Mathematical Biosciences and Engineering

MBE, 19(9): 9098–9124.
DOI: 10.3934/mbe.2022423
Received: 17 March 2022
Revised: 30 May 2022
Accepted: 05 June 2022
Published: 22 June 2022

http://www.aimspress.com/journal/MBE

Research article

Harris hawks optimization algorithm and BP neural network for ultra-wideband indoor positioning

Xiaohao Chen[1], Maosheng Fu[1,*], Zhengyu Liu[1], Chaochuan Jia[1,2,*] and Yu Liu[1]

[1] College of Electronics and Information Engineering, West Anhui University, Lu'an, China
[2] Robot Research Center, Shandong University of Science and Technology, Qingdao, China

* **Correspondence:** Email: fums@wxc.edu.cn, ccjia@hfcas.ac.cn; Tel: +8618860453758; Fax: +8618860453758.

Abstract: Traditional back propagation neural networks (BPNNs) for ultrawideband (UWB) indoor localization can effectively improve localization accuracy, although there is high likelihood of becoming trapped in nearby minima. To solve this problem, the random weights and thresholds of the BPNN are optimized using the Harris Hawks optimization algorithm (HHO) to obtain the optimal global solution to enhance the UWB indoor positioning accuracy and NLOS resistance. The results show that the predicted trajectory of the HHO and BPNN hybrid algorithm (HHO-BP) matches the actual position in the two-dimensional localization scenario with four base stations; the optimized average positioning error is effectively reduced in both indoor LOS and NLOS environments. In the LOS environment, the total mean error of the traditional BPNN algorithm is 6.52 cm, which is 26.99% better than the UWB measurement error; in the NLOS environment, the total mean error of the conventional BPNN is 14.82 cm, which is 50.08% better than the UWB measurement error. The HHO–BP algorithm is further optimized on this basis, and the total mean error in the LOS environment is 4.50 cm, which is 22.57% better than the conventional BPNN algorithm; in the NLOS environment, the total mean error is 9.56 cm, which is 17.54% better than the conventional BPNN algorithm. The experimental findings suggest that the approach has greater calibration accuracy and stability than BPNN, making it a viable choice for scenarios requiring high positional precision.

Keywords: UWB; BPNN; Harris hawks optimization algorithm; line-of-sight; non-line-of-sight indoor positioning

1. Introduction

With the rapid development of innovations in communication technology and the notoriety of remote systems, the desire for enhanced indoor positioning continues to expand [1]. However, when facing complex indoor environments, such as with the real-time positioning of personnel, equipment and materials in office buildings, factories and construction sites [2], real-time care and management of the elderly [3], precise electronic positioning guidance inside hospitals [4] and precise location trajectories of supervised persons in prisons [5], there is an expanding requirement for high-precision situating innovation. Thus, strategies for advancing the exactness of indoor situating tasks have become of increasing interest to a range of researchers.

Indoor positioning technology is developing rapidly today and is classified as building-dependent or building-independent, depending on the sensors used for positioning [6]. The most common building-independent indoor positioning techniques are pedestrian dead reckoning (PDR) systems [7] and image-based techniques that use cameras [8]. Although the noise is small and stable in the short term when using PDR systems, such approaches cannot be used for long times due to the accumulation of errors, and toward solving this problem, the mainstream research direction is currently the fusion algorithm. Various fusion algorithms are proposed in [9–11], such as fusion algorithms with Wi-Fi, Bluetooth, RSS fingerprinting, etc., which effectively reduce positioning error and improve robustness. The four aspects of a typical inertial measurement unit(IMU)-based PDR system framework are direction updating, step detection, step estimation and map matching [12]. Navigation systems include techniques such as orientation updating and map matching, whereas step detection and estimation are specialized to human motion since they exploit the constraint of pedestrian steps. The camera is used for positioning in image-based indoor positioning tasks. The camera used in such applications can be monocular [13], stereo [14] or RGB-D [15]. Because it is difficult to obtain accurate positioning results with a single method, a hybrid positioning method in such a positioning system is introduced as the reliable option. The existing techniques for IMU-based positioning, camera-based positioning and various existing hybrid methods for indoor positioning are discussed in [16], and a hybrid indoor positioning system that uses IMU sensors and smartphone cameras is proposed, which effectively improves indoor positioning accuracy.

Building-independent positioning technologies, of which there are many types, fall into two main categories: namely, those that utilize the existing building infrastructure and those that require dedicated infrastructure. The former category includes Wi-Fi, cellular and Bluetooth. Most Wi-Fi-based positioning systems can be used as RSS-based measurement systems. Various Wi-Fi restrictions must also be considered: for example, signal attenuation in static contexts such as walls and floors, as well as changes in interior environments [17]. Although the cellular network has a wide coverage area, it still suffers from low positioning accuracy. One of the disadvantages of employing Bluetooth technology for localization is that it performs the device discovery operation for each location finding event; as a result, the localization latency (10–30 s) and consumption are greatly increased. As a result, the latency of a Bluetooth device is inadequate for real-time positioning applications. However, for the previously described Wi-Fi localization, a hybrid deep learning model (HDLM)-based indoor localization system was developed in [18]. Instead of raw RSSI signals from APs, the proposed HDLM-based localization system uses RSSI heatmaps. As a result, the Wi-Fi-RSSI signal-based localization system's localization performance is improved. For the latter, the main positioning technologies are RFID, UWB, infrared, ultrasonic, ZigBee, VLC, acoustic, etc. RFID technology requires a large number of tags, which do not meet the requirements of real-time positioning, and its positioning accuracy is related to the deployment density of tags [19]. Indoor positioning systems based on infrared possess several drawbacks, such as security and privacy concerns. The position determination of infrared signals is limited due to interference from fluorescent light and sunlight. In addition, the hardware and maintenance expenses of IR-based indoor systems are high [20]. Reflected waves propagated by other sources can cause severe interference for ultrasound devices (e.g., metal collisions) [21]. The ZigBee positioning accuracy is low, as is its data transfer rate [22].

UWB is a remote correspondence innovation that transfers a bandwidth of more than 1 GHz. Due to its strong anti-interference performance, resistance to multipath interference, high transmission rate, low transmission power and increased penetration capability, UWB positioning technology is widely used for positioning in indoor environments. UWB can provide high positioning accuracy [23–25]. Currently, the various UWB ranging methods can be classified into time of arrival (TOA), received signal strength (RSS) and time difference of arrival (TDOA) [26]. However, in complex indoor environments, where walls or other signals block objects, the signal transmission distance increases, and the transmission time lengthens, significantly reducing UWB positioning accuracy. A single indoor positioning method can no longer meet the growing requirements for localization accuracy. Therefore, mixed localization algorithms are becoming increasingly attractive [27].

The localization equation is usually a nonlinear equation based on TOA or TDOA; as a result, the equation must be converted into a linear equation for solving [28]. Currently, nonlinear problems are mainly solved by the least-squares estimation (LSE) algorithm, Fang algorithm, Chan algorithm and Taylor algorithm. The weighting matrix of the LSE algorithm should be decided according to the precision of the measurement data; in practical applications, the selection of the weighting matrix is difficult. The best estimation performance of the algorithms developed by Chan and Fang is achieved when the time difference measurement error is small, although the performance of both algorithms is seriously affected by the large time difference measurement error in the real channel environment [29]. In recent years, deep learning approaches have been commonly employed for indoor positioning because of their excellent nonlinear mapping capabilities and self-adaptive and self-learning abilities [30]. For example, Zhuo et al. suggested a UWB locating technique based on BPNN that is applied to automatic parking, showing good real-time performance and accuracy [31]. In [32], a long short-term memory (LSTM) network is proposed to reduce the position error in UWB systems; this network solves the LOS problem associated with UWB systems and improves the positioning accuracy. Shuai et al. applied the 60 GHz pulsed-IR algorithm to a BPNN, which was used to improve the ability of conventional algorithms to accurately estimate signal fading due to the complexity of indoor channels, thus improving localization accuracy [33]. Lian et al. proposed an optimized K-means clustering BPNN UWB positioning approach, which enhances the original data in the UWB positioning system and preprocesses the K-means clustering algorithm to filter the data with a significant positional deviation, and the experimental measurement accuracy can reach 26 cm with good stability [34]. Zhang and Jin proposed a BP neural network indoor positioning algorithm based on VLC. Through simulation experiments, the authors showed that the algorithm has a lower relative positioning error and better indoor positioning accuracy than the MDS-MAP and MDS-MAP(P) algorithms [35].

However, the BPNN is sensitive to random weights and thresholds, which enhances the likelihood of being trapped in nearby minima, thus generating significant localization errors [36]. Therefore, numerous optimization algorithms have been proposed to reduce the downsides of the BPNN. Among them, population intelligence optimization algorithms are the most widely applied algorithms toward solving the inherent defects of traditional training algorithms [37]. For example, a location fusion method of genetic algorithm (GA) optimization of the BP neural network that effectively reduces the localization error was proposed by Yang et al. [38]. Li et al. [39] proposed a mixed model that combines the cuckoo search optimization algorithm (CS) with the BPNN, where the random weights and thresholds of the BPNN are optimally trained, and the measured time differences are corrected after training. Finally, the Chan algorithm is used for localization calculation according to the corrected time difference values. In addition to these studies, the current swarm intelligence algorithm is developing rapidly, and better results have been demonstrated [40–44]. Previous literature has investigated indoor localization methods using the particle swarm optimization algorithm and BPNN hybrid algorithm (PSO-BP) combined with RFID using artificial intelligence techniques [45] and UWB indoor localization methods using the immune algorithm and BPNN hybrid algorithm (IA-BP) [46]. The above research shows that artificial intelligence techniques show strong capability in UWB positioning. It is essential to study new artificial intelligence techniques and prediction models to improve UWB positioning accuracy [47].

As indicated by the no free lunch (NFL) theorem [48], theoretically, a certain algorithm cannot be treated as a generic global optimal optimization algorithm. Therefore, although a range of algorithms is used to optimize UWB indoor positioning, the presented algorithms are far from being able to solve all problems or be applied to all indoor environments. To the authors' best knowledge, UWB indoor localization has not yet been attempted using a combination of the HHO algorithm and BPNN.

This study proposes a new hybrid algorithm, called HHO-BP, to improve UWB indoor positioning accuracy and resistance to non-line-of-sight ranges. The algorithm uses the HHO algorithm to optimize the BPNN to find a set of optimal weights and thresholds, thereby improving the positioning accuracy. Finally, the optimization performance of the HHO-BP algorithm and the conventional BPNN are compared and analysed through experimental validation.

The major contribution of this paper is to propose a new hybrid algorithm, HHO-BP, for optimizing ultrawideband indoor positioning, outlined as follows:

(1) A new algorithm for hybrid localization, namely, HHO-BP, is proposed.

(2) Combining the BP neural network with an intelligent population algorithm, the structure of the BP neural network is optimized, thus solving the local optimal solution problem.

(3) The algorithm greatly improves the accuracy and anti-jamming capability of ultrawideband indoor positioning, with centimetre-level accuracy for both line-of-sight and non-line-of-sight positioning.

The structure of the paper is as follows: First, the principles of UWB localization and BPNN are explained in Section 2. Section 3 introduces the principle of the proposed swarm intelligence algorithm (HHO), as well as the constructed new model. Section 4 offers the findings as well as a discussion and analysis. Finally, in Section 5, the study's findings are presented.

