IEEE Access
Multidisciplinary : Rapid Review : Open Access Journal

Received December 3, 2020, accepted December 15, 2020, date of publication December 17, 2020, date of current version January 4, 2021.

Digital Object Identifier 10.1109/ACCESS.2020.3045555

CWDV-Hop: A Hybrid Localization Algorithm With Distance-Weight DV-Hop and CSO for Wireless Sensor Networks

JIAXING CHEN[1], WEI ZHANG[2], ZHIHUA LIU[3], RUI WANG[4], (Member, IEEE), AND SHUJING ZHANG[1]

[1]College of Engineering, Hebei Normal University, Shijiazhuang 050024, China
[2]College of Electronic and Information Engineering, Hebei University of Technology, Tianjin 300400, China
[3]College of Computer and Cyber Security, Hebei Normal University, Shijiazhuang 050024, China
[4]College of Opto-Electronic Information Science and Technology, Yantai University, Yantai 264000, China

Corresponding author: Shujing Zhang (shujingzhang@hebtu.edu.cn)

This work was supported in part by the National Natural Science Foundation of China under Grant 61701165, Grant 61771181, and Grant 61501168; in part by the Natural Science Foundation of Hebei Province under Grant F2016205182; and in part by the Natural Science Foundation of Shandong Province under Grant ZR2017BF040.

ABSTRACT Localization problem is a hot topic in the field of wireless sensor networks. The distance vector-hop (DV-Hop) is a typical range-free localization algorithm which is widely used in many applications owing to its advantages such as simplicity, feasibility, low cost, and less hardware requirements. However, its localization error is relatively large. In this article, a novel approach designated as CWDV-Hop was presented to improve the localization accuracy. In the first step of CWDV-Hop, the distance-weighted hop distance was calculated by introducing the new defined distance-weighted factor and single-node hop distance, which can effectively reduce the impact of curved path on the estimation of average hop distance. Furthermore, the locations of unknown nodes are obtained by utilizing two-dimensional hyperbolic scheme. At last, Chicken Swarm Optimization (CSO), which is a kind of nature inspired algorithm, is introduced to optimize the locations of unknown nodes. In comparison with the traditional DV-Hop approach, the localization accuracy of CWDV-Hop can be raised by 53.6%, 39.5% and 53.1% for the square, X-shaped and O-shaped random distribution environment respectively, with the time complexity slightly gained. It demonstrates the effectiveness of CWDV-Hop not only in the isotropy network, but also in the anisotropy network with holes or other uneven distributions.

INDEX TERMS Wireless sensor networks, localization, DV-Hop, two-dimensional hyperbola, CSO.

I. INTRODUCTION

Wireless sensor network (WSN) is a popular research area which has attracted considerable attention recently. It composes of numerous randomly distributed nodes with built-in sensors which can collect various data from the limited surroundings [1], [2]. Due to the diverse function and low energy consumption, WSNs have been utilized in many fields such as military, civil and scientific research [3], [4]. For most scenarios, the combination of location data and sensor messages is momentous. The sensors' location data are conducted to calculate network consumption and attain route management etc. Therefore, localization proves to be a significant research direction in WSNs [5]–[7]. The characteristics of numerous sensor nodes, limited power, random distribution and complex communication environment, put forward higher demands for localization algorithms of WSNs.

Generally speaking, the localization algorithms are classified into two categories, namely range-based and range-free [8]. The formers take into account the length or angle messages among the target and anchor nodes. As the implementation of positioning is strongly dependent upon the transmission of radio signals, they are vulnerable to external environment (multipath and noise) and need additional hardware support. Consequently, range-based technology is not suitable for large-scale WSN. The commonly used range-based localization methods include Received Signal Strength Indicator (RSSI) [9]–[11], Time of Arrival (TOA) [12], [13],

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

Time Difference of Arrival (TDOA) [14], [15] and Angle of Arrival (AOA) [16], [17]. Instead, range-free algorithms are utilized to compute the coordinates of sensor nodes with the help of communication or connectivity message, favored because of its low cost and strong network adaptability. Although their localization accuracy is relatively low, it can basically meet realistic demands. Distance Vector-Hop (DV-Hop) [18], [19], Approximate Point-In-Triangulation Test (APIT) [20], Centroid algorithm [21] are typical representatives of range-free methods.

The most well-known range-free distributed localization technology is the DV-hop algorithm which holds many decided advantages of simplicity, feasibility, low cost and less hardware requirements. However, it still provides coarse position estimation. Achieving better accuracy with some improved algorithms remains a matter of research. Therefore, a novel localization method designated as CWDV-Hop is put forward on the base of distance-weighted DV-Hop and Chicken Swarm Optimization (CSO) in this article. The major contributions are summarized below:

- A novel DV-Hop-based localization approach, named WDV-Hop, is presented to work as an underground positioning mechanism. Compared to the hop-weight-based DV-Hop methods, WDV-Hop develops a distance-weight hop distance by introducing the new defined distance-weight factor and single-node hop distance. This makes the influence of distance among different anchors taken into consideration and can largely reduce the impact of curved path on average hop distance. Instead of using Least Square (LS) algorithm in the traditional DV-Hop, WDV-Hop applies the two-dimensional hyperbola approach to estimate the final locations of unknown nodes.
- In order to reduce the localization errors of unknown nodes, a CSO-based nature inspired algorithm is utilized to find the best coordinates of target nodes. On one hand, the minimal model of CSO is built by utilizing the distance information among anchor and unknown nodes. On the other hand, Chicken individuals look for the best locations of target nodes by several iteration and update. CSO-based localization algorithm behaves intelligently and efficiently for the optimization problems.
- By combining WDV-Hop with CSO, we also provide a new efficient localization solution called CWDV-Hop. Unlike other similar localization algorithms, the proposed algorithm overcomes the short coming of the traditional DV-Hop algorithms which are only applicable to isotropic network, therefore has a strong adaptability to the complex deployment environment. CWDV-Hop is compared to other competitive algorithms under different distribution scenarios. Simulation results show that the proposed algorithm has improved by 44.5% $\sim$ 53.6% with the variation of correlative parameters.

The following content is arranged below. Section 2 presents the literature review of associated researches. Section 3 illustrates the several system models. Section 4 describes the proposed CWDV-Hop algorithm. Before conclusion, simulation results and analysis are presented in section 5.

II. RELATED WORK

Generally, node localization consists of two steps: one is distance estimation and the other is coordinate estimation. When using these two steps to determine the coordinate of unknown nodes, error is inevitable, and the smaller the error, the higher the location accuracy. Recently, scholars use the weight model and nature inspired methods to optimize the DV-hop algorithm to achieve a certain degree of accuracy improvement.

A. RELATED WORK ON WEIGHT-BASED VARIANTS OF DV-HOP

DV-Hop estimates the distance between the anchor node and the unknown node by multiplying the minimal hop count and the average distance per hop. Scholars consider that the essential reason of poor distance estimation is the inaccuracy of average hop distance. Therefore, most of the weighted approaches are proposed to improve the precision of average hop distance in the process of distance estimation [22]–[24]. In [22] and [23], the distance error between anchors and the hop counts from the unknown nodes to the anchors are both considered in the weight model which is then used to modify the average hop distance. [22] establishes the weight function by the sum of hop count factor and distance error factor, while [23] by the product form. The limitation of [22] is that the weight also relies on an importance coefficient which must be adjusted. In [24], a weighted DV-hop algorithm is presented by adding a correction parameter to the average hop distance. The correction factor is based on the mean hop distance error. However, it also dependents on a balance coefficient which highly depends on the network environment. Similarly, [25] adopts the inverse distance weighting (IDW) correction method to obtain more accurate average hop distance. IDW uses an inverse distance power value for the adjustments. The power of distance is set as 2, which does not have solid reason.

Alternatively, the weight models are introduced in the process of coordinate estimation. In [26], the Locally Weighted Linear Regression (LWLR) approach is applied to optimize the least square estimation of DV-hop. LWLR uses a Gaussian kernel to assign a weight for each neighboring anchor. Though the localization accuracy is largely improved, the relationship of linearity or nonlinearity between distances and hop counts must be determined for all anchor nodes. [27] replaces the maximum likelihood estimation of DV-hop with a weighted centroid approach. The weights determine the relative importance of each anchor on the location of unknown nodes by measuring the hop counts. However, this method does not perform well in the anisotropic network.

In summary, the weight models above are all built on the conventional wisdom which thinks the weights will decrease with the hop counts or/and distances increase. Although these algorithms improve the positional precision of the sensor nodes, they do not provide a convincing proof of the relationship among the hop counts, distances and weights. Therefore, the theoretical foundations are not solid. To address this issue, we construct a clear demonstration and find more accurate position estimation in this paper.

B. RELATED WORK ON NATURE INSPIRED ALGORITHMS-BASED VARIANTS OF DV-HOP

Node localization is formulated as an unconstrained NP-hard optimization problem. Recently, many nature inspired algorithms are applied to solve this problem. Fig. 1 describes the general classification of nature inspired algorithms, which are divided into two categories: individual and swarm.

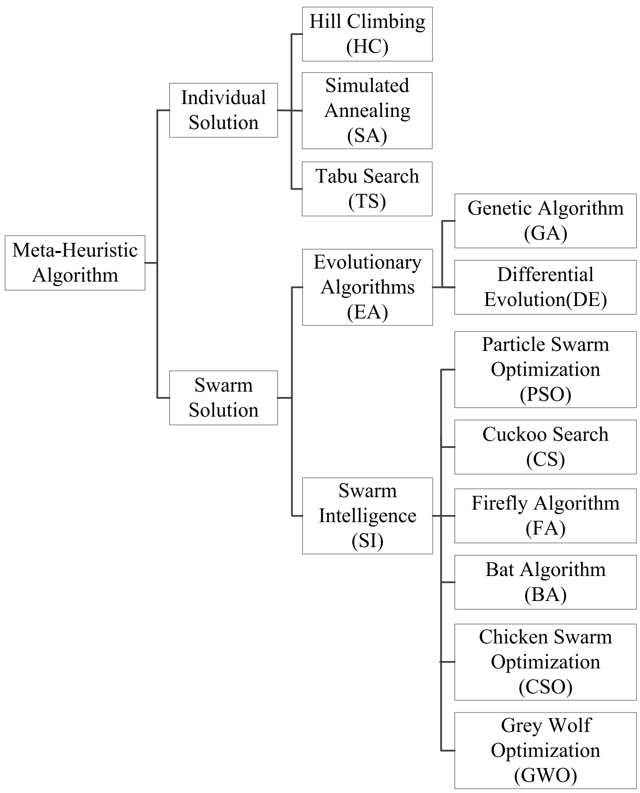

FIGURE 1. Classification of nature inspired algorithms.

Recently, many swarm intelligent algorithms are applied to find an optimal solution in the localization area. The swarm optimization methods for precision improvement of DV-Hop algorithm can be summarized from the following three aspects:

I. The first one tries to reduce the distance estimation error by optimizing the average hop distance.
II. The second is to optimize the estimated coordinate from DV-Hop to find the best result of unknown nodes.
III. The third one manages to estimate the coordinates of unknown nodes directly instead of the multilateration method.

Table 1 lists the frequently-used swarm optimization algorithms for DV-Hop during recent three years. In order to describe the above three aspects for short, we use I, II, and III to represent them in Table 1.

TABLE 1. Swarm optimization-based DV-Hop.

Algorithm	Description	Reference
GA	II	[8], [28]
	III	[29]
DE	III	[30] [31]
PSO	I	[32]
	II	[18], [33]–[35]
	III	[36], [37]
CS	II	[38]
	III	[39]
BA	I	[40]
	II	[41], [42]
GWO	I	[5]

In [8], [28], [29], Genetic Algorithm (GA) is applied to increase the localization accuracy of DV-Hop by building the objective function and fitness function. [8] utilizes GA to optimize the estimated coordinates of unknown nodes, which are calculated by the improved DV-Hop algorithm. In [28], Cai *et al*. propose a three-dimensional DV-hop localization algorithm by introducing the Non-Dominated Sorting Genetic Algorithm II (NSGA-II). In addition, this paper uses the multi-objective positioning model rather than the traditional single-objective model to find the best solution. Being different from [8] and [28], [29] computes the coordinates of unknown nodes directly with the help of GA after analyzing the correlation between connectivity and position.

With the merits of faster convergence, fewer parameters and more robustness, Differential Evolution (DE) [30], [31] also can be used to solve the localization problem. In [30], the object function of the minimized optimization problem is established on the weighted squared errors of estimated distance and DE is applied to obtain the estimated location of unknown nodes instead of the multilateration method. With the improvements of mutation operation and crossover operation of the basic DE algorithm, Han *et al*. propose an improved DE algorithm which is applied to obtain the global optimal solution corresponding to the estimated location of the unknown node [31]. Although the DE-based algorithms improve the positional precision of the unknown nodes, they also induce vast time overhead and energy consumption.

Besides, Particle Swarm Optimization (PSO) is widely lead-in to heighten the localization precision of sensor nodes. [32] uses PSO as the learning mechanism of quantum neural network (QNN) and proposes a PSO-QNN model which is applied to optimize the average hop distance of DV-HOP. In [18], [33]–[35], PSO tries to optimize the estimated coordinate of unknown nodes obtained by DV-Hop. Singh and Sharma [33] firstly use hyperbola method instead of Least Square (LS) approach to obtain the estimated location of unknown node and then optimize that location with the help of PSO. Shi *et al*. [34] modify the average hop distance of relevant nodes by a path matching algorithm and optimize the initial location of each unknown node with a modified PSO algorithm. In [35], an improved PSO is applied to settle the matter of high localization deviation aroused by initial value sensitivity of LS approach. PSO in [18] is used twice that the first time is introduced to build the dynamic set for selecting appropriate anchors by the binary particle swarm optimization (BPSO) algorithm and the second is used to optimize the coordinates of unknown nodes with the continuous PSO method. Additionally, [36] and [37] replace the multitrilateration algorithm of DV-Hop with PSO. [36] designs a relative hop angle connectivity-based DV-Hop localization scheme, in which PSO is applied to search for the location of unknown nodes given the fitness function. In [37], an improved PSO is presented with Monte Carlo localization boxed (MCB) for three-dimensional mobile wireless sensor networks.

