent neural network long short-term memory (rnn-lstm) in soybean cultivation. *Sensors* **23**, 1976–1985 (2023).
40. Sami, A. *et al.* Federated learning for sarcasm detection: A study of attention-enhanced bilstm, gru, and lstm architectures. *IEEE Access* (2024).
41. Gong, X., Wang, X., Guo, J. & Wang, A. DEDV: A data collection method for mobile sink based on dynamic estimation of data value in WSN. In *2016 International Conference on Networking and Network Applications (NaNA)*, https://doi.org/10.1109/NaNA.2016.63 (IEEE, Northwest University, China, 2016).
42. Chai, Q.-W. *et al.* A parallel WOA with two communication strategies applied in DV-Hop localization method. *EURASIP Journal on Wireless Communications and Networking* 1–10, https://doi.org/10.1186/s13638-020-01753-4 (2020).
43. Zhang, W., Yang, X. & Song, Q. Improved dv-hop algorithm based on artificial bee colony. *International Journal of Control and Automation* **8**, 135–144. https://doi.org/10.14257/ijca.2015.8.11.14 (2015).
44. Liu, Z. et al. Enhancing 3d localization in wireless sensor network: a differential evolution method for the dv-hop algorithm. *Journal of Supercomputing* **80**, 23120–23146. https://doi.org/10.1007/s11227-024-06293-5 (2024).
45. Mahmood, T., Li, J., Saba, T., Rehman, A. & Ali, S. Energy optimized data fusion approach for scalable wireless sensor network using deep learning-based scheme. *Journal of Network and Computer Applications* **224**, 103841 (2024).
46. Cao, Y., Wu, X. & Xu, J. Improved dv-hop localization algorithm based on variable communication radius for wireless sensor networks. *Wireless Personal Communications* https://doi.org/10.1007/s11277-024-11571-9 (2024).

Acknowledgements
The authors extend their appreciation to the Princess Nourah bint Abdulrahman University Researchers Supporting Project number (PNURSP2025R513), Princess Nourah bint Abdulrahman University, Riyadh, Saudi Arabia. Authors are also thankful to Prince Sultan University, Riyadh Saudi Arabia for support of Article Processing Charges (APC) for this publication.

Author contributions
Conceptualization, Amjad Tariq, Tahani, Ahmad, and Tanzila; Data Curation, Amjad Tariq, Tahani, Ahmad, and Tanzila; Formal analysis, Amjad Tariq, Tahani, Ahmad, and Tanzila; Funding acquisition, Tahani Jaser Alahmadi; Methodology, Amjad Tariq, Tahani, Ahmad, and Tanzila; Software, Amjad Tariq, Tahani, Ahmad, and Tanzila; Supervision, Tanzila.

Funding
The Princess Nourah bint Abdulrahman University Researchers Supporting Project number (PNURSP2025R513), Princess Nourah bint Abdulrahman University, Riyadh, Saudi Arabia.

Declarations

Competing interests
The authors declare no competing interests.

Institutional Review Board Statement
Not applicable. All methods were carried out in accordance with relevant guidelines and regulations.

Informed Consent Statement
Not applicable.

Additional information
Correspondence and requests for materials should be addressed to T.J.A.