2. Model building

2.1. UWB position principle

The principle of the trilateral positioning algorithm is as follows: The three base stations are located at $A_0(x_1,y_1)$, $A_1(x_2,y_2)$ and $A_2(x_3,y_3)$, and it is known that the distance values of the tag to be measured, $Tn(x_0,y_0)$, to the 3 base stations are d_1, d_2 and d_3, as displayed in Figure 1.

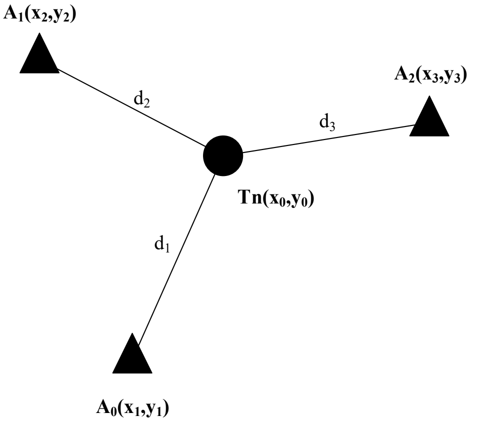


Figure 1. Trilateral positioning algorithm.

In the case of two-dimensional positioning, according to the Pythagorean theorem, the formula for calculating the position of the tag, Tn, and the focus are attained using Eq (1).

$$\begin{aligned} d_1 &= \sqrt{(x_1-x_0)^2+(y_1-y_0)^2} \\ d_2 &= \sqrt{(x_2-x_0)^2+(y_2-y_0)^2} \\ d_3 &= \sqrt{(x_3-x_0)^2+(y_3-y_0)^2} \end{aligned} \tag{1}$$

Under ideal circumstances, the distance values of tag Tn to base Stations A_0, A_1 and A_2 are accurate; tag Tn has only one correct solution, as shown by the red dot in Figure 2.

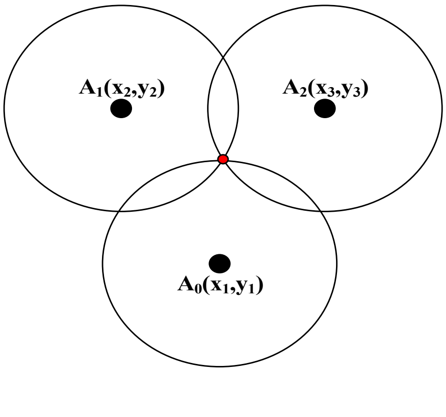


Figure 2. Ideal circumstances.

Under actual circumstances, real distance measurement is subject to error; the reasons for the error include the error of the range itself and the error of the actual scene, such as obstruction or metal, which affects the penetration and transmission of electromagnetic waves. In a line-of-sight environment, the measured value fluctuates above and below the actual value; however, in a non-line-of-sight environment, the distance measured is theoretically higher than the actual value due to the presence of occlusion between the base station and the tag, resulting in the effect shown below, where the solved *Tn* coordinate is no longer a point. Nevertheless, it is a region, so we need to find an optimal solution in the red area, as shown in Figure 3.

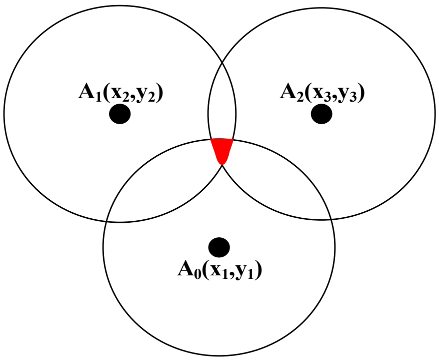


Figure 3. Actual circumstances.

2.2. BPNN model

There are many models of artificial neural networks, but the foremost broadly utilized, natural and simple-to-understand is the backpropagation algorithm, called BPNN [49]. Training is performed through signal forward propagation and error backward propagation.

The network generally has a multilayer structure, including an input layer for data input, an output layer for result output and several hidden layers, and each hidden layer contains one or more neurons. The topological diagram of the BPNN is shown in Figure 4.

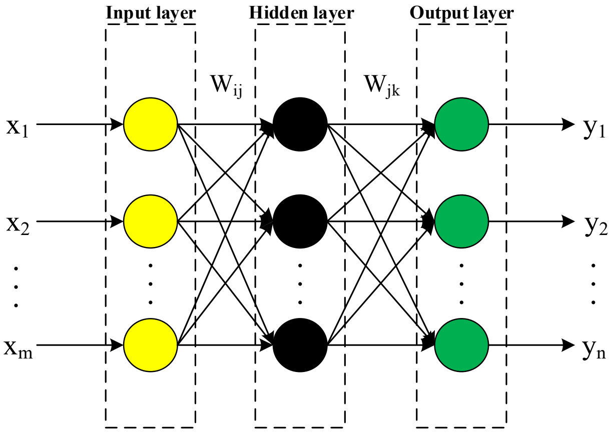


Figure 4. The topology of BPNN.

In the structure, *x1*, *x2*, …, *xm* are inputs, and *y1*, *y2*, …, *yn* are outputs. A hidden layer contains *s* neurons, the output function of the hidden layer is b_j , and the threshold values for the hidden and output layers are θ_j and θ_k , respectively. The most frequently used activation function is the sigmoid function, which is set to $S(x)$ here. f_1 and f_2 are the transfer functions of the hidden and output layers, respectively. The output of the model is set to y_k , and the expected output is t_k [50]. The above can be modelled as follows.

The output of the *Jth* neuron of the hidden layer is [50]

$$b_j = f_1\left(\sum_{i=1}^{m} w_{ij}x_i - \theta_j\right) \quad (i = 1,2,...,m; j = 1,2,...,s). \tag{2}$$

a. Calculate the output y_k of the output layer as in Eq (3) [50]:

$$\mathrm{y}_k = f_2\left(\sum_{i=1}^{s} w_{jk}b_j - \theta_k\right) \quad (j = 1,2,...,s; k = 1,2,...,n). \tag{3}$$

b. Define the error function by the actual output of the network as in Eq (4) [50]:

$$e = \sum_{k=1}^{n} (t_k - y_k)^2 . \tag{4}$$

c. The sigmoid activation function is given as Eq (5) [50]:

$$S(x) = \frac{1}{1+e^{-x}} . \tag{5}$$

3. The proposed hybrid positioning method

3.1. Harris hawks optimization (HHO) algorithm

A new bionic intelligent optimization approach, the Harris hawks optimization (HHO) algorithm, was developed in 2019. The method is a metaheuristic approach that combines Lévy flights to solve large multidimensional problems by simulating the collaborative group behaviour of Harris hawks while hunting. Compared with previous algorithms, this algorithm has a higher search capability, acceptable development and exploration and thus has been successfully applied to many practical engineering problems [51,52].

As a result, since BPNN is more sensitive to initial weights and thresholds, this research proposes the Harris hawks optimization method for BPNN random weights and threshold optimization.

The whole optimization process includes a global exploration phase, a transition phase and a local exploitation phase [52].

1) Exploration phase:

In HHO, the Harris hawks perch randomly on some locations in a waiting form, monitoring the area [*lb*, *ub*] to search for prey. The prey is simultaneously searched at random locations according to two different strategies, and the location update is performed with probability *q* during the iteration. The mathematical formula for the Harris hawks's location, which is closest to the prey, is the following:

$$X(t+1)=\begin{cases} X_{rand}(t) - \mathrm{r}_1 \cdot \left|X_{rand}(t) - 2\cdot \mathrm{r}_2 \cdot X(t)\right|, q \ge 0.5 \\ \left(X_{rabbit}(t) - X_b(t)\right) - r_3 \cdot \left(lb + r_4 \cdot (ub - lb)\right), q < 0.5 \end{cases}. \tag{6}$$

In Eq (6), $X(t+1)$ is the hawk's location in the following iteration, t. The location of the rabbit is $X_{rabbit}(t)$, and the current position of the hawk is $X(\mathrm{t})$. r_1 ,r_2 ,r_3 ,r_4 and q are all random numbers ranging from 0 to 1. The upper and lower bounds of the variables are represented by *ub* and *lb*, respectively, while $X_{rand}(t)$ is the hawk chosen at random from the population, which is updated in each iteration. $X_b(t)$ is the current hawk average location in Eq (7).

$$X_b(t)=\frac{1}{N}\sum_{i=1}^{N} X_i(t) \tag{7}$$

where N is the number of hawk, and in iteration t, $X_i(t)$ shows the location of each hawk.

2) Transition phase:

The HHO algorithm moves from exploration to exploitation and then between multiple exploitation behaviours depending on the energy, E, of prey escape, as shown below in Eq (8):

$$E = 2E_0(1-\frac{t}{T}) \tag{8}$$

where E_0 is the energy starting stage, and T is the highest number of iterations [52].

3) Exploitation phase:

Four distinct tactics are employed to recreate the assault in this phase, depending on how the prey flees and how the Harris hawk pursues it, combined with Lévy flights, representing the prey's chance of escaping [52].

a)

$$r \ge 0.5 \,\&\, |E| \ge 0.5$$

In this case, the rabbit still has enough energy, and the hawk aim to exhaust its energy before launching the assault. The modelling rules given in Eqs (9) and (10) [52]:

$$X(t+1) = \Delta X(t) - E\left|JX_{rabbit}(t) - X(t)\right|, \tag{9}$$

$$\Delta X(t) = X_{rabbit}(t) - X(\mathrm{t}). \tag{10}$$

To replicate the nature of the rabbit's movements, the J value varies randomly in each iteration, and the rabbit's random hop intensity throughout the escape phase is denoted by $J = 2(1-r_5)$. r_5 is a number in the range from 0 to 1. The disparities between the rabbit's prior location vector and the current position vector are $\Delta X(t)$.

b)

$$r \geq 0.5 \,\&\, |E| < 0.5$$

Under this condition, the rabbit is fatigued, and the hawk can launch a surprise assault. Equation (11) expresses how the model updates the rabbit's position [52]:

$$X(t+1) = X_{rabbit}(t) - E\left|\Delta X(t)\right|. \tag{11}$$

c)

$$r < 0.5 \,\&\, |E| \geq 0.5$$

The rabbit has enough energy in this situation to effectively flee while still constructing a soft envelope before the attack. The following rule, given in Eq (12), can be used to update the Harris hawks' positions [52]:

$$Y = X_{rabbit}(t) - E\left|JX_{rabbit}(t) - X(t)\right|. \tag{12}$$

The hawk will dive utilizing the following rule, Eq (13), using LF-based patterns:

$$Z = Y + S \times LF(D) \tag{13}$$

where D is the dimension, S is a one-dimensional random vector, and LF is the Levy flight function and is based on the equation below (Eq (14)).

$$LF(x) = 0.01 \times \frac{\mu \times \sigma}{|v|^{\frac{1}{\beta}}}, \sigma = \left(\frac{\Gamma(1+\beta) \times \sin(\frac{\pi\beta}{2})}{\Gamma(\frac{1+\beta}{2}) \times \beta \times 2^{(\frac{\beta-1}{2})}} \right)^{\frac{1}{\beta}} \tag{14}$$

Here, β is 1.5, and μ and v are random numbers between 0 and 1.