In [38], [39], Cuckoo Search (CS)-based DV-Hop method is designed for improvement of the precision performance. [38] incorporates a new oriented cuckoo search algorithm with Lévy distribution and Cauchy distribution (OCS-LC) into the methodology of DV-Hop algorithm to improve the predicted precise. The correction factor is applied to modify the hop count of DV-Hop and then the node coordinates are calculated by CS instead of the maximum likelihood estimation method in [39].

The Bat algorithm (BA) can also be introduced to optimize the average hop distance [40] or the estimated coordinate of unknown nodes [41], [42].

In [5], the average hop distance is refined by Grey Wolf Optimization (GWO) for randomly deployed 2D and 3D WSN.

In general, many nature inspired algorithms have been successfully applied to optimize the DV-Hop localization problem. However, achieving reasonable accuracy with faster convergence time remains a matter of research (see Fig.17). To solve this issue, we will find a more suitable nature inspired algorithm for the localization optimization problem.

C. RELATED WORK ON DV-HOP DERIVATIONS IN ISOTROPY AND ANISOTROPY NETWORKS

As is well known, DV-Hop-based localization algorithms demonstrate acceptable performance in isotropic networks in which sensor nodes distribute evenly. However, these algorithms are easily affected by network topology, causing a significant decrease in positioning accuracy for the anisotropic networks. Taking Fig. 2 as an example, the blue dashed lines denote the geometric distances between sensor nodes A and B, while the green solid lines indicate the hop distances of the shortest path between them. P1 in the isotropic network (see Fig. 2(a)) shows the ideal case in which the sum of hop distances is approximately equal to the geometric distance. In general case, the shortest path between nodes A and B occurs as P2. The large deviation between the geometric distances and hop distances leads to a sharp drop in localization accuracy. For the anisotropic networks, the curved path like P3 always appears. This is exactly the reason that the DV-Hop-based localization methods perform so poorly in anisotropic networks.

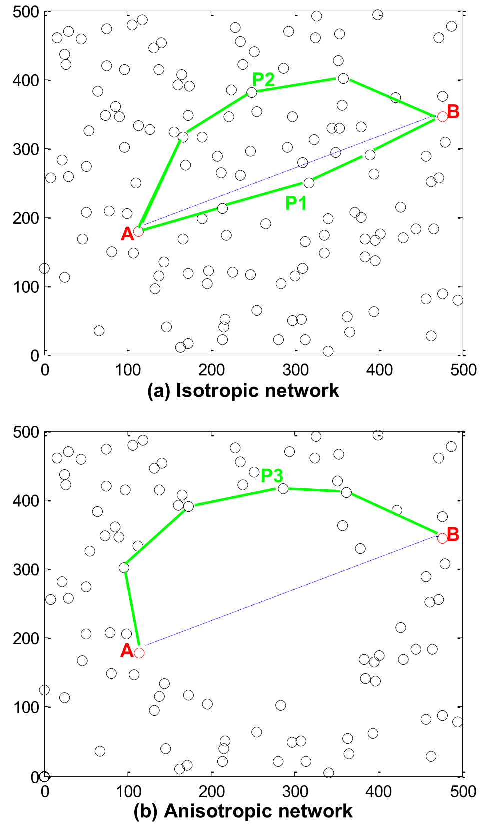

FIGURE 2. An example of network topology.

Recently, researchers propose many DV-Hop derivations which can work for both isotropy and anisotropy networks. In [23], quantum-behaved particle swarm optimization algorithm (QPSO) is reconstructed with the memetic algorithm and Lévy flight to find the best coordinates of unknown nodes. Simulations are conducted in the square and C-shaped scenarios with the randomly deployed nodes and results show that the LMQPDV-hop algorithm can effectively improve the position precision. A hybrid DECHDV-Hop localization algorithm with DV-Hop and DE is designed in [30], and its effectiveness is tested in random, grid, C-shaped random and C-shaped grid network situations. [43] proposes a new framework to localize newly deployed nodes in a pre-localized network using GADV-Hop algorithm which is simulated in random topology, C-shaped topology and W-shaped topology. As can be seen from Fig. 2, the anisotropy networks undermine the balance relationship between hop count values and estimated distances, thus its localization error is higher than that of the isotropy networks. However, these three papers avoid the estimated locations of unknown nodes falling into an infeasible region by using the meta-heuristic algorithms to locate the unknown nodes.

III. SYSTEM MODEL

This section presents several models which are applied in the proposed algorithm.

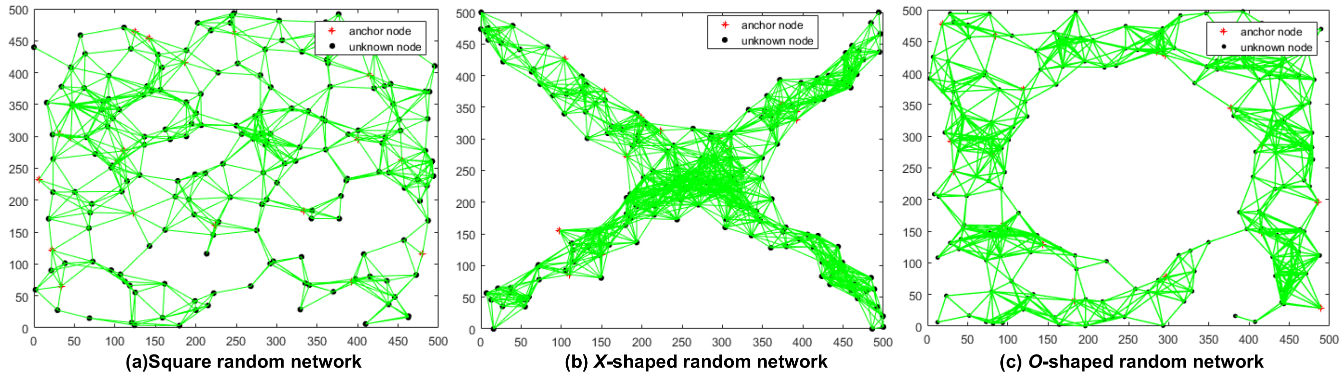

FIGURE 3. Nodes distribution of the network with $n = 200$, $m = 20$ and $R = 60$.

A. NETWORK MODEL

The model of a traditional WSN consists of n sensor nodes among which the number of anchors is m and that of unknown nodes is u in the sensing field. All anchor nodes are aware of their positions clearly with the deployment of Global Positioning System (GPS) or other devices. (x_i, y_i) and (x, y) represent coordinates of i-th anchor node and unknown node respectively. It is supposed that the communication radius is R for all sensor nodes.

In order to verify the algorithm performance, the isotropy and anisotropy networks are all chosen in this paper. There are many different models for anisotropy network [44], [45]. Here, the most representative X-shaped and O-shaped environments are built for the anisotropy network. Fig. 3 represents three different node distribution in the area of $500m \times 500m$: the square area with the length of l m, the X-shaped area with width of d m and the hole area with radius of γm.

B. DV-HOP LOCALIZATION MODEL

Briefly speaking, the DV-Hop procedure is summarized as the following steps:

Step1: Collection of minimal hop count.

All anchor nodes broadcast data packets including the coordinate information and the hop count, which is initialized to 0 and raised by 1 after each hop. If the gained value is less than the former one, the node will update it and store the minimum value. At the end of broadcast, all nodes possess the minimal hop counts.

Step2: Calculation of average hop distance.

The minimal hop value and coordinates of anchors are acquired between any two anchor nodes in stage 1. Hence, the average hop distance described as H_i is achieved by:

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

where h_{ij} represents the minimal hop value between i-th and j-th anchors. Then all the average hop distances will be broadcasted to the whole network.

Step3: Calculation of unknown nodes coordinates.

Each unknown node calculates the distance d_{ik} from the i-th anchor node to the k-th unknown node by means of multiplying average hop distance H_i obtained from stage 2 by the minimal hop count h_{ik} as follows:

$$d_{ik} = H_i \times h_{ik} \tag{2}$$

Once obtaining three or more anchors distance data, the coordinates of unknown node could be estimated by the least square method. The complete process is below.

The distances between unknown node (x, y) and anchor nodes (x_i, y_i) are:

$$(x_i - x)^2 + (y_i - y)^2 = d_i^2 \tag{3}$$

Thus, the set of distance equations for all m anchors are:

$$\begin{cases} (x_1 - x)^2 + (y_1 - y)^2 = d_1^2 \\ (x_2 - x)^2 + (y_2 - y)^2 = d_2^2 \\ \quad \vdots \\ (x_i - x)^2 + (y_i - y)^2 = d_i^2 \\ \quad \vdots \\ (x_{m-1} - x)^2 + (y_{m-1} - y)^2 = d_{m-1}^2 \\ (x_m - x)^2 + (y_m - y)^2 = d_m^2 \end{cases} \tag{4}$$

By subtracting the last equation from the previous m-1 equations, we can obtain the following forms:

$$(x_i - x)^2 - (x_m - x)^2 + (y_i - y)^2 - (y_m - y)^2 = d_i^2 - d_m^2 \tag{5}$$

$$2(x_i - x_m)x + 2(y_i - y_m)y = x_i^2 - x_m^2 + y_i^2 - y_m^2 + d_m^2 - d_i^2 \tag{6}$$

$$\text{Let } X = [x, y]^T \tag{7}$$

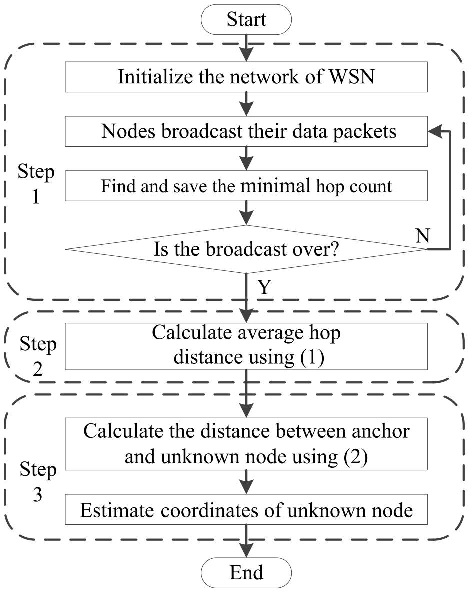

FIGURE 4. Flowchart of DV-Hop.

$$A = \begin{bmatrix} 2(x_1 - x_m) & 2(y_1 - y_m) \\ 2(x_2 - x_m) & 2(y_2 - y_m) \\ \vdots & \vdots \\ 2(x_i - x_m) & 2(y_i - y_m) \\ \vdots & \vdots \\ 2(x_{m-2} - x_m) & 2(y_{m-2} - y_m) \\ 2(x_{m-1} - x_m) & 2(y_{m-1} - y_m) \end{bmatrix} \tag{8}$$

$$b = \begin{bmatrix} x_1^2 - x_m^2 + y_1^2 - y_m^2 + d_m^2 - d_1^2 \\ x_2^2 - x_m^2 + y_2^2 - y_m^2 + d_m^2 - d_2^2 \\ \vdots \\ x_i^2 - x_m^2 + y_i^2 - y_m^2 + d_m^2 - d_i^2 \\ \vdots \\ x_{m-2}^2 - x_m^2 + y_{m-2}^2 - y_m^2 + d_m^2 - d_{m-2}^2 \\ x_{m-1}^2 - x_m^2 + y_{m-1}^2 - y_m^2 + d_m^2 - d_{m-1}^2 \end{bmatrix} \tag{9}$$

We can have

$$AX = b \tag{10}$$

According to the least mean square estimation, coordinates of unknown nodes are gotten by:

$$\hat{X} = (A^T A)^{-1} A^T b \tag{11}$$

The unknown node coordinates can be described as:

$$\begin{cases} x = \hat{X}(1) \\ y = \hat{X}(2) \end{cases} \tag{12}$$

Fig. 4 gives the flowchart of DV-Hop.

C. CHICKEN SWARM OPTIMIZATION MODEL

CSO is a popular nature inspired optimizer which imitates the hierarchical order and behavior of chickens to find food.

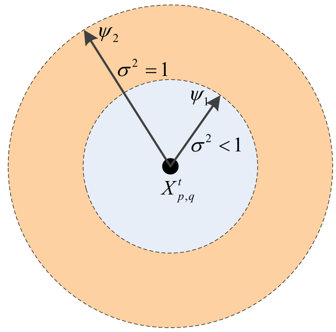

FIGURE 5. Sketch map of rooster's movement.

It can effectively extract the wisdom of chickens to optimize WSNs localization.

CSO contains lots of chicken groups, and each group includes a rooster, some hens and chicks. The clustering principle depends on the fitness function value of each chicken. The role of the individual chicken as a rooster, a hen or a chick is determined by the fitness value from low to high sequentially [46].

It is assumed that *RN*, *HN* and *CN* mean the number of roosters, hens and chicks respectively. The N fictitious individuals are depicted with locations $X^t_{p,q}$, $p \in [1, N]$, $q \in [1, D]$ at time step t foraging on D-dimensional area. The hierarchical relationship of chicken swarm remains unchanged in the same round. Current states will be corrected after the time interval of G.

Roosters with higher fitness values have the privilege of finding and searching for food than lower ones. Simply speaking, roosters with better fitness values can search for food in a wider range of places than that of the roosters with worse fitness values (Fig. 5). The mathematical expressions are as follows:

$$X^{t+1}_{p,q} = X^t_{p,q} \cdot (1 + Randn(0, \sigma^2)) \tag{13}$$

$$\sigma^2 = \begin{cases} 1 & \text{if } f_p \le f_k \\ \exp\left(\dfrac{f_k - f_p}{|f_p| + \varepsilon}\right), & \text{otherwise} \end{cases} \quad k \in [1, N], k \neq p \tag{14}$$

where $Randn(0, \sigma^2)$ shows the normal distribution with mean 0 and variance σ^2. The index k of rooster is stochastically chosen. f denotes the fitness value. The minimum constant ε is introduced to avert the divisor of 0.

Hens usually go after their partner rooster to find food. Furthermore, they also stochastically pilfer food discovered by other roosters as described in Fig. 6(a). The ruling hens are more skilled at contending for food than the compliant individuals. It can be described below.

$$X^{t+1}_{p,q} = X^t_{p,q} + O_1 \cdot Rand \cdot (X^t_{r1,q} - X^t_{p,q}) + O_2 \cdot Rand \cdot (X^t_{r2,q} - X^t_{p,q}) \tag{15}$$

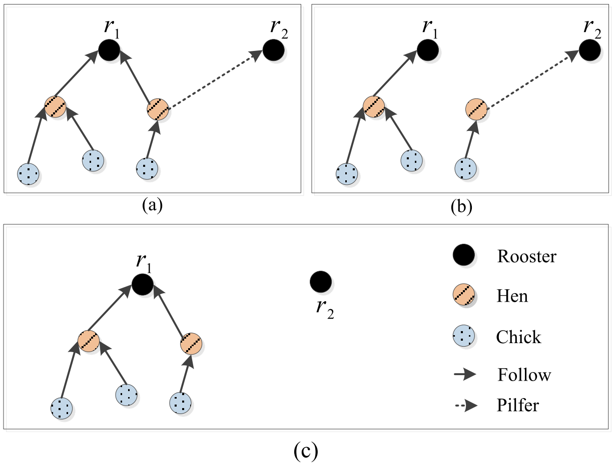

FIGURE 6. Sketch map of hen's movement.

$$O_1 = \exp\left(\frac{f_p - f_{r1}}{|f_p| + \varepsilon}\right) \tag{16}$$

$$O_2 = \exp(f_{r2} - f_p) \tag{17}$$

where *Rand* represents a random number in [0, 1]. $r_1 \in [1, N]$ expresses a rooster's index that is the *p-th* hen's partner. $r_2 \in [1, N]$ shows the index of the rooster or hen selected from the individuals where $r_1 \neq r_2$. O_1 and O_2 are selection coefficients. If $O_1 = 0$, the *p-th* hen will look for food followed by others as illustrated in Fig. 6(b). If $O_2 = 0$, the *p-th* hen will forage for food in their own territory as depicted in Fig. 6(c).

The chicks follow the mother hen to search for food. This behavior is described below.

$$X_{p,q}^{t+1} = X_{p,q}^{t} + FL \cdot (X_{z,q}^{t} - X_{p,q}^{t}) \tag{18}$$

where $X_{z,q}^{t}$ expresses the location of mother hen for the *z-th* chick ($z \in [1, N]$). *FL* is a constant, which usually be chosen randomly in the range of 0-2 considering the individual differences.

The flow chart of the CSO algorithm is illustrated in Fig. 7.

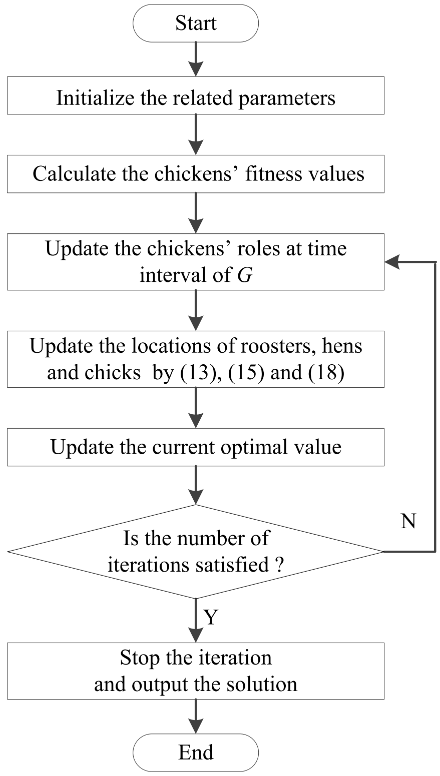

FIGURE 7. Flowchart of CSO.

IV. PROPOSED ALGORITHM

A. MOTIVATION AND PRINCIPLE OF CWDV-HOP

It is universally acknowledged that DV-Hop estimates sensor nodes' location through obtaining the distance estimation between nodes and the essential reason of poor distance estimation is the inaccuracy of average hop distance. To enhance the positional performance, weighted approach is used to improve the precision of average hop distance. The conventional wisdom is that the bigger values of hop count or/and distance are, the smaller value of weight will be. Although the weighted algorithms improve the positional precision of the sensor nodes, this statement related to the relationship among the hop counts, distances and weights has not been proved. Motivated by the weighted conjectures put forward by scholars, this study analyzes the mathematical model of average hop distance by the idea of weight, reveals its fundamental cause of error and innovatively proposes a distance-weight hop distance by introducing the new defined distance-weight factor and single-node hop distance.

In order to get more accurate positions, many nature inspired algorithms have been applied to optimize the DV-Hop localization problem. However, achieving reasonable accuracy with faster convergence time remains a matter of research. This issue motives us to find a better nature inspired algorithms for the localization optimization problem. As a bio-inspired multi-swarm algorithm, CSO not only inherits the major advantages of PSO and DE, but also extracts the chickens' swarm intelligence to solve the optimization problems efficiently. Therefore, this paper selects CSO to optimize the location of unknown nodes.

Based on the above analysis, this study designs a novel Chicken Swarm Optimization-based distance-weighted DV-Hop algorithm (CWDV-Hop). CWDV-Hop consists of four major steps.

Step 1: Acquisition of the minimum hop count.

Step 2: Calculation of distance-weighted hop distance.

Step 3: Estimation of unknown node's location.

Step 4: Optimization the coordinates of unknown node.

The proof and implementation details of CWDV-Hop will be described as the following subsections.

B. THE PROOF OF CWDV-HOP ALGORITHM

In original DV-hop algorithm, the main reason that results in error is the deviation of average hop distance for anchor nodes with (1). Next, the relationship between average hop distance and node distance will be explored.

Proposition 1: The greater the distance d_{ij} between *i-th* anchor node and *j-th* anchor node ($i \neq j$) is, the higher its effect on the average hop distance is.

Proof: On the basis of definition in (1), we can deduce that:

$$
\begin{aligned}
H_i &= \frac{\sum\limits_{i\neq j}\sqrt{(x_i-x_j)^2+(y_i-y_j)^2}}{\sum\limits_{i\neq j} h_{ij}} \\
&= \frac{\sum\limits_{i\neq j} d_{ij}}{\sum\limits_{i\neq j} h_{ij}} = \frac{d_{i1}+d_{i2}+\cdots+d_{im}}{h_{i1}+h_{i2}+\cdots+h_{im}} \\
&= \frac{h_{i1}}{\sum\limits_{i\neq j} h_{ij}} \times \frac{d_{i1}}{h_{i1}} + \frac{h_{i2}}{\sum\limits_{i\neq j} h_{ij}} \times \frac{d_{i2}}{h_{i2}} + \cdots + \frac{h_{im}}{\sum\limits_{i\neq j} h_{ij}} \times \frac{d_{im}}{h_{im}} \\
&= \alpha_{i1} \times \frac{d_{i1}}{h_{i1}} + \alpha_{i2} \times \frac{d_{i2}}{h_{i2}} + \cdots + \alpha_{im} \times \frac{d_{im}}{h_{im}} \qquad (19)
\end{aligned}
$$

where

$$\alpha_{ij} = \frac{h_{ij}}{\sum\limits_{i\neq j} h_{ij}} \qquad (20)$$

can be defined as **hop-weighted factor**. m represents the number of anchors. Here, (19) reveals that average hop distance is equivalent to the accumulation which is product of weight on the hop count and the hop distance of each anchor node.

Substituting the distance for the hop count into (19), it can be rewritten as follows:

$$
\begin{aligned}
H_i' &= \frac{d_{i1}}{\sum\limits_{i\neq j} d_{ij}} \times \frac{d_{i1}}{h_{i1}} + \frac{d_{i2}}{\sum\limits_{i\neq j} d_{ij}} \times \frac{d_{i2}}{h_{i2}} + \cdots + \frac{d_{im}}{\sum\limits_{i\neq j} d_{ij}} \times \frac{d_{im}}{h_{im}} \\
&= \omega_{i1} \times \frac{d_{i1}}{h_{i1}} + \omega_{i2} \times \frac{d_{i2}}{h_{i2}} + \cdots + \omega_{im} \times \frac{d_{im}}{h_{im}} \qquad (21)
\end{aligned}
$$

where

$$\omega_{ij} = \frac{d_{ij}}{\sum\limits_{i\neq j} d_{ij}} \qquad (22)$$

ω_{ij} is defined as**distance-weighted factor**.

The above equation reveals that with the increasing of the distance d_{ij} between any two anchors, the distance weight ω_{ij} increases. That is to say, as the distance increases, its effect on the average hop distance increases. Here the proof is ending.

The hop counts, which are obtained from the minimum path method, always have errors. But the distances between anchor nodes are certain physical values. Therefore, considering the influence of anchor node itself, CWDV-Hop algorithm introduces the distance as the weight factor.

Definition 1: Given $i, j \in [1, m]$ and m is the number of anchors, the new hop distance H_i' in (21) can be rewritten and defined as **distance-weighted hop distance**:

$$H_i' = \sum_{j=1}^{m}\left(\frac{d_{ij}}{\sum\limits_{i\neq j} d_{ij}} \times \frac{d_{ij}}{h_{ij}}\right) = \sum_{j=1}^{m}(\omega_{ij} \times H_{ij}) \qquad (23)$$

$$H_{ij} = \frac{d_{ij}}{h_{ij}} \qquad (24)$$

H_{ij} is defined as **single-node hop distance**.

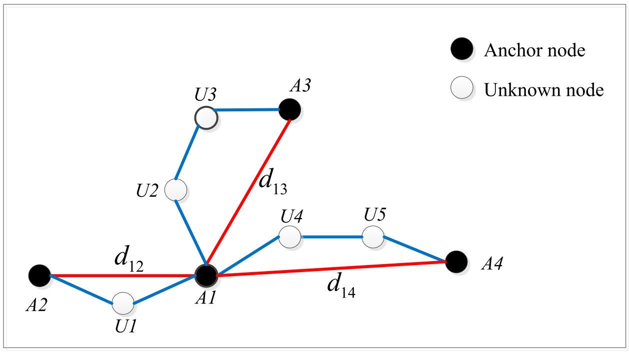

FIGURE 8. **Diagram of an example WSN.**

Fig. 8 is a sample diagram, which further explains the distance-weighted method. There are four anchor nodes (described as *A1*, $A2, \ldots, A4$) and five unknown nodes (described as *U1*, $U2, \ldots, U5$) in the diagram. The red lines are used to represent the physical distance d_{12}, d_{13}, d_{14} between anchors *A1* and *A2*, *A1* and *A3*, and *A1* and *A4*. The blue lines are used to indicate the shortest hop-counts h_{12}, h_{13}, h_{14} between them. Anchor node *A1* computes original average hop distance and new defined distance-weighted hop distance based on (19) and (23) as:

$$H_{A1} = \alpha_{12} \times \frac{d_{12}}{h_{12}} + \alpha_{13} \times \frac{d_{13}}{h_{13}} + \alpha_{14} \times \frac{d_{14}}{h_{14}} \qquad (25)$$

$$H_{A1}' = \omega_{12} \times \frac{d_{12}}{h_{12}} + \omega_{13} \times \frac{d_{13}}{h_{13}} + \omega_{14} \times \frac{d_{14}}{h_{14}} \qquad (26)$$

For

$$\alpha_{12} = \frac{h_{12}}{\sum\limits_{i\neq j} h_{1j}} \qquad (27)$$

$$\omega_{12} = \frac{d_{12}}{\sum\limits_{i\neq j} d_{1j}} \qquad (28)$$

Fig. 9 (a) is captured from Fig. 8 to show the local details of anchors *A1* and *A2*. Fig. 9 (b) and (c) illustrate other situations when there are more hops between anchors *A1* and *A2*.

We can see that the hop count h_{12} between anchor node *A1* and *A2* increases, the hop-weighted factor α_{12} increases accordingly. However, the distance-weighted factor ω_{12} is unchanged. As shown in Fig. 9(a)-(c), when the hop counts increase, the path between any two nodes may be curved. It means that the distance-weighted method can reduce the impact of curved path on average hop distance.

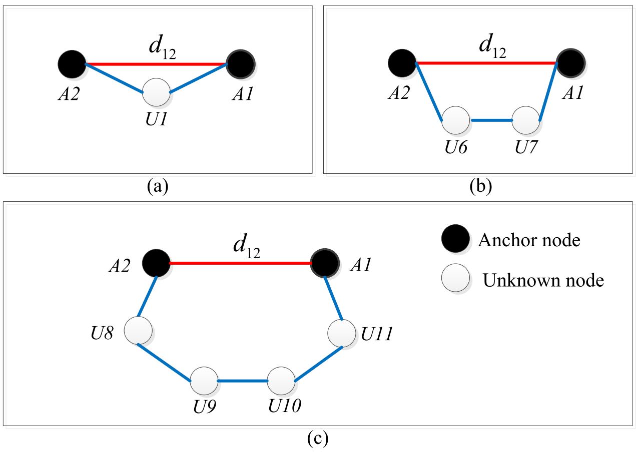

FIGURE 9. Partial diagram with increasing hops.