By calculation, Eq (15) can execute the ultimate plan to update the locations of the hawks:

$$X(t+1) = \begin{cases} \mathrm{Y} & \mathrm{F(Y)<F(H(}t\mathrm{))} \\ \mathrm{Z} & \mathrm{F(Z)<F(H(}t\mathrm{))} \end{cases}. \tag{15}$$

d)

$$r < 0.5 \,\&\, |E| < 0.5$$

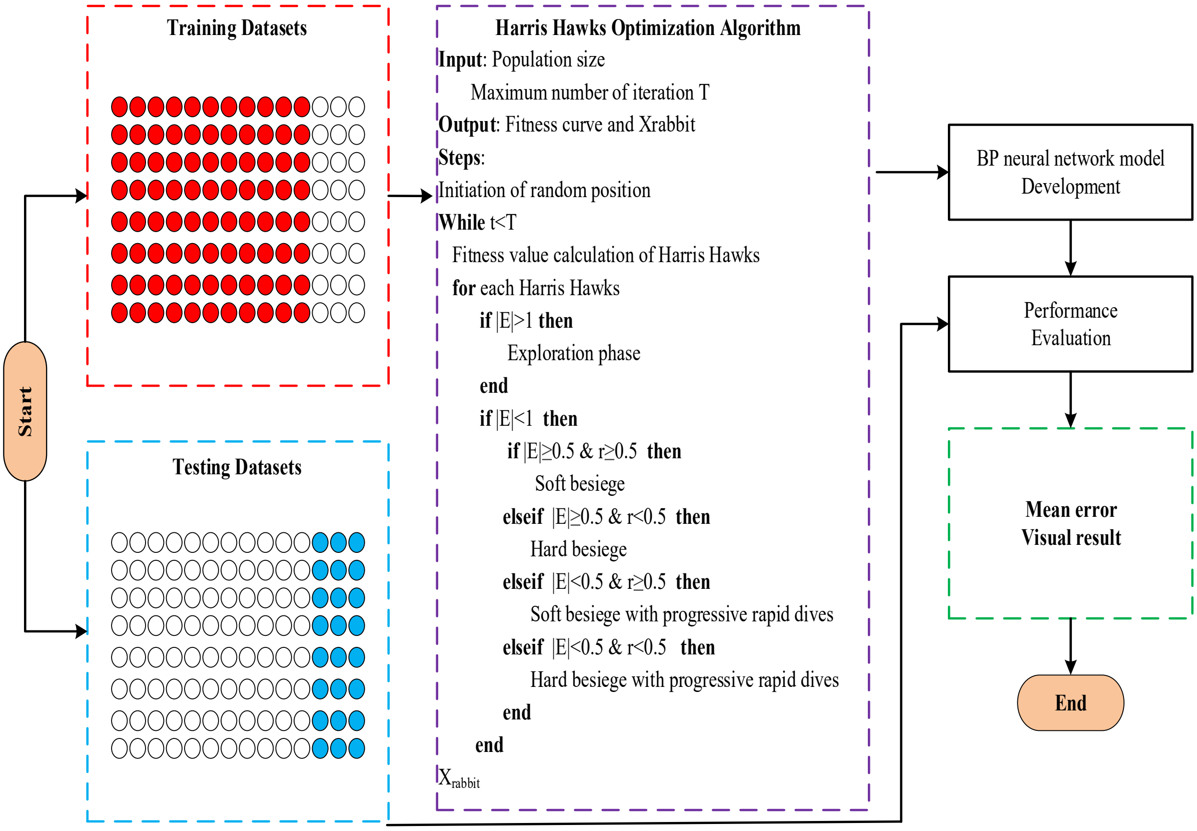
In this situation, the rabbit is unable to flee due to a lack of energy, so the hawks set up a hard siege. The modelling rules are given in Eqs (16) and (17) [52]:

$$Y = X_{rabbit}(t) - E\left|JX_{rabbit}(t) - X_b(t)\right|, \tag{16}$$

$$Z = Y + S \times LF(D). \tag{17}$$

Here, Y references Eq (15).

3.2. HHO-BP model



```
Input: Population size
    Maximum number of iteration T
Output: Fitness curve and Xrabbit
Steps:
Initiation of random position
While t<T
  Fitness value calculation of Harris Hawks
  for each Harris Hawks
    if |E|>1 then
      Exploration phase
    end
    if |E|<1 then
      if |E|≥0.5 & r≥0.5 then
        Soft besiege
      elseif |E|≥0.5 & r<0.5 then
        Hard besiege
      elseif |E|<0.5 & r≥0.5 then
        Soft besiege with progressive rapid dives
      elseif |E|<0.5 & r<0.5 then
        Hard besiege with progressive rapid dives
      end
    end
X_rabbit
```

Figure 5. A method for creating a Harris hawk optimization-BP neural network (HHO-BP) model is described.

Figure 5 depicts the development path of the HHO-BP model. First, the obtained database is divided into training and test datasets. The BP neural network is first set to random weights and thresholds, and then the training set data are input for training, where the Harris hawk optimization algorithm develops the final BP model by searching for the best combination of weights and thresholds in the BP model in a preset number of iterations. Finally, the test set data are fed into the final constructed BP neural network for predicting the UWB localization results. Performance metrics such as the mean error and position comparison can be used to evaluate the optimized BP model, and the actual and measured coordinates of the test set and the prediction results are used to test its performance and calculate the prediction error.

The computational complexity of HHO in this approach is determined using three key processes: initial optimization, fitness evaluation and candidate update. The computing complexity of the initialization process is $O(N)$ for all N. The update mechanism has a computational complexity of $O(T\times N)+O(T\times N\times D)$, which comprises updating the search agent location and finding the optimal solution, where T is the maximum number of iterations, set to 200 to assure convergence, and D is the problem's dimensionality. As a result, HHO's computational complexity is $O\left(N\times\left(T+TD+1\right)\right)$ [52].

4. Performance evaluation and analysis

4.1. Experimental scenario configuration

Through the following site environment and hardware equipment, UWB measurement data are acquired. Then, they are compared with the actual position, and visualized positioning points and error curves are plotted to see if the algorithm pair is effective and accurate.

4.1.1. Test environment and trajectory

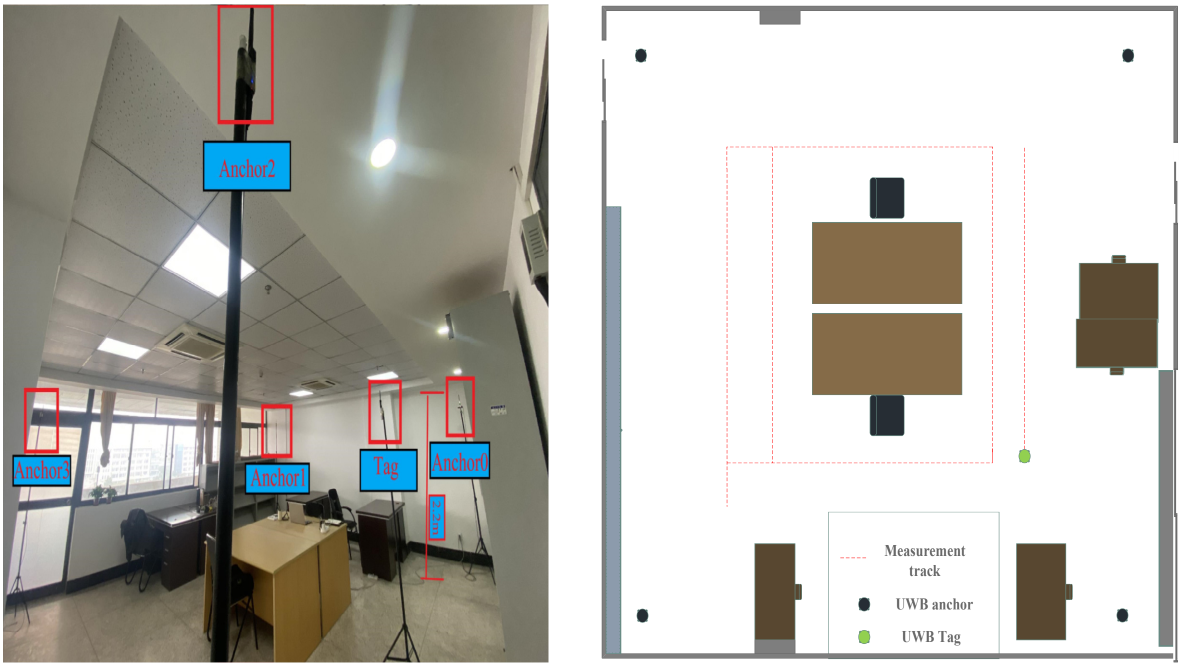
To test the accuracy of UWB localization and the algorithm's efficiency, a combination of offline data acquisition and MATLAB simulation is adopted. For the acquisition, an LOS environment and an NLOS environment were created, and the experimental sites are shown in Figures 6 and 7.

LOS environment: To create an indoor positioning scenario, four UWB anchors with heights of 2.2 m and positions of (0, 0), (0, 6.45), (5.58, 0) and (5.58, 6.45) were placed in the region illustrated in Figure 6. UWB anchor 0 is connected via Wi-Fi to a computer with the developed interactive interface installed, and the real-time position of the UWB tag is obtained through the host computer software to obtain indoor positioning data for the LOS environment.

NLOS environment: Because of the indoor environment's complexity, the height of the four UWB anchors was reduced to 1.4 m, and the positioning test points were built around the table using the shelter of the table to build a non-line-of-sight testing environment, as shown in Figure 7.

4.1.2. Major hardware equipment

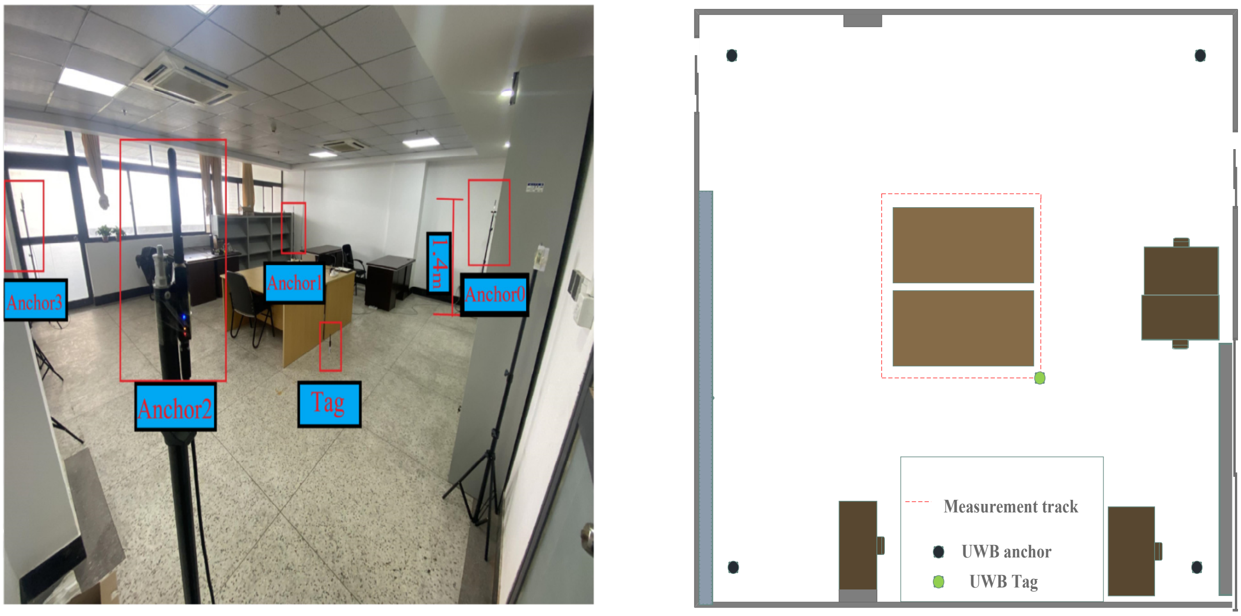
The DWM1000 UWB module used in the range circuit is made by DECAWAVE. The Wi-Fi is used to communicate with the PC to obtain UWB measurement data. Figure 8 depicts the system structure and hardware circuit diagram [53].