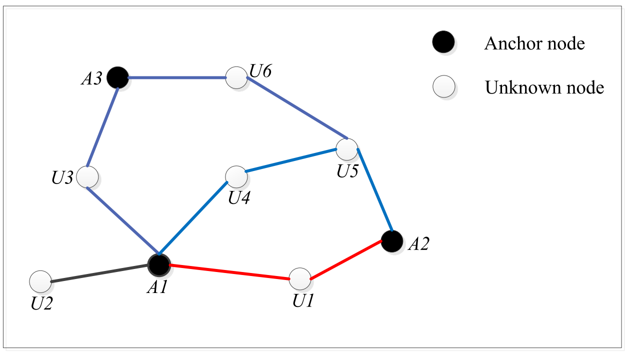

FIGURE 10. Diagram of minimum path method.

C. THE STEPS OF CWDV-HOP ALGORITHM

CWDV-Hop can be summarized as the following four steps:

Step1: Acquisition of the minimum hop count.

All nodes gain the minimum hop counts by data flooding meanwhile anchors also obtain the coordinates information. As described in Fig. 10, there are three paths between anchors *A1* and *A2*, and the hop counts are below.

5: *A1—U3—A3—U6—U5—A2*

3: *A1—U4—U5—A2*

2: *A1—U1—A2*

Therefore, the minimum hop counts between *A1* and *A2* are 2 (see the red lines).

Step2: Calculation of distance-weighted hop distance.

The inaccuracy of average hop distance mainly attributes to the final result of localization. In accordance with the relationship between average hop distance and corresponding distance proved in Proposition 1, the distance-weighted hop distance can be calculated by (23) according to the distance-weighted factor in (22) and single-node hop distance (24).

Step3: Estimation of unknown node's location.

Two-dimensional hyperbolic approach is utilized to gain the coordinates of unknown nodes in CWDV-Hop. As the distance difference between an unknown node U and two

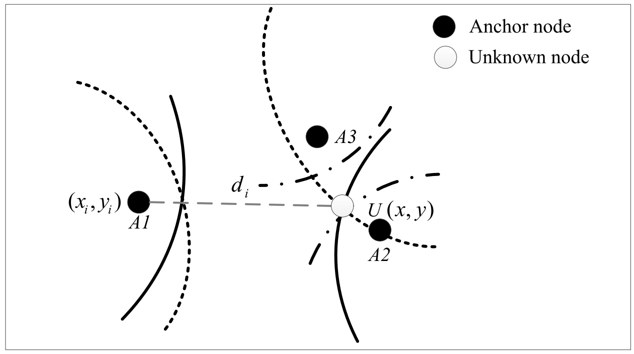

FIGURE 11. Diagram of two-dimensional hyperbola method.

anchors *A1* and *A2* is a fixed value, the location of unknown node U is must on the hyperbola curves (black solid curves in Fig. 11) basing on the definition of hyperbola. This hyperbola takes anchors *A1* and *A2* as the focus and distance between *A1* and *A2* as the focal length. In the same way, we can get the hyperbolas for anchors *A1* and *A3* (the dotted curves) and *A2* and *A3* (the dash-dot curves). Finally, the location of unknown node U is determined by the intersection point of three or more groups of hyperbolas. The whole process is explained below.

According to (3), there are the following expressions:

$$x_i^2 + y_i^2 - 2x_i x - 2y_i y + x^2 + y^2 = d_i^2 \tag{29}$$

$$-2x_i x - 2y_i y + x^2 + y^2 = d_i^2 - x_i^2 - y_i^2 \tag{30}$$

$$\text{Let} \quad Z_c = \left[x, y, x^2 + y^2\right]^T \tag{31}$$

$$G_c = \begin{bmatrix} -2x_1 & -2y_1 & 1 \\ -2x_2 & -2y_2 & 1 \\ \ldots & \ldots & \ldots \\ -2x_i & -2y_i & 1 \\ \ldots & \ldots & \ldots \\ -2x_{m-1} & -2y_{m-1} & 1 \\ -2x_m & -2y_m & 1 \end{bmatrix} \tag{32}$$

$$L_c = \begin{bmatrix} d_1^2 - x_1^2 - y_1^2 \\ d_2^2 - x_2^2 - y_2^2 \\ \vdots \\ d_i^2 - x_i^2 - y_i^2 \\ \vdots \\ d_{m-1}^2 - x_{m-1}^2 - y_{m-1}^2 \\ d_m^2 - x_m^2 - y_m^2 \end{bmatrix} \tag{33}$$

We can get

$$G_c Z_c = L_c \tag{34}$$

With the estimation method of standard minimum mean square, it can be got

$$Z_c = (G_c^T G_c)^{-1} G_c^T L_c \tag{35}$$

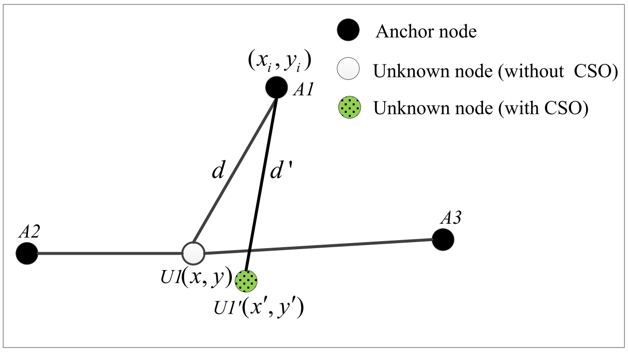

FIGURE 12. Schematic diagram of objective function.

Unknown node coordinates can be described as:

$$\begin{cases} x = Z_c(1) \\ y = Z_c(2) \end{cases} \tag{36}$$

Comparing the least square method (*i.e.* (4)-(12)) with the hyperbolic method (*i.e.* (29)-(36)), they have something in common: both transform the nonlinear equation (*i.e.* (3)) into the linear ones (*i.e.* (10) and (34)). However, in the whole process of transformation, the least square method requires that the first *m-1* equation subtract the *m-th* equation. In (9), once a large error in d_m will lead to a large error in $d_m^2 - d_i^2$. That is to say, the error of d_m will flood into all m-1 sub-equations of (9). But, the hyperbolic method does not involve the action of subtraction. Therefore, the more precise hyperbolic method is used in this paper.

Step4: Optimization the coordinates of unknown node.

In WSNs, it is the primary objective for localization approaches to acquire more accurate location of unknown nodes. Consequently, CSO is used to optimize the estimated coordinate obtained by Step 3 to find the best result of unknown nodes. The specific implementation of CSO is as follows:

1) Initialization: The parameters of CSO, including the size of chicken swarm N, the maximum iterations *M1*, the time interval G of status update for chicken swarm, the number of roosters *RN*, hens *HN*, chicks *CN*, are initialized.

2) Calculating the fitness value: In WSN, the localization problem is formed as an optimization problem where the localization error is considered as the objective function.

As shown in Fig. 12, (x_i, y_i) represents the position of *i-th* anchor *A1*. Meanwhile $U1(x, y)$ and $U1'(x', y')$ express the estimated positions of unknown node without and with the CSO method respectively. The distances between *A1* and *U1*, *A1* and *U1'* are d and d'. The minimal localization deviation model is built by applying the distance difference of d and d'. The objective function is shown below:

$$F(x', y') = Min(\left|d' - d\right|) \tag{37}$$

where

$$\begin{cases} d = \sqrt{(x_i - x)^2 + (y_i - y)^2} \\ d' = \sqrt{(x_i - x')^2 + (y_i - y')^2} \end{cases} \tag{38}$$

In accordance with (37), the fitness function of all chickens in the swarm is usually calculated as follows:

$$f(x', y') = \frac{1}{m}\sum_{i=1}^{m}\left|F(x', y')\right| \tag{39}$$

or

$$f(x', y') = \frac{1}{m}\sum_{i=1}^{m}\left(F(x', y')\right)^2 \tag{40}$$

Considering the fact that the estimated distance error also increases as the value of hop-count increases, the fitness function of DV-Hop-based localization algorithm can be optimized in a weighted manner by the reciprocal of hop-count h [18], [33], [35].

$$f(x', y') = \frac{1}{m}\sum_{i=1}^{m}\frac{1}{h^2}\left(F(x', y')\right)^2 \tag{41}$$

For each unknown node, the estimated position (x', y') with the CSO method is assigned initially as:

$$(x', y') = (x, y) + \frac{R}{2} \times rand(1, 2) \tag{42}$$

where (x, y) is obtained from the above step 3 of proposed algorithm by (36).

Based on the fitness value of the chickens, the hierarchal order, mother-child relationship and dominance relationship will be assigned every time interval of G.

3) Update: The positions of all chickens (*i.e.* roosters, hens and chicks) are updated according to (13), (15) and (18) respectively where $X_{p,q}^t$ is just the estimated position (x', y') of unknown node. CSO obtains the possible optimal solution of each unknown node by minimizing the fitness function after several iterations.

4) Output: The above process of update stops when the ending conditions are met. The best position found currently by the chicken swarm will be considered as the final result of each unknown node.

It is worthy noted that the location estimation process of each unknown node is independent of one another. That means every unknown node requires a solitary CSO particle to obtain its coordinates.

D. THE COMPLETE CWDV-HOP ALGORITHM

The flowchart of CWDV-Hop is shown in Fig. 13. Only to study the merits of distance-weighted hop distance, CWDV-Hop without CSO is named as WDV-Hop for convenience. In order to verify the performance of CSO in optimizing the localization results of DV-Hop, CWDV-Hop without distance-weighted hop distance is named as CDV-Hop.

The pseudo code of CWDV-Hop is described in Table 2.

V. PERFORMANCE EVALUATION

A. SIMULATION PARAMETERS

To verify the performance of the proposed scheme, three groups of simulations have been conducted in Matlab2016a,

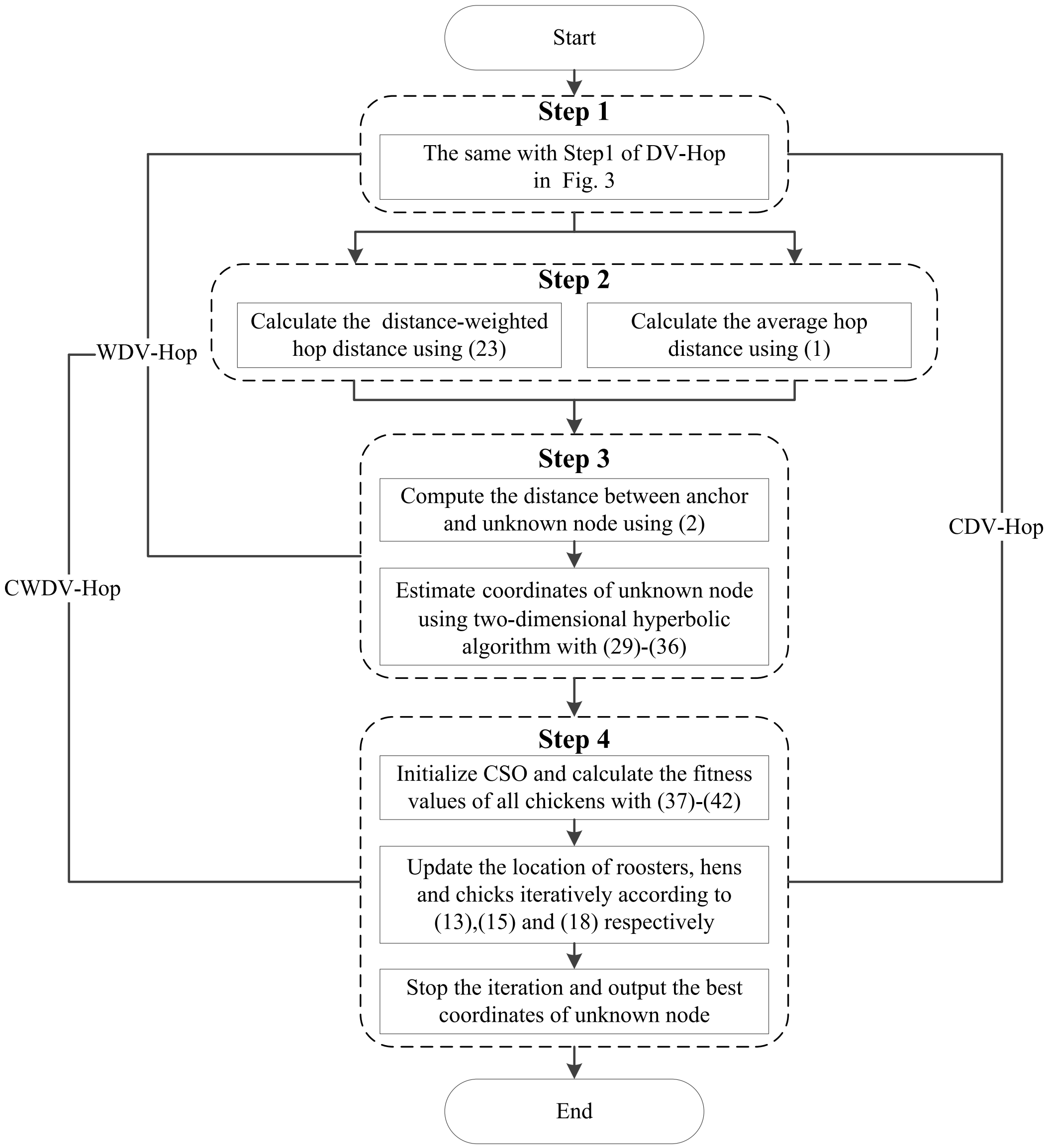

FIGURE 13. Flowchart of proposed CWDV-Hop algorithm.

and implemented on Intel core i5-3320m CPU and RAM of 4 GB. The first one just conducts a comparative study on the proposed WDV-Hop method, traditional method and iDV-Hop1 in [22]. The second one compares the CDV-Hop algorithm with the traditional method and iDV-Hop2 in [33]. The third one evaluates the proposed CWDV-Hop algorithm compared with iDV-Hop2 in [33], iDV-Hop3 in [24] and iDV-Hop4 in [30].

100 independent tests are carried out to assess the performance of all methods. Three different network topologies are selected as Fig. 3. For each network, the setups of system parameters are given in Table 3. The intention of selecting different number of anchors, different number of nodes and different communication range is to explore the localization performance of all algorithms under different network topology in terms of the anchor proportion, the node density and the node connectivity. PSO and DE variables are set according to the comparative paper [33] and [30]. CSO parameters are also varied to find the best optimization performance.

To compare the superiority of CWDV-Hop with its counterparts, the following metrics are taken into account.

Localization Error (EL): the deviation between the true position (x_0, y_0) and estimated coordinate (x, y) of unknown node. Mathematically, it is expressed by the following formula:

$$EL = \sqrt{(x - x_0)^2 + (y - y_0)^2} \tag{43}$$

Localization Accuracy (LA): the proportion of sum of localization error to the number of unknown nodes u. It can be expressed as follows:

$$LA = \frac{\sum_{i=1}^{u} \sqrt{(x_i - \hat{x}_i)^2 + (y_i - \hat{y}_i)^2}}{u \times R} \times 100\% \tag{44}$$

where $u = n\text{-}m$.

TABLE 2. The Pseudo Code for CWDV-Hop.

```
Input: total number of sensor nodes n, the number of anchors m and
        communication range R;
Generate random network topology;
for i=1: n do
  for j=1: n do
    Compute the distance d between all nodes using (2) and initialize
    h_ij=0;
    if d<=R
      h_ij = 1;
    else
      if i==j
        h_ij = 0;
      else
        h_ij = ∞;
      end if
    end if
  end for
end for
Calculate the minimum hop count h_ij between nodes by shortest
  path algorithm;
for i==1: m
  for j==1: m
    Calculate the correction factor ω_ij using (22);
  end for
end for
Calculate the distance-weighted hop distance H' using (23);
for i=1 to n-m
  Compute the distance d_i using (2);
end for
Calculate the unknown node coordinates (x, y) by two-dimensional
  hyperbolic localization algorithm using (29)-(36);
Initialize the related parameters of CSO where the estimated
  coordinates (x', y') of unknown node is assigned initially by (42);
Evaluate the chicken's fitness values using (41);
Update the location (x', y') iteratively in accordance with (13),(15)
  and (18);
Stop the iteration until the ending requirement is satisfied;
Output: the best coordinates (x', y') of unknown node.
```

TABLE 3. Simulation parameters.

Parameters	Symbol	Value
Network		
Total runs	M	100
Length of area	l	500
Total number of nodes	n	300 - 500
Number of anchor nodes	m	30 - 60
Communication radius	R	60 - 100
CSO		
Number of iterations	$M1$	1-100
Size of chicken swarm	N	5-100
Time of status update	G	2-30
Number of roosters	RN	0.05N-0.4N
Number of hens	HN	0.55N-0.9N
Number of chicks	CN	0.05N
PSO		
Number of iterations	$M2$	1-100
Size of particle swarm	P	5-100
Learning coefficient	$C1$, $C2$	2.05, 2.05
Particle's velocity	$Vmax$	10

Localization Coverage Rate (LCR): the ratio of the number of nodes u_f localized successfully to the total number of unknown nodes u.

$$LCR = \frac{u_f}{u} \times 100\% \tag{45}$$

For the parameters in Table 3, the localization coverage rate can reach almost 100%. As all algorithms being compared are based on the traditional DV-Hop algorithm which is a multihop scheme, localization coverage rates of all algorithms are the same. Therefore, we only compare the localization accuracy in the next simulations.

B. SIMULATION FOR WDV-HOP

The first group of simulation is designed to contrast WDV-Hop with traditional DV-Hop and iDV-Hop1. From (44), it is obvious that three parameters n, m, R can affect the average localization accuracy. Therefore, the sensitivity analysis of these parameters is provided in this section.

1) SENSITIVITY OF THE NUMBER OF ANCHOR NODES

It seems to be a hard task to localize an unknown node when it cannot calculate or measure its distance from three or more anchor nodes. In this situation, increasing the number of anchor nodes could be a possible solution. Considering this case, simulations are done to evaluate the effect of anchor proportion in localization accuracy. In this experiment, the number of anchor nodes m varies from 30 to 60 with $n=300$ and $R = 60$ unchanged. As illustrated in Fig. 14(a), Fig. 15(a) and Fig. 16(a), localization errors of the three algorithms show a gradual downward trend with the number of anchors increasing. However, localization accuracy of WDV-Hop increases by 25.8%, 17.5% and 21.6% in comparison with that of iDV-Hop, and about 23.9%, 19.1% and 19.4% in comparison with that of iDV-Hop1 for square, X-shaped and O-shaped networks respectively.

2) SENSITIVITY OF THE COMMUNICATION RADIUS

The communication range R gradually increases from 60m to 100m, and the other two parameters remain $n = 300$, $m = 30$ unchanged for all situations. As illustrated in Fig. 14(b), Fig. 15(b) and Fig. 16(b), when the communication radius increases, localization error decreases for all mentioned algorithms. Therefore, communication radius is counted as one of the most important parameter in WSN localization. The WDV-Hop scheme can improve the localization accuracy by 25.1%, 17.8% and 26.1% in contrast with DV-Hop, and about 24%,18.2% and 29.6% compared with iDV-Hop1 in square, X-shaped and O-shaped situations respectively.

3) SENSITIVITY OF THE TOTAL NUMBER OF NODES

In this experiment, the total number of nodes n is varied from 300 to 500 with the interval of 50, that is, the node density is assigned with [0.12, 0.14, 0.16, 0.18, 2] m^2. The communication range R is kept to 60m and the percentage of anchor nodes is fixed at 10%. As illustrated in Fig. 14(c), Fig. 15(c) and Fig. 16(c), as the number of nodes increases, errors show a downward trend. This is because with the increase of node density, each node can contain more one-hop nodes and the connectivity of WSN also becomes stronger simultaneously. However, the localization accuracy of WDV-Hop scheme can improve 23.5%, 16.2% and 15% compared

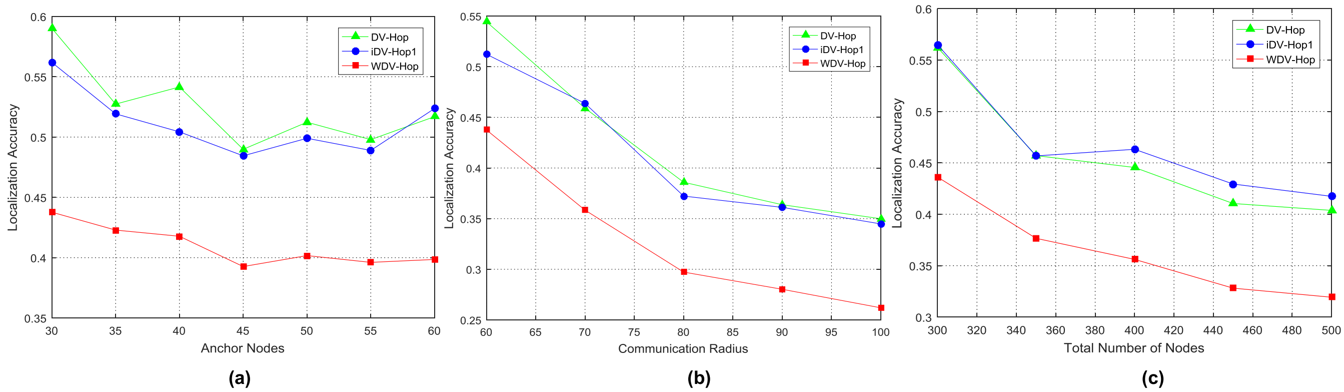

FIGURE 14. **Localization accuracy in square network (a) n=300, R=60; (b) n=300, m=30; (c) R=60, Percent of anchors=10%.**

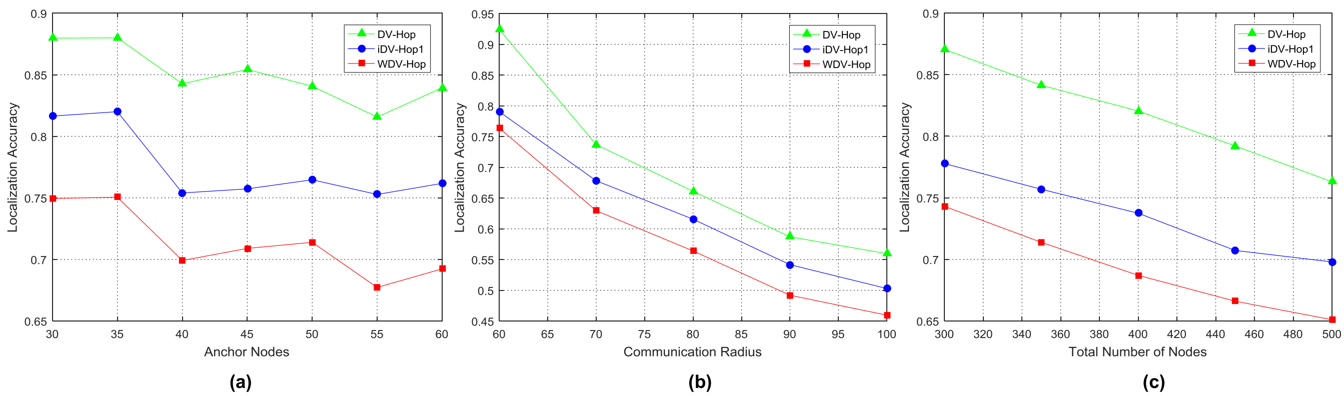

FIGURE 15. **Localization accuracy in X-shaped network (a) n=300, R=60; (b) n=300, m=30; (c) R=60, Percent of anchors=10%.**

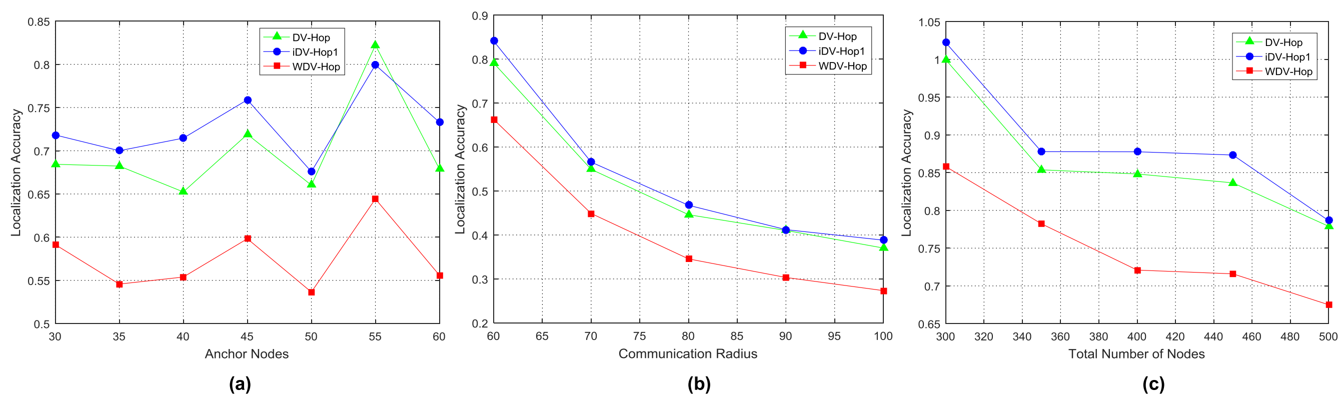

FIGURE 16. **Localization accuracy in O-shaped network (a) n=300, R=60; (b) n=300, m=30; (c) R=60, Percent of anchors=10%.**

with DV-Hop, about 22.4%, 15.1% and 17.9% compared with iDV-Hop1 for square, X-shaped and O-shaped random situation respectively.

For the three different scenarios of square, X-shaped and O-shaped network, the proposed WDV-Hop method significantly outperforms its counterparts no matter which parameter is concerned. This is because the distance-weighted method can reduce the impact of curved path on average hop distance. Sometimes, the localization errors of iDV-Hop1 are even worse than the traditional DV-Hop method, especially in the O-shaped network.

C. SIMULATION FOR CDV-HOP

The second group of simulation is designed to investigate the optimization performance of CSO with different parameters. Next, the sensitivity of CSO parameters is analyzed with the system parameters of Dv-Hop setting to $n=300$, $m=30$ and $R = 60$.

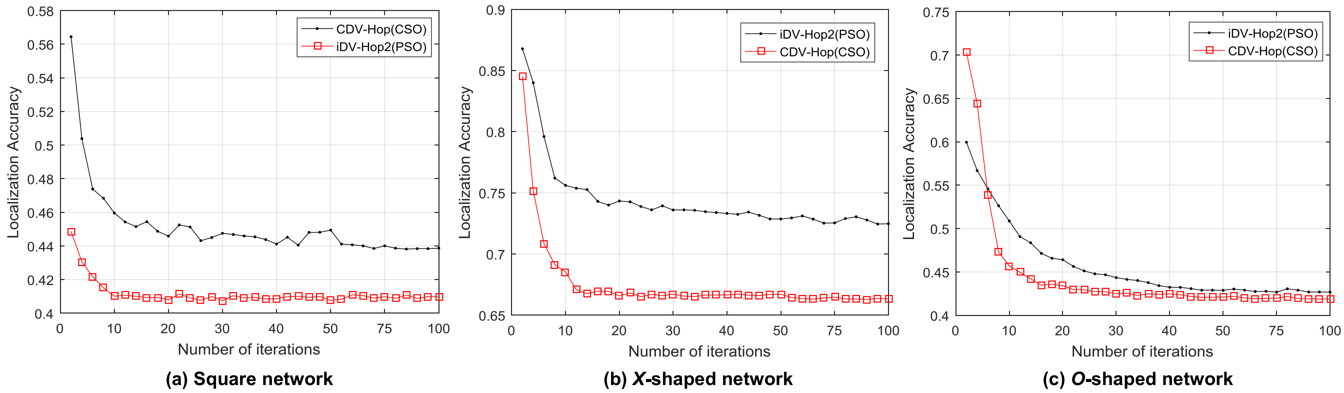

FIGURE 17. Localization accuracy with different number of iterations.