(a)

(b)

Figure 6. The UWB test field has a size of 6.45 × 5.58 m^2. The height of the UWB anchors is 2.2 m. The coordinate's origin is set to A0. (a) LOS distance actual view. (b) LOS distance plan and measurement trajectory.



(a)

(b)

Figure 7. NLOS experimental scene diagram. The height of the UWB anchors is 1.4 m. (a) NLOS distance actual view. (b) NLOS distance plan and measurement trajectory.

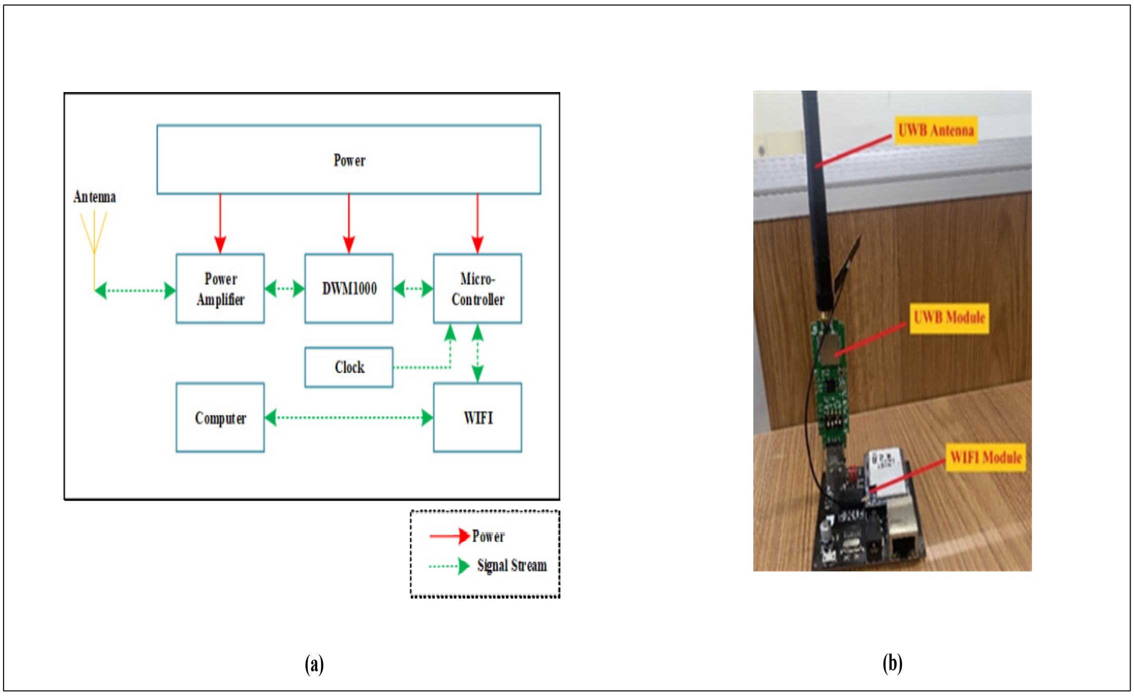


Figure 8. The UWB ranging system. (a) System structure diagram. (b) Hardware circuit diagram.

4.2. Results presentation and analysis

4.2.1. LOS environment experiment

The positioning data were collected separately at the same height at the site. Four label heights were tested (1.3, 1.55, 1.7 and 1.95 m) to ensure the feasibility of experimental rigor. The actual locations and the measured locations were used to form the sample set, which was divided into the training and test sets.

In Figure 9, the four panels a–d represent the total dataset of the four altitude UWB measurements and the actual coordinate positions. The black asterisks represent the actual coordinate positions measured with the metric ruler; the red diamonds represent the upper computer positioning software data. The data in the graph are all shifted by different magnitudes. To verify the effectiveness and accuracy of the algorithm, the simulation results were subsequently compared with the redraw data. Additionally, to better analyse the error of the original UWB positioning, the error curve is plotted using MATLAB 2020 software, as shown in Figure 10.

As seen from Figure 10, for different altitudes, there are positioning anomalies, and some points do not achieve centimetre-level positioning accuracy. In contrast, some positions are more volatile in comparison. In other words, conventional UWB positioning either has large fluctuations or large deviations. For this purpose, from the aforementioned sample set, a training dataset and a testing dataset are constructed, which are fed into the BPNN and the HHO-BP neural network, which predicts the localization sites. Finally, the error analysis is performed with the actual coordinate points. To analyse the effect of the algorithm, we choose nonidentical locus points as the test set.

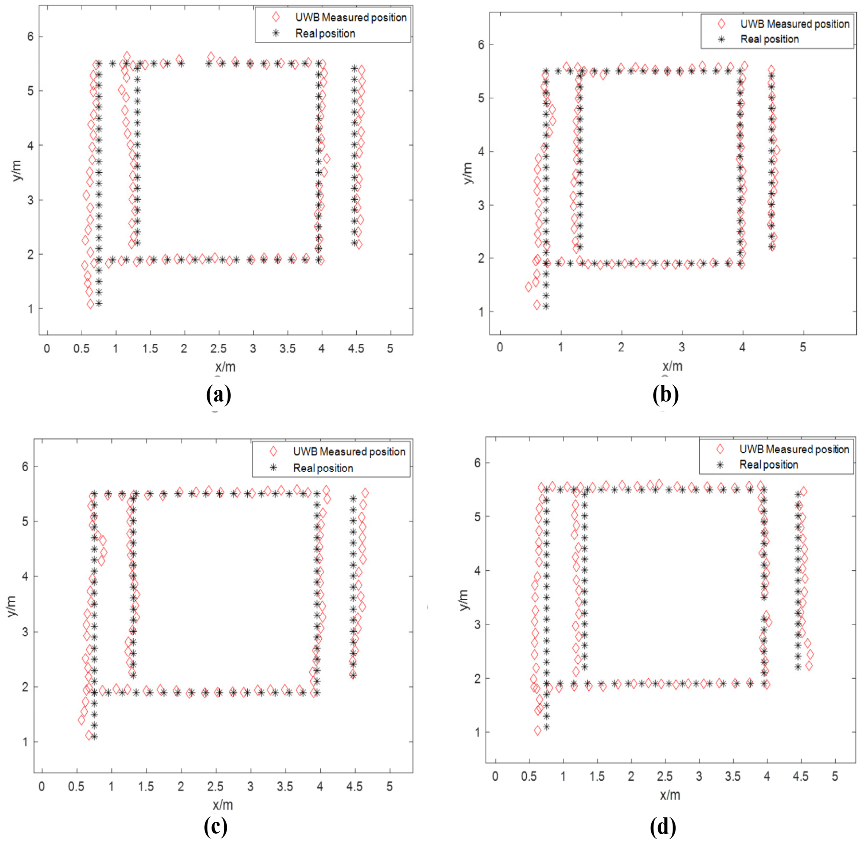
The black points in Figure 11 are the actual locations (i.e., the metric measurement data). The red points are the original UWB positioning measurement data. As the figure shows, the initial UWB positioning is not ideal, and there are significant errors.

In Figure 12, the black points are the actual coordinates (measured in metres), the blue triangles are the unoptimized BP neural network (random weight threshold) localization points, and the red circles are the HHO-BP neural network localization points.

When comparing Figures 11 and 12, the findings reveal that both the BPNN and the HHO-BP have greatly increased accuracy. To more accurately evaluate the difference in effectiveness between the BPNN and the optimized HHO-BP, each test point was compared with the actual location, and three sets of error data were obtained. The initial UWB measurement error, the BPNN prediction result error, and the HHO-BP prediction result error were all identified, as demonstrated in Figures 13 and 14.

In Figure 13, the black line segment indicates the original UWB error, the blue line segment shows the BP neural network optimization error, and the red line segment indicates the HHO-BP neural network optimization error.

The experimental findings in Figures 13 and 14 reveal that the HHO-BP algorithm has a superior optimization impact relative to the BP neural algorithm at all four heights of the LOS environment, which effectively reduces the error of UWB indoor positioning and achieves centimetre-level accuracy. At the same time, HHO-BP can effectively decrease outliers and reduce accuracy variation to a certain amount.



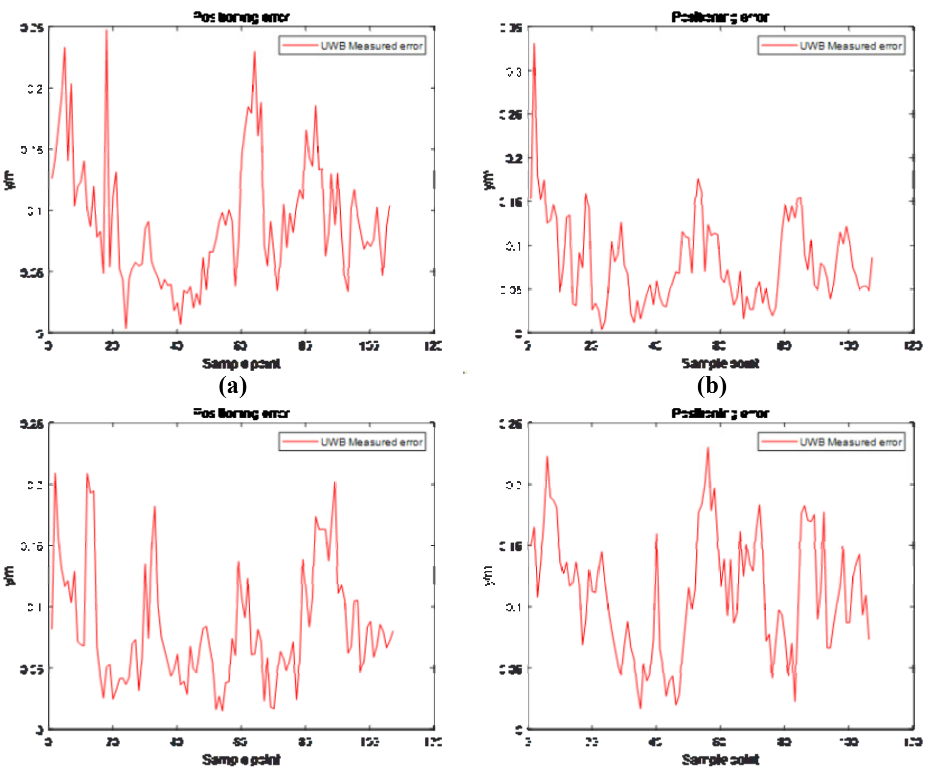
(a)