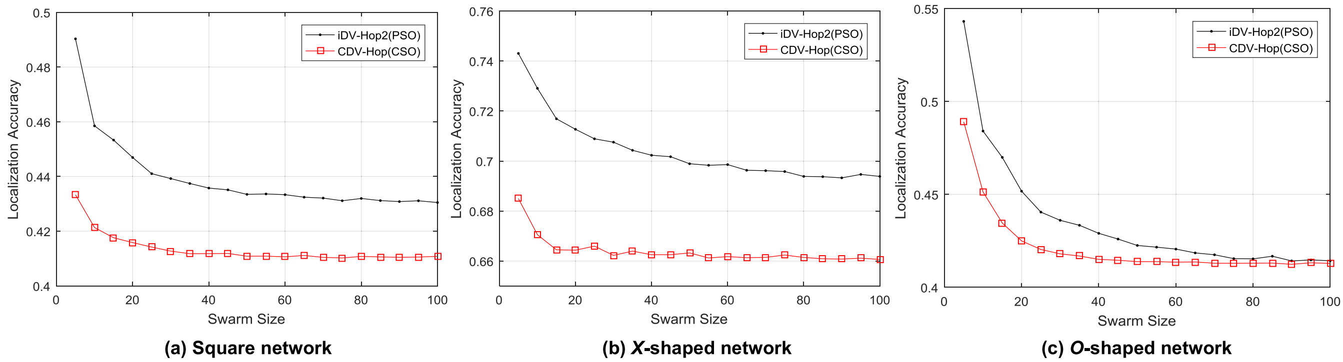

FIGURE 18. Localization accuracy with different swarm size.

1) SENSITIVITY OF THE NUMBER OF ITERATIONS

In this experiment, the number of iterations for CSO and PSO varies from 1 to 100. That is to say, $M1 \in [1, 100]$, $M2 \in [1, 100]$. The other parameters of CSO are selected as $N=20$, $RN=0.35N$, $HN=0.6N$, $CN=0.05N$ and $G=10$. While those of PSO are set to $P=20$, $C1= C2=2.05$, $Vmax=10$, which are the same as the original paper [33]. Figure 17 shows the effect of iterations on localization accuracy, that is the convergence speed of algorithms.

According to the results of Fig. 17, the proposed CDV-Hop method with CSO has faster convergence speeds and smaller localization error than iDV-Hop2 with PSO in the square, *O*-shaped and *X*-shaped networks. That is to say, CSO can achieve more reasonable localization accuracy with faster convergence time than PSO. It also can be clearly seen that the localization errors of iDV-Hop2 continue to fluctuate with the increase of iterations. In contrast, CDV-Hop always has a stable value, which demonstrates the robustness of CSO. The relationship between the number of iterations and the localization accuracy in three topological networks proves that the CDV-Hop method with CSO has small localization error, fast convergence speed, robustness and stability.

2) SENSITIVITY OF THE SWARM SIZE

In this experiment, the size of chicken swarm N for CSO is varied from 5 to 100 with the interval of 5. In order to make a fair comparison, the size of particle swarm P for PSO is selected as the same as that for CSO. The other parameters of CSO are chosen as $M1=50$, $RN=0.35N$, $HN=0.6N$, $CN=0.05N$ and $G=10$. While those of PSO are set to $M2=50$, $C1= C2=2.05$, $Vmax=10$. Figure 18 shows the effect of swarm size on localization accuracy. As shown in Fig.18, with the increase of swarm size, the localization errors of CDV-Hop and iDV-Hop2 first decline sharply and then get the optimal value gradually. By comparison, it can be seen that the proposed CDV-Hop algorithm with CSO outperforms the iDV-Hop2 method with PSO.

3) SENSITIVITY OF THE TIME INTERVAL OF STATUS UPDATE

The time interval G of status update gradually increases from 1 to $M1$ in this experiment, and the other parameters of CSO are selected as $M1=50$, $N=20$, $RN=0.35N$, $HN=0.6N$ and $CN=0.05N$. The experimental results are shown in Fig. 19 where it is clear that the differences in the localization results of different time intervals are not large. This means that the time interval of status updating has little impact on the performance of CDV-Hop.

4) SENSITIVITY OF THE NUMBER OF ROOSTERS

In CSO algorithm, rooster is considered as the head of a group and leads the subordinates. Therefore, the number of roosters, which ultimately indicates the number of groups in the chicken swarm, plays a significant role in CSO algorithm. That is to say, the more the number of roosters is, the more

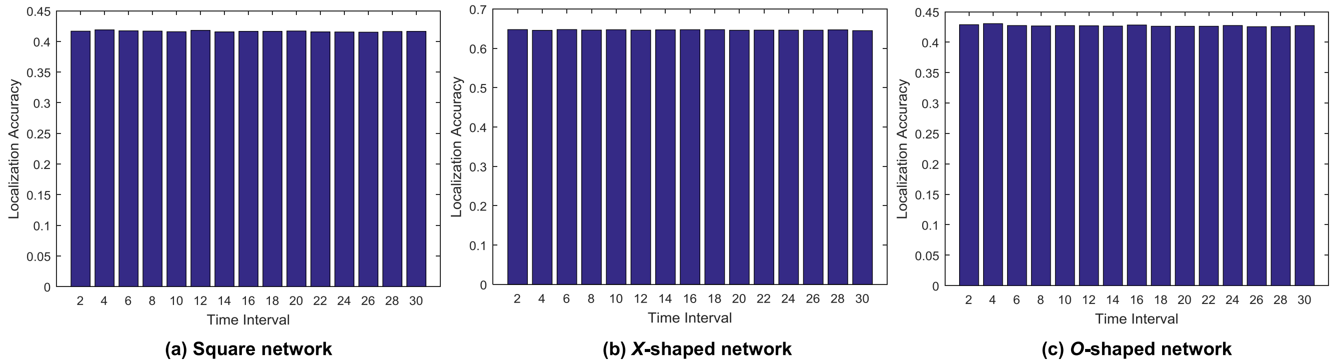

FIGURE 19. Localization accuracy with different time interval of status update.

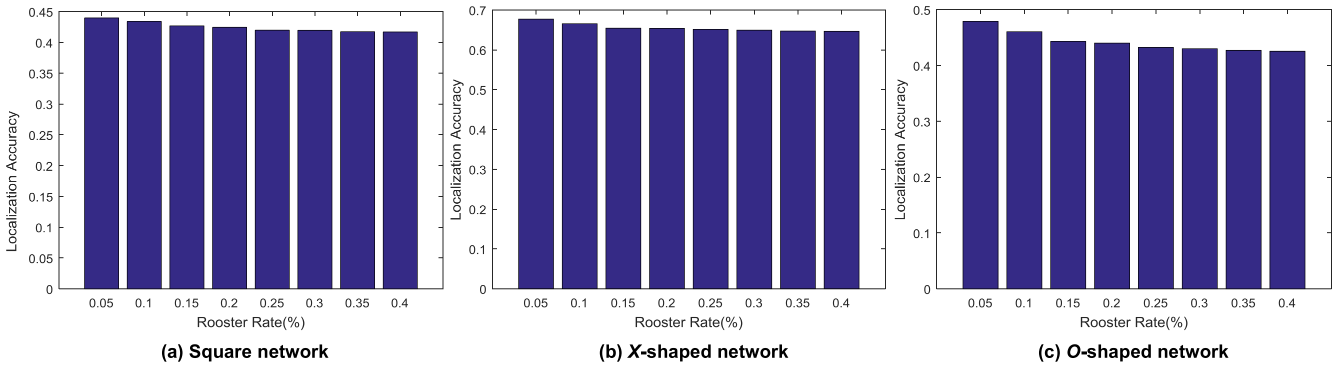

FIGURE 20. Localization accuracy with different time interval of status update.

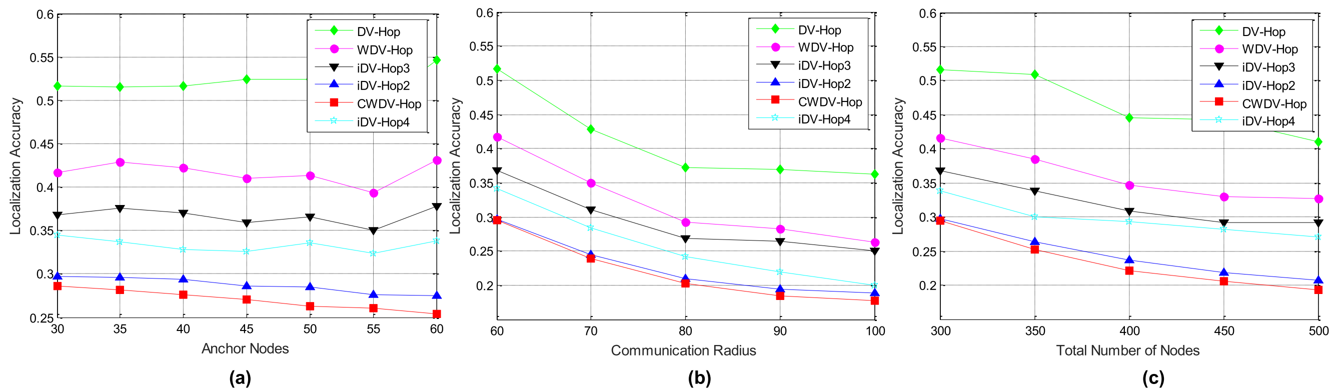

FIGURE 21. Localization accuracy in square network (a) n=300, R=60; (b) n=300, m=30; (c) R=60, Percent of anchors=10%.

the number of groups is and the better the localization accuracy obtained through the collective searching approach is. The effect of rooster number in WSN localization is shown in Fig. 20 where the number of roosters *RN* is represented by the product of rooster rate β and swarm size N. The percentage of rooster β is tuned from 0.05 to 0.4 with the interval of 0.05. Other parameters are set to $M1$=50, N=20, G=10, $RN=\beta N$, HN=(1-$\beta-$0.05) N and CN=0.05 N. It is clear that when the rooster rate increases the localization error decreases, though different rooster rate β has a tiny influence on the localization accuracy.

D. SIMULATION FOR CWDV-HOP

To research the performance of CWDV-Hop method, Figs. 21-23 illustrate the localization accuracy in square, X-shaped and O-shaped random networks respectively.

Fig. 21(a), Fig. 22(a) and Fig. 23(a) show the localization accuracy when the number of anchors increases from 30 to 60 with the step of 5. The number of nodes is 300, and communication radius is 60m. Generally, as the number of anchors rises, the trend of the localization error declines gradually for most approaches in square, X-shaped and O-shaped random networks. When there are more anchors,

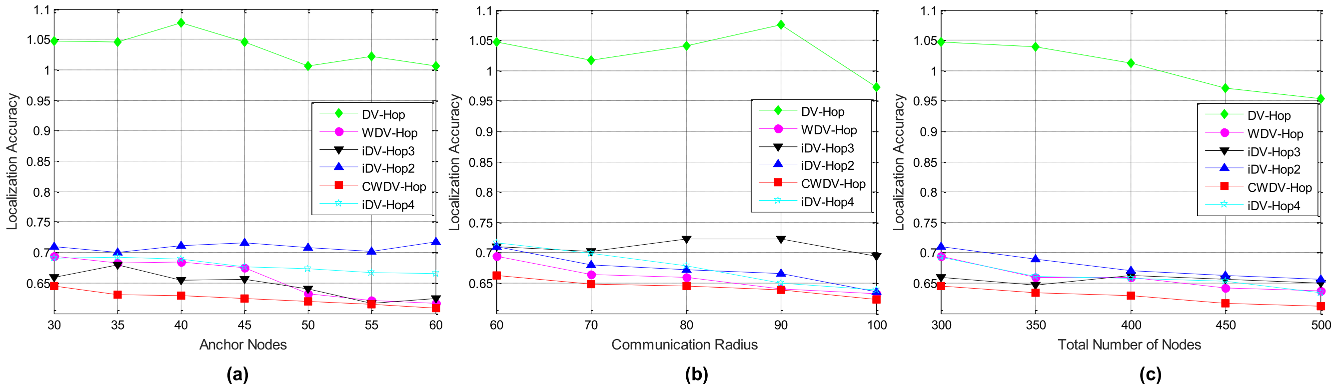

FIGURE 22. Localization accuracy in X-shaped network (a) n=300, R=60; (b) n=300, m=30; (c) R=60, Percent of anchors=10%.

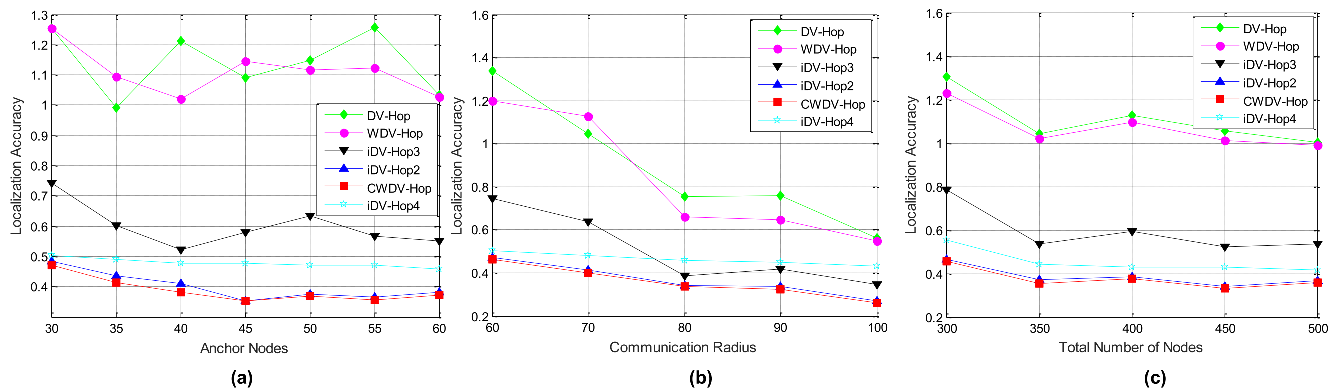

FIGURE 23. Localization accuracy in O-shaped network (a) n=300, R=60; (b) n=300, m=30; (c) R=60, Percent of anchors=10%.

the hop counts between anchor and unknown nodes increase, the error is reduced accordingly. Hence, the localization accuracy of all methods decreases. When the number of anchors is 60, in comparison with DV-Hop, WDV-Hop, iDV-Hop3, iDV-Hop4 and iDV-Hop2, the localization accuracy of CWDV-Hop algorithm increases about 53.6%, 41.2%, 32.9%, 18.9% and 7.6% respectively in square random network. In contrast with its counterparts, the performance of CWDV-Hop algorithm is improved about 39.5%, 1.3%, 2.5%, 8.1% and 15.1% in X-shaped network. For the O-shaped network, CWDV-Hop increases by 40.3%, 38.4%, 26.4%, 18.8% and 2.1%.

Communication radius of sensor nodes is an important factor for the localization algorithms. Thus, this set of tests computes the localization accuracy with the communication radius varied from 60m to 100m with an interval of 10m in Fig. 21(b), Fig. 22(b) and Fig. 23(b). The number of nodes is 300 and anchors is 30. As can be seen from Fig. 21(b), Fig. 22(b) and Fig. 23(b), with the increase of the communication range, localization accuracy shows a downward trend in square, X-shaped and O-shaped random networks. To a certain extent, in contrast with DV-Hop, WDV-Hop improves the accuracy largely. It has almost the same performance with CWDV-Hop in Fig. 21(b). When communication radius is 100m, the accuracy of CWDV-Hop is improved by 51.1%, 32.6%, 29.1%, 11.2% and 5.9% respectively in comparison with the counterparts in square network, 65.9%, 1.5%, 50.2%, 2.4% and 2.1% in X-shaped random network and 53.1%, 52.4%, 24.5%, 39.4% and 3.7% in O-shaped network.

Fig. 21(c), Fig. 22(c) and Fig. 23(c) shows the final result of the total number of nodes from 300 to 500, where the number of anchor nodes is 30 and the communication radius is 60 m. As the total number of nodes increases, the trend is downwards. That is because the network connectivity is greatly improved with the more nodes increase. When the number of nodes is 500 in square situation, the localization accuracy of CWDV-Hop is promoted by 52.8%, 41%, 33.9%, 28.8% and 6.7% respectively in comparison with the others. Localization accuracy of CWDV-Hop algorithm increases by 35.8%, 3.9%, 5.9%, 3.5% and 6.6% in X-shaped situation, and it increases by 46.9%, 32.1%, 27.4%, 13.1% and 3.4% respectively in O-shaped situation. Obviously, our algorithm performs the best.

The proposed CWDV-Hop method has better localization accuracy than its counterparts in both isotropy and anisotropy networks. This is because the compared algorithms use the hop-weight factor to calculate the average hop distance, and the hop counts always have errors when the minimum path is curved. That is to say, the inevitable curve paths, especially in the anisotropy networks, undermine the balance relationship between hop count values and estimated distances (see Fig.2 and Fig.9). As the distances between anchors are certain physical values, CWDV-Hop depends on the distance-weight factor to reduce the impact of curved path on average hop distance. In addition, the meta-heuristic algorithm, CSO, also plays important roles in reducing the positioning error produced by curved paths. CWDV-Hop avoids the estimated locations of unknown nodes falling into an infeasible region by using the meta-heuristic algorithms to locate the unknown nodes.

E. LOCALIZATION ERROR OF CWDV-HOP

Localization error of all above algorithms is listed in Table 4-6. The simulation is conducted for 300 nodes randomly deployed in the field of $500m \times 500m$ where anchor nodes are 30 and communication radius of each node is 60m.

TABLE 4. Localization error for square.

Localization Method	Max(m)	Min(m)	Average(m)
DV-Hop	92.32	1.78	35.79
WDV-Hop	88.07	0.49	23.17
iDV-Hop1	84.13	0.58	25.85
iDV-Hop2	89.69	0.39	16.86
iDV-Hop3	85.50	1.12	19.89
iDV-Hop4	88.02	0.61	18.71
CWDV-Hop	88.94	0.51	16.23

TABLE 5. Localization Error for *X*-shaped.

Localization Method	Max(m)	Min(m)	Average(m)
DV-Hop	163.55	1.37	63.02
WDV-Hop	131.79	3.57	39.59
iDV-Hop1	166.17	0.51	43.81
iDV-Hop2	117.61	0.93	42.31
iDV-Hop3	98.91	1.88	41.90
iDV-Hop4	89.50	1.02	40.09
CWDV-Hop	71.63	0.82	39.02

TABLE 6. Localization Error for *O*-shaped.

Localization Method	Max(m)	Min(m)	Average(m)
DV-Hop	139.25	7.41	60.98
WDV-Hop	130.34	1.99	53.62
iDV-Hop1	134.61	7.77	60.15
iDV-Hop2	75.73	1.51	26.73
iDV-Hop3	87.80	1.63	34.04
iDV-Hop4	86.13	1.47	28.53
CWDV-Hop	81.90	1.13	23.58

Table 4-6 list the localization errors of the square, X-shaped and O-shaped network respectively. It can be seen that the localization error of WDV-Hop is less than the traditional DV-Hop and iDV-Hop1. That is because the distance-weighted method introduced in WDV-Hop, greatly reduces the effect of curved path on the average hop distance, and two-dimensional hyperbola approach also increases the accuracy in the process of computing the coordinates of unknown nodes. Moreover, the CSO algorithm applied in CWDV-Hop optimizes the position accuracy of unknown nodes and is verified to be of better performance than PSO and DE. According to the three merits, the proposed CWDV-Hop method gives better performance compared with iDV-Hop2, iDV-Hop3 and iDV-Hop4.

TABLE 7. Algorithm complexity.

Algorithms	Time Complexity	Space Complexity
DV-Hop	$O(n^2)$	$O(1)$
WDV-Hop	$O(n^2)$	$O(1)$
CDV-Hop	$O(M1*N*(n-m))$	$O(N)$
iDV-Hop1	$O(n^2)$	$O(1)$
iDV-Hop2	$O(M2*P*(n-m))$	$O(P)$
iDV-Hop3	$O(n^2)$	$O(1)$
iDV-Hop4	$O(M3*Q*(n-m))$	$O(Q)$
CWDV-Hop	$O(M1*N*(n-m))$	$O(N)$

F. LOCALIZATION RESULT ANALYSIS OF CWDV-HOP

Fig. 24(a)-(c) and Fig. 25(a)-(c) show the actual location and estimated location of DV-Hop and CWDV-Hop in square, X-shaped and O-shaped simulation area. In general, the error based on CSO is much smaller than that of the original algorithm. Moreover, CWDV-Hop can greatly reduce the error for some nodes distributed on boundary.

G. COMPLEXITY ANALYSIS OF CWDV-HOP

The time complexity and space complexity are taken into consideration in this section. Assuming the WSN consists of n sensor nodes among which the number of anchors is m, then the maximum iterations of CSO, PSO and DE are $M1$, $M2$ and $M3$, the sizes of CSO and PSO are N, P and Q respectively. The time interval of status update for chicken swarm is G.

In step 1, all algorithms need calculate the minimum hop counts, so the time complexity is $O\left(n^2\right)$. In step 2 of calculating the average hop distance, DV-Hop, CDV-Hop, iDV-Hop2, iDV-Hop3 and iDV-Hop4 use the traditional way and the time complexity is $O\left(m^2\right)$. Although iDV-Hop1, WDV-Hop and CWDV-Hop use different weighted methods, they have the same time cost. In step 3 of estimating unknown node's position, DV-Hop, CDV-Hop, iDV-Hop1 and iDV-Hop4 use the least square method, and the time complexity is $O\left(m * (n-m)\right)$. iDV-Hop3 uses the weighted least square method, but its time complexity is unchanged. WDV-Hop, iDV-Hop2 and CWDV-Hop using the two-dimensional hyperbolic approach which is a kind of multilateral localization algorithm, and their time complexity

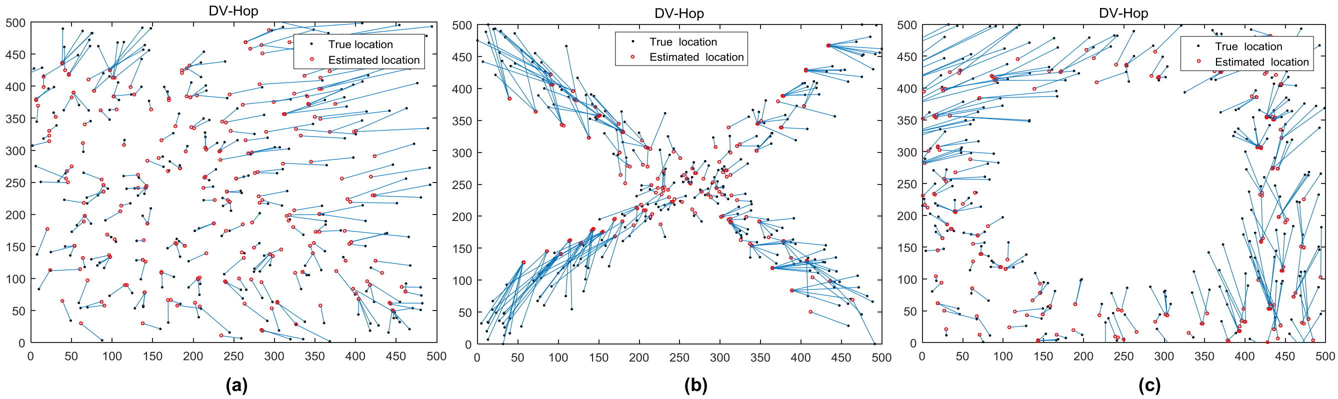

FIGURE 24. Localization result of DV-Hop.

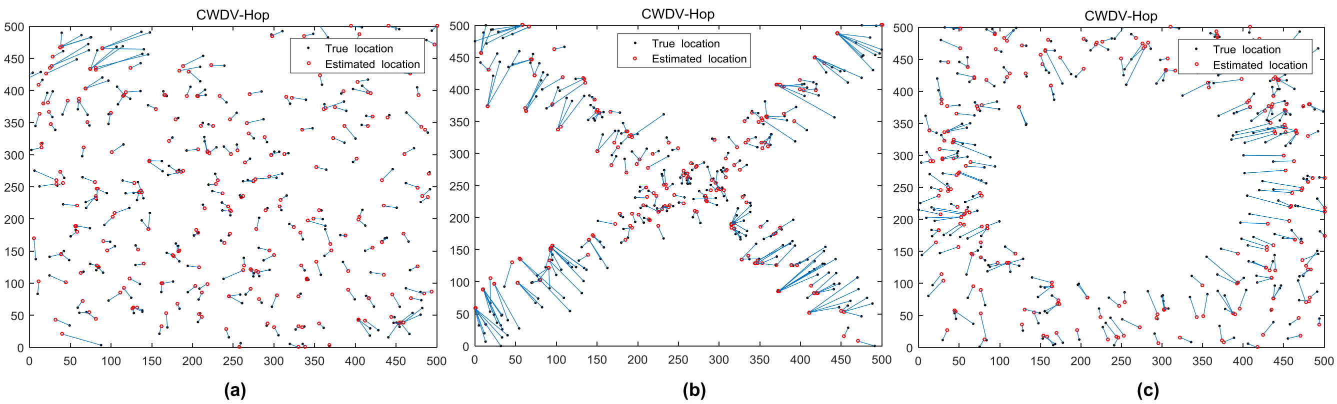

FIGURE 25. Localization result of CWDV-Hop.

are also $O(m * (n-m))$. In the last step of optimizing coordinates of unknown nodes, the time complexity of CDV-Hop and CWDV-Hop with CSO is $O(M1 * N * (n-m))$, that of iDV-Hop2 with PSO and that of iDV-Hop4 with DE are $O(M2 * P * (n-m))$ and $O(M3 * Q * (n-m))$.

In CSO, the time complexity generated by calculating the fitness value is $O(m * N)$. The processes to update the chickens' roles at interval of G require a time complexity of $O(M1/G)$. Meanwhile, the time complexity for updating the locations of roosters, hens and chicks is $O(M1 * N)$. In addition, the time complexity required for updating the current optimal value is $O(N)$. Because the maximum number of iterations and the size of bat group is larger than the other parameters, the maximum time complexity of CSO is $O(M1 * N)$. Therefore, the fourth step of CDV-Hop and CWDV-Hop with CSO has a time complexity of $O(M1 * N * (n-m))$.

Table 7 summarizes the time complexity of the eight algorithms. It is observed that WDV-Hop has the same complexity as the traditional method and the CWDV-Hop algorithm increases slightly in time complexity. In terms of space complexity, the temporary storage spaces occupied by the operational process of CWDV-Hop, iDV-Hop2 and iDV-Hop4 are correlated with the sizes of CSO, PSO and DE groups. Therefore, the space complexities of CWDV-Hop and iDV-Hop2 are $O(N)$, $O(P)$ and $O(Q)$.