(b)

(c)

(d)

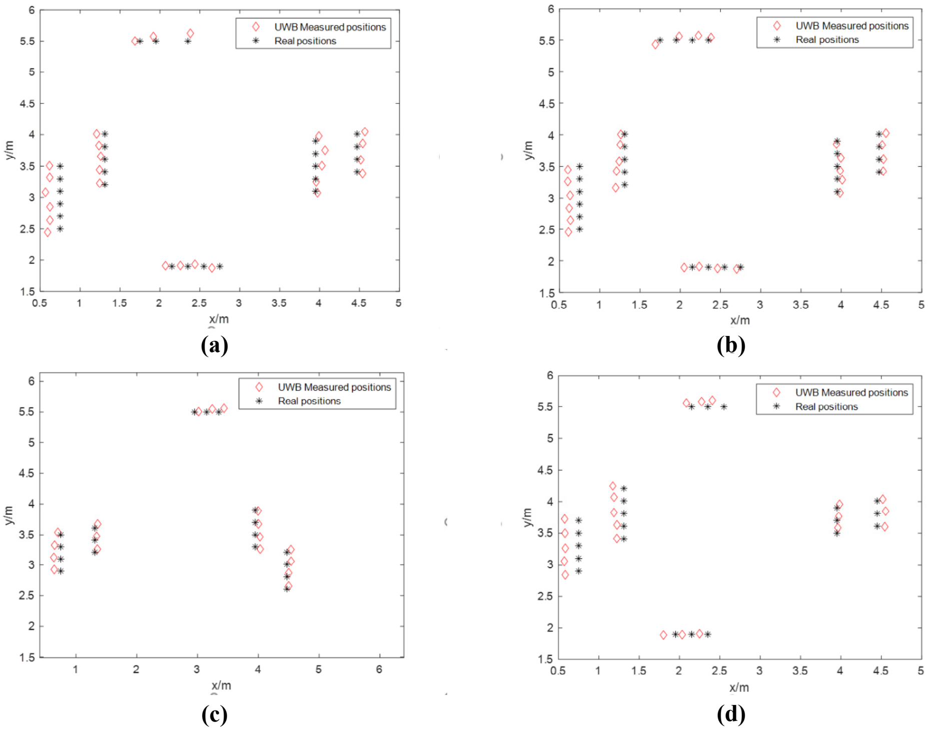
Figure 9. The LOS environment has four total altitude test positioning points and actual coordinate points. (a) 1.3 m. (b) 1.55 m. (c) 1.7 m. (d) 1.95 m.



(a) (b)

(c) (d)

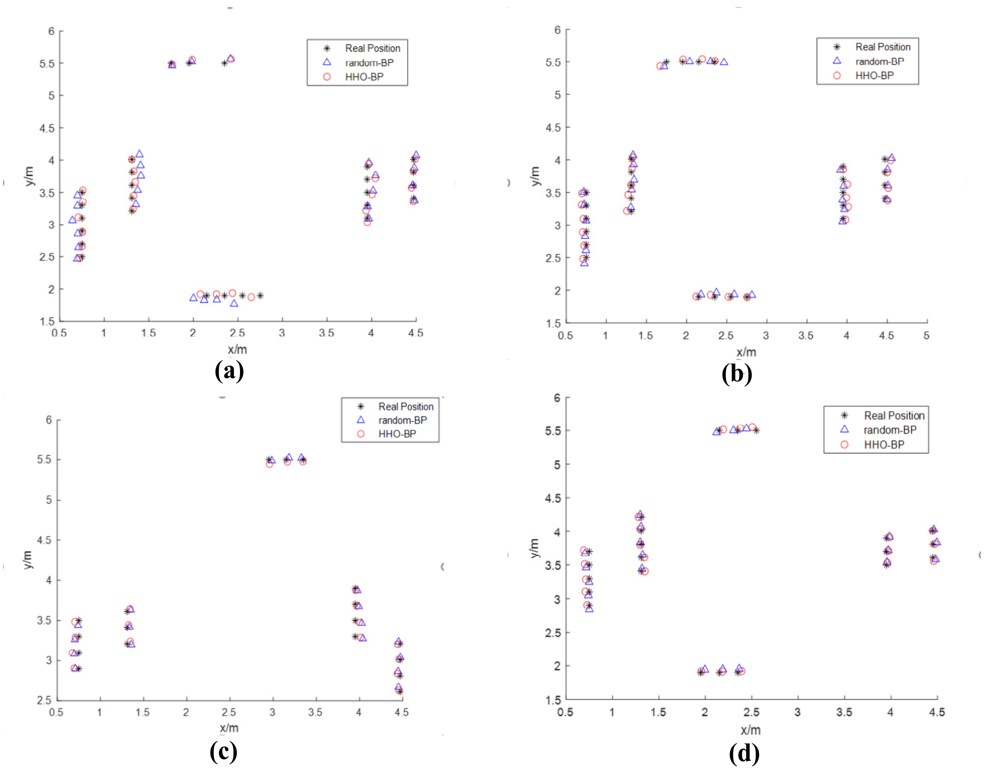
Figure 10. The original error between the actual position and the test position for each point at four different heights. (a) 1.3 m. (b) 1.55 m. (c) 1.7 m. (d) 1.95 m.



(a) (b)

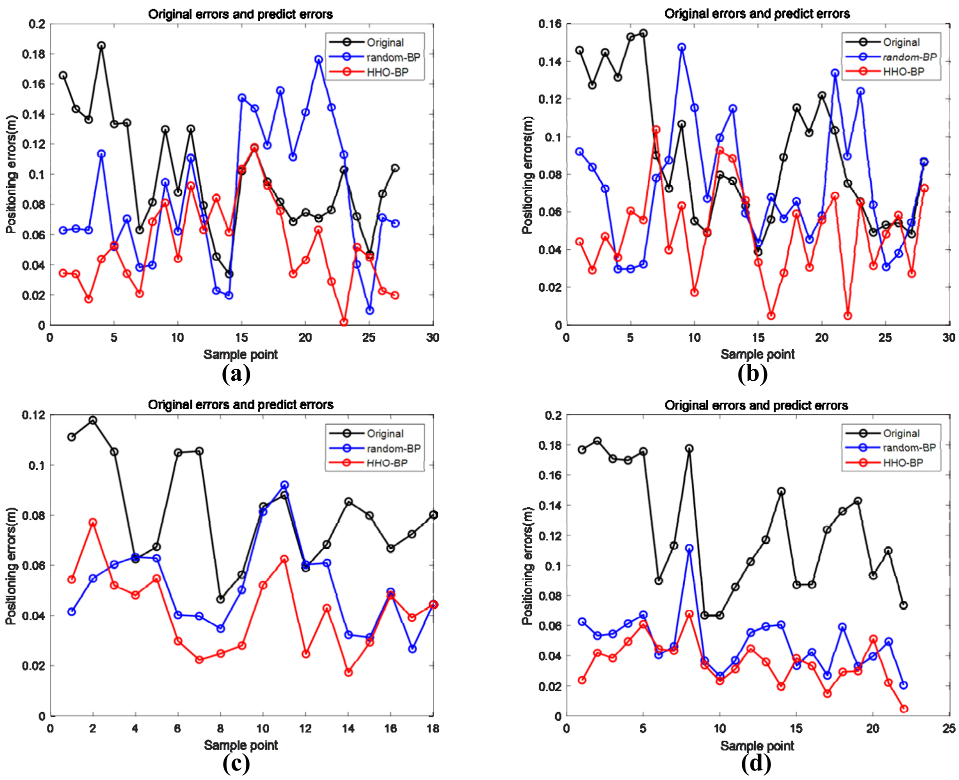
(c) (d)

Figure 11. The test set actual position and measured position. (a) 1.3 m. (b) 1.55 m. (c) 1.7 m. (d) 1.95 m.



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

Figure 12. The test set actual position and optimized points. (a) 1.3 m. (b) 1.55 m. (c) 1.7 m. (d) 1.95 m.



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

Figure 13. The test set original error and optimized error. (a) 1.3 m. (b) 1.55 m. (c) 1.7 m. (d) 1.95 m.

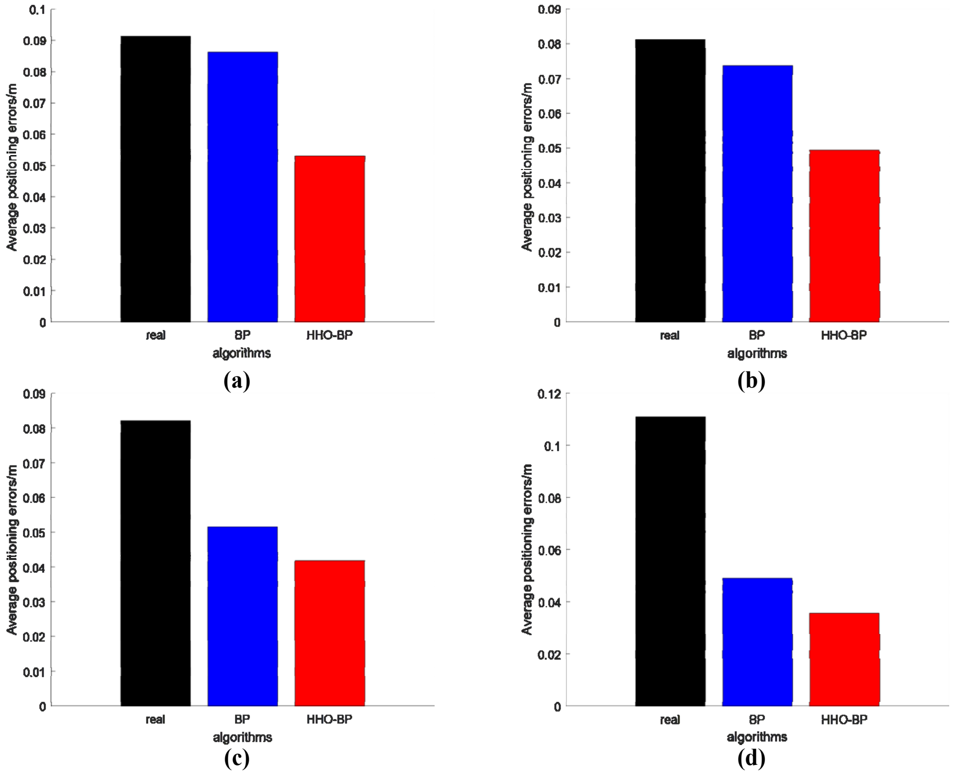


Figure 14. Test set average error and optimized error. (a) 1.3 m. (b) 1.55 m. (c) 1.7 m. (d) 1.95 m.

4.2.2. NLOS environment experiment

In this environment, to achieve the NLOS environmental conditions, four UWB anchors were adjusted to a height of 1.4 m, and the test points were located around the table with the table positioned as an occluder. The localization coordinates of the UWB tag heights of 0.24, 0.4, 0.55 and 0.67 m were measured separately as the sample set for this environment. From the aforementioned sample set, a training dataset and a testing dataset were constructed.