VI. CONCLUSION

In this article, a novel CWDV-Hop algorithm is proposed. Considering the effect of distance on different anchors, CWDV-Hop replaces the average hop distance of traditional DV-Hop with distance-weighted hop distance. Then the coordinates of unknown nodes are obtained from two-dimensional hyperbolic approach. Those improvements based on the former are termed as WDV-Hop. The proof of WDV-Hop presents its simplicity in theory and experimental results show its superiority over the traditional DV-Hop. And then, Chicken Swarm Optimization (CSO) is applied to minimize the error of estimated coordinates. The localization accuracy of CWDV-Hop is improved in comparison with its counterparts. In addition, the introduction of CSO cause a slight but acceptable time complexity increase during the optimal operation. Theoretically, the proposed algorithm can be applied in both indoor and outdoor environment. However, the indoor environment is complex, and the signal propagation must face the problems of reflection, scattering and masking. This results in the inaccuracy localization. Therefore, the proposed CWDV-Hop is more suitable for the outdoor environment. Considering the resource constrains (e.g., small storage, low computation and energy) of sensor nodes in real-world applications, the localization algorithms using bio-inspired techniques can adopt the centralized network structure where the base station (BS) firstly collects data of each node through multi-hops and then implements the following localization processes. In the future, extending proposed scheme to the 3D space and the mobile WSN is our research direction.

ACKNOWLEDGMENT

Jiaxing Chen and Wei Zhang are co-first authors.

REFERENCES

[1] Z. Wang, H. Zhang, T. Lu, and T. A. Gulliver, "Cooperative RSS-based localization in wireless sensor networks using relative error estimation and semidefinite programming," *IEEE Trans. Veh. Technol.*, vol. 68, no. 1, pp. 483–497, Jan. 2019.

[2] X. Wu, J. Huang, J. Ling, and L. Shu, "BLTM: Beta and LQI based trust model for wireless sensor networks," *IEEE Access*, vol. 7, pp. 43679–43690, Mar. 2019.

[3] J. Kumari, P. Kumar, and S. K. Singh, "Localization in three-dimensional wireless sensor networks: A survey," *J. Supercomput.*, vol. 75, no. 8, pp. 5040–5083, Aug. 2019.

[4] A. Akbas, H. U. Yildiz, A. M. Ozbayoglu, and B. Tavli, "Neural network based instant parameter prediction for wireless sensor network optimization models," *Wireless Netw.*, vol. 25, no. 6, pp. 3405–3418, Aug. 2018.

[5] A. Kaur, P. Kumar, and G. P. Gupta, "Nature inspired algorithm-based improved variants of DV-hop algorithm for randomly deployed 2D and 3D wireless sensor networks," *Wireless Pers. Commun.*, vol. 101, no. 1, pp. 567–582, Apr. 2018.

[6] X. Yan, A. Song, Z. Yang, and W. Yang, "An improved multihop-based localization algorithm for wireless sensor network using learning approach," *Comput. Electr. Eng.*, vol. 48, pp. 247–257, Nov. 2015.

[7] F. Xu, H. Ye, F. Yang, and C. Zhao, "Software defined mission-critical wireless sensor network: Architecture and edge offloading strategy," *IEEE Access*, vol. 7, pp. 10383–10391, Jan. 2019.

[8] G. Sharma and A. Kumar, "Improved range-free localization for three-dimensional wireless sensor networks using genetic algorithm," *Comput. Electr. Eng.*, vol. 72, pp. 808–827, Nov. 2018.

[9] Y. Yao, H. Qi, X. Xu, and N. Jiang, "A RSSI-based distributed weighted search localization algorithm for WSNs," *Int. J. Distrib. Sens. Netw.*, vol. 11, no. 4, pp. 1–11, Apr. 2015.

[10] M. Burtowy, M. Rzymowski, and L. Kulas, "Low-profile ESPAR antenna for RSS-based DOA estimation in IoT applications," *IEEE Access*, vol. 7, pp. 17403–17411, Jan. 2019.

[11] J. Zhang, G. J. Han, N. Sun, and L. Shu, "Path-loss-based fingerprint localization approach for location-based services in indoor environments," *IEEE Access*, vol. 5, pp. 13756–13769, Jul. 2017.

[12] H. Chen, B. Liu, P. Huang, J. Liang, and Y. Gu, "Mobility-assisted node localization based on TOA measurements without time synchronization in wireless sensor networks," *Mobile Netw. Appl.*, vol. 17, no. 1, pp. 90–99, Feb. 2012.

[13] M. Shalaby, M. Shokair, and N. W. Messiha, "Performance enhancement of TOA localized wireless sensor networks," *Wireless Pers. Commun.*, vol. 95, no. 4, pp. 4667–4679, Aug. 2017.

[14] W. Meng, L. Xie, and W. Xiao, "Optimal TDOA sensor-pair placement with uncertainty in source location," *IEEE Trans. Veh. Technol.*, vol. 65, no. 11, pp. 9260–9271, Nov. 2016.

[15] Y. Sun, F. Zhang, and Q. Wan, "Wireless sensor network-based localization method using TDOA measurements in MPR," *IEEE Sensors J.*, vol. 19, no. 10, pp. 3741–3750, May 2019.

[16] B. Thoen, S. Wielandt, and L. De Strycker, "Improving AoA localization accuracy in wireless acoustic sensor networks with angular probability density functions," *Sensors*, vol. 19, no. 4, pp. 1–16, Feb. 2019.

[17] H.-J. Shao, X.-P. Zhang, and Z. Wang, "Efficient closed-form algorithms for AOA based self-localization of sensor nodes using auxiliary variables," *IEEE Trans. Signal Process.*, vol. 62, no. 10, pp. 2580–2594, May 2014.

[18] Y. Cao and Z. Wang, "Improved DV-hop localization algorithm based on dynamic anchor node set for wireless sensor networks," *IEEE Access*, vol. 7, pp. 124876–124890, Aug. 2019.

[19] Rekha, G. Kumar, and M. K. Rai, "An advanced DV-hop localization algorithm for random mobile nodes in wireless sensor networks," *Arabian J. Sci. Eng.*, vol. 44, no. 11, pp. 9787–9803, Aug. 2019.

[20] J. Liu, Z. Wang, M. Yao, and Z. Qiu, "VN-APIT: Virtual nodes-based range-free APIT localization scheme for WSN," *Wireless Netw.*, vol. 22, no. 3, pp. 867–878, Jul. 2015.

[21] K. Y. Kim and Y. Shin, "A distance boundary with virtual nodes for the weighted centroid localization algorithm," *Sensors*, vol. 18, no. 4, pp. 1–10, Apr. 2018.

[22] X. P. Zhao, J. N. Ma, R. Rao, and J. Gong, "Improvement of DV-hop localization algorithm based on hop distance weighting," *Comput. Technol. Devices*, vol. 28, no. 3, pp. 83–92, May 2018.

[23] J. Yang, Y. Cai, D. Tang, and Z. Liu, "A novel centralized range-free static node localization algorithm with memetic algorithm and Lévy flight," *Sensors*, vol. 19, no. 14, pp. 1–30, Jul. 2019.

[24] J. Mass-Sanchez, E. Ruiz-Ibarra, J. Cortez-González, A. Espinoza-Ruiz, and L. A. Castro, "Weighted hyperbolic DV-hop positioning node localization algorithm in WSNs," *Wireless Pers. Commun.*, vol. 96, no. 4, pp. 5011–5033, Oct. 2017.

[25] J. Wang, A. Hou, Y. Tu, and H. Yu, "An improved DV-hop localization algorithm based on selected anchors," *Comput., Mater. Continua*, vol. 65, no. 1, pp. 977–991, 2020.

[26] W. Zhao, S. Su, and F. Shao, "Improved DV-hop algorithm using locally weighted linear regression in anisotropic wireless sensor networks," *Wireless Pers. Commun.*, vol. 98, no. 4, pp. 3335–3353, Oct. 2017.

[27] G. Song and D. Tam, "Two novel DV-hop localization algorithms for randomly deployed wireless sensor networks," *Int. J. Distrib. Sens. Netw.*, vol. 11, no. 7, pp. 1–9, Jan. 2015.

[28] X. Cai, P. Wang, L. Du, Z. Cui, W. Zhang, and J. Chen, "Multi-objective three-dimensional DV-hop localization algorithm with NSGA-II," *IEEE Sensors J.*, vol. 19, no. 21, pp. 10003–10015, Nov. 2019.

[29] T. Najeh, H. Sassi, and N. Liouane, "A novel range free localization algorithm in wireless sensor networks based on connectivity and genetic algorithms," *Int. J. Wireless Inf. Netw.*, vol. 25, no. 1, pp. 88–97, Mar. 2018.

[30] L. Cui, C. Xu, G. Li, Z. Ming, Y. Feng, and N. Lu, "A high accurate localization algorithm with DV-hop and differential evolution for wireless sensor network," *Appl. Soft Comput.*, vol. 68, pp. 39–52, Jul. 2018.

[31] D. Han, Y. Yu, K. Li, and R. Mello, "Enhancing the sensor node localization algorithm based on improved DV-Hop and DE algorithms in wireless sensor networks," *Sensors*, vol. 20, no. 343, pp. 1–24, Oct. 2020.

[32] Y. L. Liu, X. M. Yu, and Y. H. Hao, "Wireless sensor node localization algorithm based on particle swarm optimization and quantum neural," *Int. J. Online Eng.*, vol. 14, no. 10, pp. 230–240, 2018.

[33] S. P. Singh and S. C. Sharma, "A PSO based improved localization algorithm for wireless sensor network," *Wireless Pers. Commun.*, vol. 98, no. 1, pp. 487–503, Jan. 2018.

[34] Q. Shi, Q. Xu, and J. Zhang, "An improved DV-hop scheme based on path matching and particle swarm optimization algorithm," *Wireless Pers. Commun.*, vol. 104, no. 4, pp. 1301–1320, Feb. 2019.

[35] D. Xue, "Research on range-free location algorithm for wireless sensor network based on particle swarm optimization," *EURASIP J. Wireless Commun. Netw.*, vol. 2019, no. 1, pp. 1–8, Sep. 2019.

[36] S. Phoemphon, C. So-In, and N. Leelathakul, "Optimized hop angle relativity for DV-hop localization in wireless sensor networks," *IEEE Access*, vol. 6, pp. 78149–78172, Dec. 2018.

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

[38] Z. Cui, B. Sun, G. Wang, Y. Xue, and J. Chen, "A novel oriented cuckoo search algorithm to improve DV-hop performance for cyber–physical systems," *J. Parallel Distrib. Comput.*, vol. 103, pp. 42–52, May 2017.

[39] X. Yu and M. Hu, "Hop-count quantization ranging and hybrid cuckoo search optimized for DV-HOP in WSNs," *Wireless Pers. Commun.*, vol. 108, no. 4, pp. 2031–2046, May 2019.

[40] X. Yang and W. Zhang, "An improved DV-hop localization algorithm based on bat algorithm," *Cybern. Inf. Technol.*, vol. 16, no. 1, pp. 89–98, Mar. 2016.

[41] Y. Liu, J. J. Chen, and Z. F. Xu, "Improved DV-hop localization algorithm based on bat algorithm in wireless sensor networks," *KSII Trans. Internet Inf. Syst.*, vol. 11, no. 1, pp. 215–236, Jan. 2017.

[42] X. J. Cai, S. Geng, P. Wang, L. Wang, and Q. Wu, "Fast triangle flip bat algorithm based on curve strategy and rank transformation to improve DV-Hop performance," *KSII Trans. Internet Inf. Syst.*, vol. 13, no. 12, pp. 580–5785, Dec. 2019.

[43] V. Kanwar and A. Kumar, "DV-hop based localization methods for additionally deployed nodes in wireless sensor network using genetic algorithm," *J. Ambient Intell. Humanized Comput.*, vol. 11, no. 11, pp. 5513–5531, Mar. 2020.

[44] W. Zhao, F. Shao, S. Ye, and W. Zheng, "LSRR-LA: An anisotropy-tolerant localization algorithm based on least square regularized regression for multi-hop wireless sensor networks," *Sensors*, vol. 18, no. 11, pp. 1–17, Nov. 2018.

[45] F. Shahzad, T. R. Sheltami, and E. M. Shakshuki, "DV-maxHop: A fast and accurate range-free localization algorithm for anisotropic wireless networks," *IEEE Trans. Mobile Comput.*, vol. 16, no. 9, pp. 2494–2505, Sep. 2017.

[46] M. Al Shayokh and S. Y. Shin, "Bio inspired distributed WSN localization based on chicken swarm optimization," *Wireless Pers. Commun.*, vol. 97, no. 4, pp. 5691–5706, Aug. 2017.

JIAXING CHEN received the Ph.D. degree in signal and information processing from the Harbin Institute of Technology. He is currently the Director of the Academic Affairs Office with Hebei Normal University, China. His current research interests include computer networks and wireless sensor networks.

WEI ZHANG is currently pursuing the Ph.D. degree with the School of Electronic and Information Engineering, Hebei University of Technology, China. Her current research interests include localization in wireless networks, signal processing, and deep learning.

ZHIHUA LIU received the M.S. degree from Yanshan University, China. She is currently a Professor with the College of Computer and Cyber Security, Hebei Normal University. Her current research interests include wireless and secure localization.

RUI WANG (Member, IEEE) received the Ph.D. degree in computer applied technology from the Ocean University of China. She is currently a Lecturer with the School of Opto-Electronic Information Science and Technology, Yantai University, China. Her research interests include computer vision and signal processing.

SHUJING ZHANG received the Ph.D. degree from the College of Information Science and Engineering, Ocean University of China. Her research interests include SLAM for AUVs, computer vision, signal processing, underwater acoustic networks, and localization in wireless sensor networks.