From Figure 15, from sample sets of different heights, it can be found that in the NLOS environment, the outliers of sample points increase significantly. Because of the shielding effect of the table, the signal's arrival time increases, such that the measured coordinates deviate from the shielding object. As the height of the UWB tag decreases gradually, the complexity of the NLOS environment increases, and the location deviation becomes more chaotic, causing the optimization task of this environment to be more difficult.

Figure 16 shows the error curves of the sample set for each altitude measurement versus the actual coordinates, indicating a significant increase in error in the NLOS environment, along with greater volatility. Figure 16 shows that UWB positioning has a large error in the NLOS environment, with some test points even showing an error of more than 1.5 m. Therefore, optimization for this situation is essential.

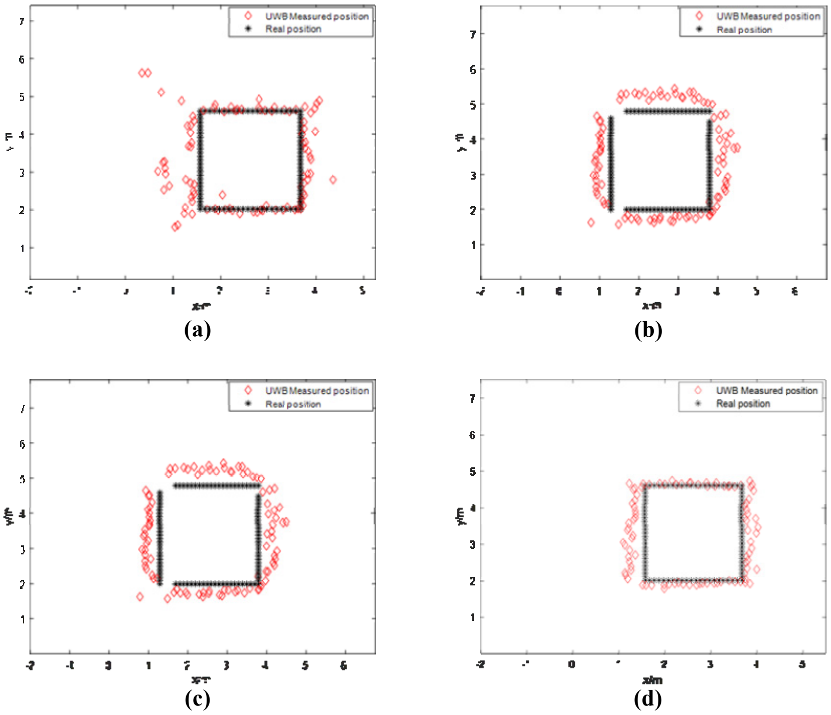


Figure 15. NLOS environment with four total altitude test positioning points and actual coordinate points. (a). 0.24 m. (b). 0.4 m. (c). 0.55 m. (d). 0.67 m.

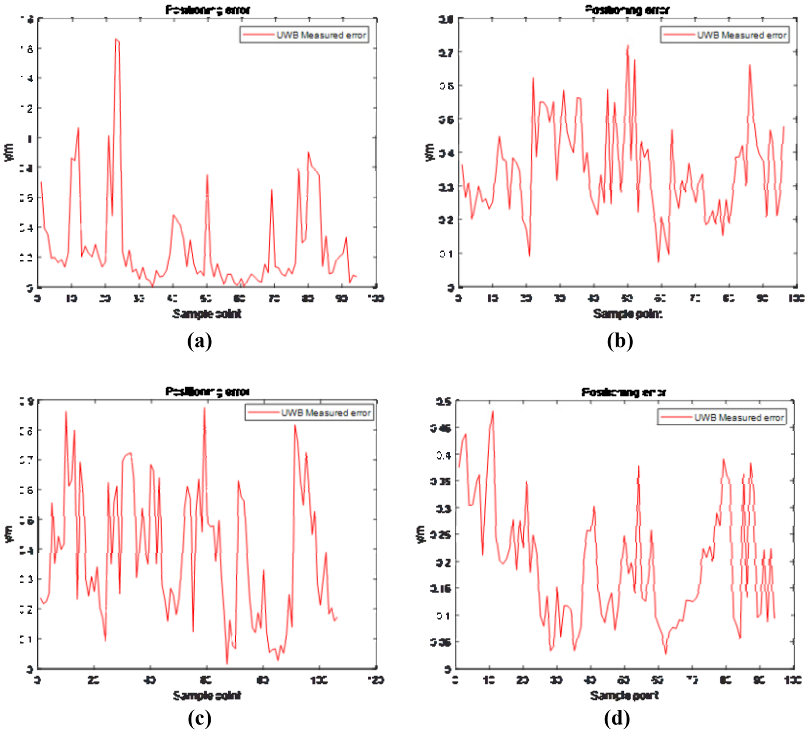


Figure 16. The original error between the actual position and the test position for each point at four different heights. (a). 0.24 m. (b). 0.4 m. (c). 0.55 m. (d). 0.67 m.

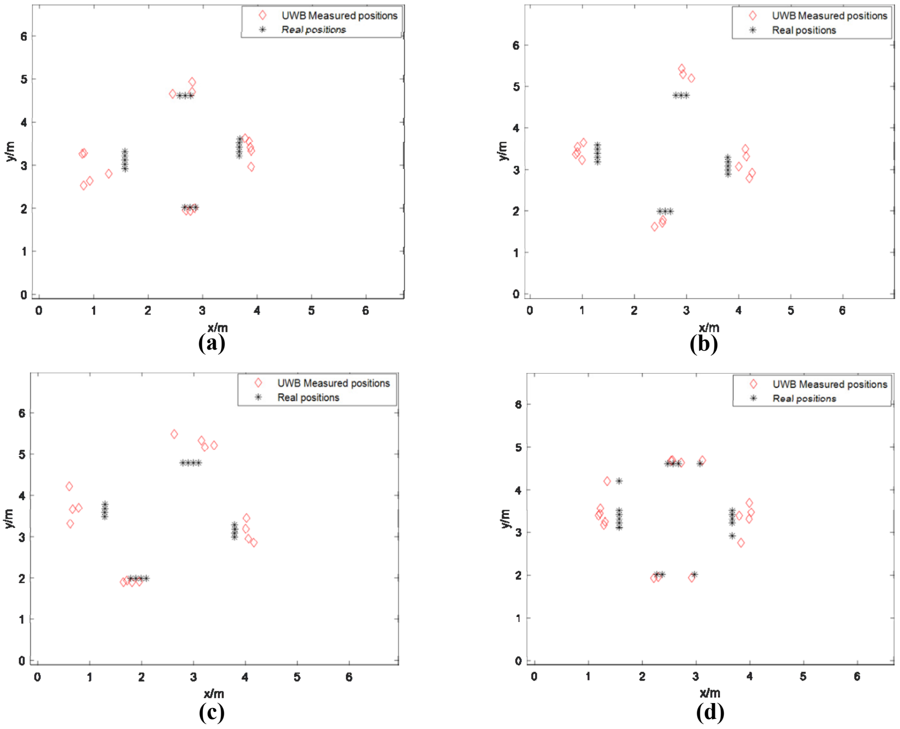


Figure 17. The training set actual position and measured position. (a). 0.24 m. (b). 0.4 m. (c). 0.55 m. (d). 0.67 m.

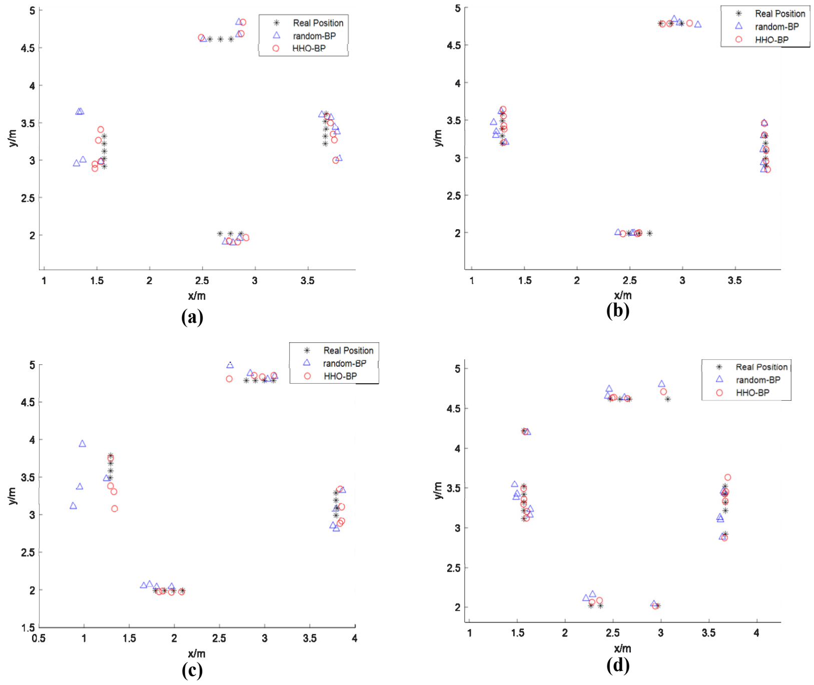
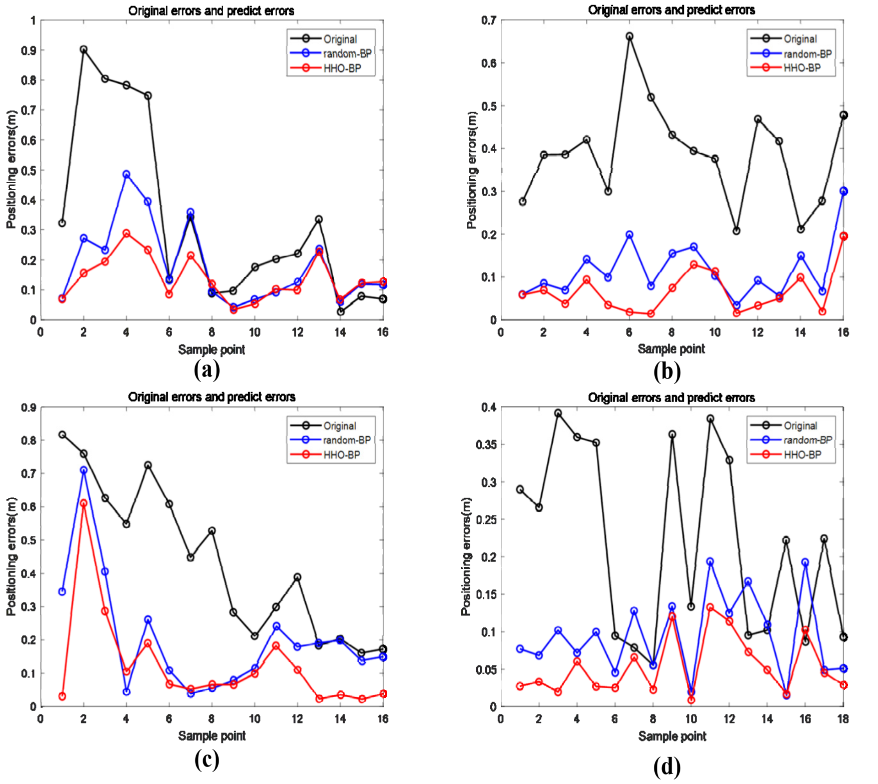


Figure 18. The test set actual position and measured position. (a). 0.24 m. (b). 0.4 m. (c). 0.55 m. (d). 0.67 m.



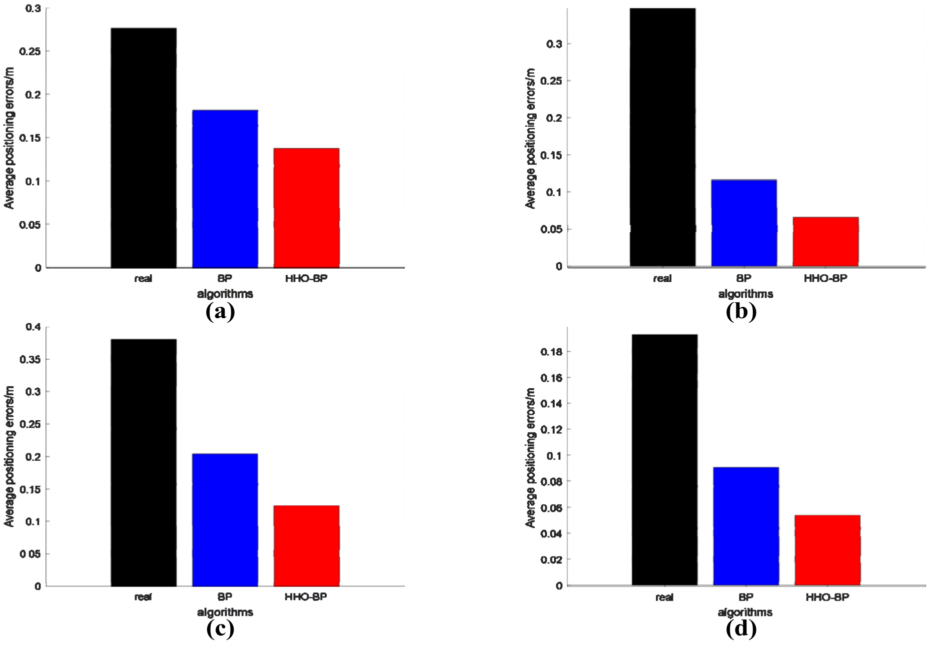
(a)

(b)

(c)

(d)

Figure 19. The test set original error and optimized error. (a). 0.24 m. (b). 0.4 m. (c). 0.55 m. (d). 0.67 m.



(a)

(b)

(c)

(d)

Figure 20. The test set average error and optimized error. (a). 0.24 m. (b). 0.4 m. (c). 0.55 m. (d). 0.67 m.

Comparing Figures 17 and 18, the results reveal that under the NLOS setting, the accuracy of the BPNN and the HHO-BP has improved significantly. To more accurately evaluate the difference in effectiveness between the BPNN and the optimized HHO-BP, the UWB measurement error is compared to the error of each point in the testing dataset, as shown in Figures 19 and 20.

Figures 19 and 20 show that HHO-BP effectively reduces errors and improves positioning accuracy at all altitudes in the complex NLOS environment.

Table 1. Results of the algorithm.

		BP			HHO-BP		
Environment	H	Max E	Min E	ME	Max E	Min E	ME
LOS	130	17.63	0.97	8.63	11.78	0.21	5.31
	155	14.77	2.99	7.39	10.38	0.50	4.94
	170	9.22	2.68	5.15	7.72	1.74	4.18
	195	11.14	4.77	4.89	6.78	2.05	3.56
NLOS	24	48.60	4.25	18.19	28.82	3.46	13.75
	40	30.12	3.45	11.65	19.57	1.47	6.65
	55	71.08	3.90	20.41	61.10	2.24	12.45
	67	19.35	1.53	9.04	13.25	0.87	5.41

Note: H: Height, Max E: Maximum Error, Min E: Minimum Error, ME: Mean Error.

Table 1 indicates the maximum errors, minimum errors and mean errors of all test data. These details are obtained from the graph above. Here, the mean error is counted separately for each height result. According to experimental data, the optimization of typical BP neural networks is not ideal, especially in nonvisual range environments, and rarely has centimetre-level accuracy.

Table 2. Results of promoting.

		UWB	BP		HHO-BP	
Environment	H	ME	ME	O P	ME	O P
LOS	130	9.15	8.63	5.70	5.31	41.99
	155	8.13	7.39	9.08	4.94	39.23
	170	8.22	5.15	37.29	4.18	49.07
	195	11.11	4.89	55.91	3.56	67.95
NLOS	24	27.68	18.19	34.28	13.75	50.33
	40	34.80	11.65	66.52	6.65	80.90
	55	38.05	20.41	46.36	12.45	67.28
	67	19.30	9.04	53.15	5.41	71.97

Note: H: Height (cm), ME: Mean Error (cm); O P: Optimization percentage (%).

Table 2 uses the statistical mean error for each height and further calculates the optimization percentages of the conventional BPNN and the HHO-BP algorithm. The mathematical formula is as follows in Eq (18):

$$
\begin{aligned}
Y_{HP} &= (Y_U - Y_H) \div Y_U \times 100\% \\
Y_{BP} &= (Y_U - Y_B) \div Y_U \times 100\%
\end{aligned}
\tag{18}
$$

where Y_{HP} is the optimization percentage of the HHO-BP algorithm, Y_{BP} is the optimization percentage of the conventional BPNN, Y_U is the mean error of the UWB measurement, Y_H is the mean error of the HHO-BP algorithm, and Y_B is the mean error of the conventional BPNN.

According to Table 2, the HHO-BP algorithm outperforms the optimization results of the conventional BPNN at every height, both for the LOS and NLOS distances. In addition, for the HHO-BP algorithm in non-line-of-sight distance conditions, compared with UWB measurement data, the degree of optimization is more than 50%, and the effect is better than the line-of-sight distance environment. For 130 cm data, the conventional BP neural network is only optimized by 5.70%, while HHO-BP performs 36.29% better than the conventional BP neural network.

Table 3. Total data comparison.

	UWB	BP		HHO-BP		HHO-BP greater than BP
Environment	MTE	MTE	O P	MTE	O P	O P
LOS	9.15	6.52	26.99	4.50	49.56	22.57
NLOS	29.96	14.82	50.08	9.56	67.62	17.54

Note: MTE: Mean total error (cm), O P: Optimization percentage (%).

Table 3 shows the total mean errors for LOS and NLOS and the optimization percentages. Here, the mean errors in Tables 1 and 2 are averaged to obtain the total mean error of the corresponding environment. According to the data in Table 3, the HHO-BP algorithm achieves centimetre-level positioning accuracy in both LOS and NLOS environments. Under line-of-sight conditions, the HHO-BP algorithm improves the localization accuracy by 22.57% compared to the conventional BP algorithm; under non-line-of-sight conditions, the HHO-BP algorithm improves the localization accuracy by 17.54% in comparison to the conventional BPNN.

5. Conclusions

In this study, an intelligent group algorithm (HHO) combined with a BP neural network is proposed to calibrate UWB indoor locations. First, the original and UWB coordinates are collected to obtain the actual errors. Next, BPNN training is used to generate the relevant weights and thresholds. Then, the HHO method optimizes the weights and thresholds of the BPNN until the minimum error is achieved. Finally, a new BPNN is constructed to calibrate the UWB measurement coordinates. The original BPNN and HHO-BP errors are compared. The experimental results show that, to some extent, BPNN calibration of the UWB indoor location will fall into the optimal local solution and that the HHO-BP method has higher accuracy. Finally, the proposed HHO-BP method is applied to indoor NLOS environments for calibrating indoor positioning environments with larger errors. The findings demonstrate that the HHO-BP algorithm's overall positioning error in an LOS environment is approximately 4.5 cm, which is 22.57% higher than the conventional BPNN, and for 130 cm data, the maximum improvement is 36.29% compared to the conventional BPNN. In an NLOS scenario, the HHO-BP algorithm has a positioning error of approximately 9.6 cm, which is 17.54% better than the conventional BPNN method. Experiments have confirmed the accuracy and efficacy of the suggested indoor positioning technique, indicating that it has greater calibration precision and stability, as well as strong future application possibilities.

Authors' contributions

CXH conceived the experimental study methodology, participated in the design and drafted the manuscript. FMS and LZY conducted the research and helped refine the experimental ideas. JCC provided technical support and specific directions for the algorithms and helped revise the manuscript. LY collected the data and analysed the results. All authors read and approved the final manuscript.

Availability of data and materials

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

Acknowledgements

We are very grateful to the Robotics Research Laboratory of Shandong University of Science and Technology for providing an experimental site for our research. This work was supported by the Natural Science Research Project of Universities in Anhui Province (NO. KJ2021A0953), Natural Science Key Scientific Research Project of West Anhui University (NO. 0041021003, WXZR201903), Key Research Programs of Shandong Province (NO. 2018GGX103011) and Key Science and Technology Innovation Programs in Shandong Province (NO. 2017CXGC0919).

Conflict of interest

There are no conflicts of interest declared by the authors.

References

1. T. Wu, H. Xia, S. Liu, Y. Qiao, Probability-based indoor positioning algorithm using ibeacons, *Sensors*, **19** (2019), 5226. https://doi.org/10.3390/s19235226
2. G. Schroeer, A real-time UWB multi-channel indoor positioning system for industrial scenarios, in *2018 International Conference on Indoor Positioning and Indoor Navigation (IPIN)*, (2018), 1–5. https://doi.org/10.1109/IPIN.2018.8533792
3. Y. Zhang, L. Duan, Toward elderly care: A phase-difference-of-arrival assisted ultra-wideband positioning method in smart home, *IEEE Access*, **8** (2020), 139387–139395. https://doi.org/10.1109/ACCESS.2020.3012717
4. A. Alsudani, NLOS mitigation and ranging accuracy for building indoor positioning system in UWB using commercial radio modules, in *AIP Conference Proceedings*, **1968** (2018), 030056. https://doi.org/10.1063/1.5039243

5. J. Zhong, S. Zhao, X. Han, Y. Liu, K. Guo, Research on indoor and outdoor positioning system for special population, in *IOP Conference Series: Materials Science and Engineering*, **719** (2020). https://doi.org/10.1088/1757-899X/719/1/012055
6. A. Alarifi, A. Al-Salman, M. Alsaleh, A. Alnafessah, S. Al-Hadhrami, M. A. Al-Ammar, et al., Ultra -wideband indoor positioning technologies: Analysis and recent advances, *Sensors*, **16** (2016), 707. https://doi.org/10.3390/s16050707
7. Y. S. Li, F. S. Ning, Low-cost indoor positioning application based on map assistance and mobile phone sensors, *Sensors*, **18** (2018), 4285. https://doi.org/10.3390/s18124285
8. Y. Ji, A. Yamashita, H. Asama, Indoor positioning system based on camera sensor network for mobile robot localization in indoor environments, *J. Inst. Control Rob. Syst.*, **22** (2016), 952–959. https://doi.org/10.5302/J.ICROS.2016.16.0079
9. Z. Li, L. Zhao, C. Qin, Y. Wang, WiFi/PDR integrated navigation with robustly constrained Kalman filter, *Meas. Sci. Technol.*, **31** (2020), 084002. https://doi.org/10.1088/1361-6501/ab87ea
10. X. Li, J. Wang, C. Liu, A Bluetooth/PDR integration algorithm for an indoor positioning system, *Sensors*, **15** (2015), 24862–24885. https://doi.org/10.3390/s151024862
11. L. F. Shi, Y. Wang, G. X. Liu, S. Chen, Y. L. Zhao, Y. F. Shi, A fusion algorithm of indoor positioning based on PDR and RSS fingerprint, *IEEE Sensors J.*, **18** (2018), 9691–9698. https://doi.org/10.1109/JSEN.2018.2873052
12. C. Lu, H. Uchiyama, D. Thomas, A. Shimada, R. I. Taniguchi, Indoor positioning system based on chest-mounted IMU, *Sensors*, **19** (2019), 420. https://doi.org/10.3390/s19020420
13. Y. Zhang, J. Tan, Z. Zeng, W. Liang, Y. Xia, Monocular camera and IMU integration for indoor position estimation, in *2014 36th Annual International Conference of the IEEE Engineering in Medicine and Biology Society*, (2014), 1198–1201. https://doi.org/10.1109/EMBC.2014.6943811
14. B. Yang, X. Xu, T. Zhang, Y. Li, J. Tong, An Indoor navigation system based on stereo camera and inertial sensors with points and lines, *J. Sensors*, **2018** (2018). https://doi.org/10.1155/2018/4801584
15. J. Duque Domingo, C. Cerrada, E. Valero, J. A. Cerrada, An improved indoor positioning system using RGB-D cameras and wireless networks for use in complex environments, *Sensors*, **17** (2017), 2391. https://doi.org/10.3390/s17102391
16. A. Poulose, D. S. Han, Hybrid indoor localization using IMU sensors and smartphone camera, *Sensors*, **19** (2019), 5084. https://doi.org/10.3390/s19235084
17. H. Shu, C. Song, T. Pei, L. Xu, Y. Ou, L. Zhang, et al., Queuing time prediction using WiFi positioning data in an indoor scenario, *Sensors*, **16** (2016), 1958. https://doi.org/10.3390/s16111958
18. A. Poulose, D. S. Han, Hybrid deep learning model based indoor positioning using Wi-Fi RSSI heat maps for autonomous applications, *Electronics*, **10** (2020), 2. https://doi.org/10.3390/electronics10010002
19. F. Zafari, A. Gkelias, K. K. Leung, A survey of indoor localization systems and technologies， *IEEE Commun. Surv. Tutorials*, **21** (2019), 2568–2599. https://doi.org/10.1109/COMST.2019.2911558
20. Z. Farid, R. Nordin, M. Ismail, Recent advances in wireless indoor localization techniques and system, *J. Comput. Networks Commun.*, **2013** (2013). https://doi.org/10.1155/2013/185138

21. T. Kim Geok, K. Zar Aung, M. Sandar Aung, M. Thu Soe, A. Abdaziz, C. Pao Liew, et al., Review of indoor positioning: Radio wave technology, *Appl. Sci.*, **11** (2020), 279. https://doi.org/10.3390/app11010279
22. J. Luo, L. Fan, H. Li, Indoor positioning systems based on visible light communication: State of the art, *IEEE Commun. Surv. Tutorials*, **19** (2017), 2871–2893. https://doi.org/10.1109/COMST.2017.2743228
23. P. Dabove, V. Di Pietra, M. Piras, A. A. Jabbar, S. A. Kazim,. Indoor positioning using Ultra-wide band (UWB) technologies: Positioning accuracies and sensors' performances, in *2018 IEEE/ION Position, Location and Navigation Symposium (PLANS)*, (2018), 175–184. https://doi.org/10.1109/PLANS.2018.8373379
24. A. Poulose, O. S. Eyobu, M. Kim, D. S. Han, Localization error analysis of indoor positioning system based on UWB measurements, in *2019 Eleventh International Conference on Ubiquitous and Future Networks (ICUFN)*, (2019), 84–88. https://doi.org/10.1109/ICUFN.2019.8806041
25. A. Poulose, Ž. Emeršič, O. S. Eyobu, D. S. Han, An accurate indoor user position estimator for multiple anchor UWB localization, In *2020 International Conference on Information and Communication Technology Convergence (ICTC)*, (2020), 478–482. https://doi.org/10.1109/ICTC49870.2020.9289338
26. A. Chaisang, S. Promwong, Indoor localization distance error analysis with UWB wireless propagation model using positioning method, in *2018 International Conference on Digital Arts, Media and Technology (ICDAMT)*, (2018), 254–257. https://doi.org/10.1109/ICDAMT.2018.8376534
27. K. He, Y. Zhang, Y. Zhu, W. Xia, Z. Jia, L. Shen, A hybrid indoor positioning system based on UWB and inertial navigation, in *2015 International Conference on Wireless Communications & Signal Processing (WCSP)*, (2015), 1–5. https://doi.org/10.1109/WCSP.2015.7341240
28. H. Liu, Z. Liang, D. Liu, L. N. Ma, Improved UWB indoor positioning algorithms based on BP neural network model, in *International Conference on Communications and Networking in China*, (2017), 114–124. https://doi.org/10.1007/978-3-319-78130-3_13
29. G. Zhuo, R. Xue, *UWB location algorithm based on BP neural network*, in *Intelligent and Connected Vehicles Symposium*, 2018. https://doi.org/10.4271/2018-01-1605
30. X. Li, S. Dong, H. S. Mohamed, G. Al Aqel, N. Pirhadi, Prediction of tubular T/Y-Joint SIF by GA-BP neural network, *KSCE J. Civil Eng.*, **24** (2020), 2706–2715. https://doi.org/10.1007/s12205-020-1200-1
31. H. Zhang, Y. Zhao, Y. Zhang, J. Zuo, M. Bian, J. Zhao, UWB indoor location algorithm based on improved BP neural network, in *International Conference on Electronic Information Engineering and Computer Technology (EIECT 2021)*, **12087** (2021), 232–236. https://doi.org/10.1117/12.2624738
32. A. Poulose, D. S. Han, UWB indoor localization using deep learning LSTM networks, *Appl. Sci.*, **10** (2020), 6290. https://doi.org/10.3390/app10186290
33. S. Xing, H. Zhang, X. Liang, T. A. Gulliver, A 60 GHz impulse radio positioning algorithm based on a BP neural network, in *2017 IEEE Pacific Rim Conference on Communications, Computers and Signal Processing (PACRIM)*, (2017), 1–5. https://doi.org/10.1109/PACRIM.2017.8121879
34. Z. K. Lian, F. Yuan, W. Qi, Improved K-means clustering BP neural network UWB indoor location method, *Mod. Comput*, **2017** (2017). https://doi.org/10.3969/j.issn.1007-1423.2017.21.003

35. J. Jin, Y. Zhang, BP neural network indoor localization algorithm based on visible light communication, *Semicond. Optoelectron.*, **4** (2019). https://doi.org/10.16818/j.issn1001-5868.2019.04.029
36. F. Huang, D. Liu, T. An, J. Cao, Port container throughput forecast based on ABC optimized BP neural network, in *IOP Conference Series: Earth and Environmental Science*, **571** (2020), 012068. https://doi.org/10.1088/1755-1315/571/1/012068
37. U. B. Tayab, A. Zia, F. Yang, J. Lu, M. Kashif, Short-term load forecasting for microgrid energy management system using hybrid HHO-FNN model with best-basis stationary wavelet packet transform, *Energy*, **203** (2020), 117857. https://doi.org/10.1016/j.energy.2020.117857
38. K. Yang, M. Liu, Y. Xie, X. Zhang, W. Wang, S. Gou, et al., Research on UWB/IMU location fusion algorithm based on GA-BP neural network, in *2021 40th Chinese Control Conference (CCC)*, (2021), 8111–8116. https://doi.org/10.23919/CCC52363.2021.9549463
39. N. Li, C. Shen, K. Zhang, X. Huang, The TDOA algorithm based on BP neural network optimized by cuckoo search, in *2019 International Conference on Robots & Intelligent System (ICRIS)*, (2019), 539–542. https://doi.org/10.1109/ICRIS.2019.00138
40. D. N. Hama Rashid, T. A. Rashid, S. Mirjalili, ANA: Ant nesting algorithm for optimizing real-world problems, *Mathematics*, **9** (2021), 3111. https://doi.org/10.3390/math9233111
41. C. M. Rahman, T. A. Rashid, A new evolutionary algorithm: Learner performance based behavior algorithm, *Egypt. Inf. J.*, **22** (2021), 213–223. https://doi.org/10.1016/j.eij.2020.08.003
42. S. Abdulhameed, T. A. Rashid, Child drawing development optimization algorithm based on child's cognitive development, *Arabian J. Sci. Eng.*, **47** (2022), 1337–1351. https://doi.org/10.1007/s13369-021-05928-6
43. J. M. Abdullah, T. Ahmed, Fitness dependent optimizer: Inspired by the bee swarming reproductive process, *IEEE Access*, **7** (2019), 43473–43486. https://doi.org/10.1109/ACCESS.2019.2907012
44. A. S. Shamsaldin, T. A. Rashid, R. A. Al-Rashid Agha, N. K. Al-Salihi, M. Mohammadi, Donkey and smuggler optimization algorithm: A collaborative working approach to path finding, *J. Comput. Des. Eng.*, **6** (2019), 562–583. https://doi.org/10.1016/j.jcde.2019.04.004
45. C. Wang, F. Wu, Z. Shi, D. Zhang, Indoor positioning technique by combining RFID and particle swarm optimization-based back propagation neural network, *Optik*, **127** (2016), 6839–6849. https://doi.org/10.1016/j.ijleo.2016.04.123
46. Y. Li, J. Liu, UWB indoor localization system based on IA-BP neural network, *Electron. Meas. Technol.*, **4** (2019).
47. Z. Yu, X. Shi, J. Zhou, X. Chen, X. Qiu, Effective assessment of blast-induced ground vibration using an optimized random forest model based on a Harris hawks optimization algorithm, *Appl. Sci.*, **10** (2020), 1403. https://doi.org/10.3390/app10041403
48. D. H. Wolpert, W. G. Macready, No free lunch theorems for optimization, *IEEE Trans. Evol. Comput.*, **1** (1997), 67–82. https://doi.org/10.1109/4235.585893
49. K. Cui, X. Jing, Research on prediction model of geotechnical parameters based on BP neural network, *Neural Comput. Appl.*, **31** (2019), 8205–8215. https://doi.org/10.1007/s00521-018-3902-6
50. S. Ding, C. Su, J. Yu, An optimizing BP neural network algorithm based on genetic algorithm, *Artif. Intell. Rev.*, **36** (2011), 153–162. https://doi.org/10.1007/s10462-011-9208-z

51. A. Abbasi, B. Firouzi, P. Sendur, On the application of Harris hawks optimization (HHO) algorithm to the design of microchannel heat sinks, *Eng. Comput.*, **37** (2021), 1409–1428. https://doi.org/10.1007/s00366-019-00892-0
52. A. A. Heidari, S. Mirjalili, H. Faris, I. Aljarah, M. Mafarja, H. Chen, Harris hawks optimization: Algorithm and applications, *Future Gener. Comput. Syst.*, **97** (2019), 849–872. https://doi.org/10.1016/j.future.2019.02.028
53. S. Wang, Z. Wu, A. Lim, Denoising, outlier/dropout correction, and sensor selection in range-based positioning, *IEEE Trans. Instrum. Meas.*, **70** (2021), 1–13. https://doi.org/10.1109/TIM.2021.3078537