ISSN 1330-3651 (Print), ISSN 1848-6339 (Online) https://doi.org/10.17559/TV-20220219093303
Preliminary communication

Iterative Virtual Force Localization Based on Anchor Selection for Three-Dimensional Wireless Sensor Networks

Yun HONG, Song WANG*, Huibing KANG, Yanzhu HU

Abstract: Network localization is an emerging paradigm that enables high-accuracy location awareness in global positioning system (GPS)-challenged environments. The existing localization methods suffer from low accuracy, inflexible node distribution, and high hardware cost, especially in three-dimensional (3D) environment. To develop an efficient 3D localization method, this paper proposes the iterative virtual force (IVF) localization based on anchor selection. To eliminate the distance estimation error of range-free measurement, the K-means clustering (KMC) was introduced to exclude anchors with distance outliers, forming a set of selected anchors, which is enclosed as a virtual space. Based on the centroid of the virtual space, a drifting coefficient was defined, and the balance point of virtual force was deduced. The centroid was drifted to the balance point, and used to replace the anchor with the farthest distance to agent, producing a new set of selected anchors. In this way, the IVF method iteratively localizes the agent. In addition, the authors configured the threshold of change rate, and proved the condition of faster convergence. Simulation results show that the IVF method excels in localization accuracy, hardware cost, and computing complexity. Our research widens the application scope and improves the robustness of IVF localization in complex 3D WSNs.

Keywords: iterative virtual force; K-means clustering (KMC); network localization; three-dimensional (3D) wireless sensor networks (WSNs)

1 INTRODUCTION

Wireless sensor networks (WSNs) are composed by randomly deployed tiny, low-power nodes, capable of sensing, processing, and communication [1-2]. Each WSN consists of anchors and agents. The anchors can measure positions themselves by global positioning system (GPS) or manual configuration, while the agents can estimate positions based on the information from the anchors [3]. Most WSN applications, ranging from environment monitoring [4], security monitoring [5], to military defense [6], require the knowledge of positions of sensors. For example, sensors like unmanned aerial vehicles (UAVs) can scatter in unknown and harsh environment to monitor the temperature or humidity, and issue warnings of fire. If the sensor positions are unknown, the monitoring data would have little value in information processing [7, 8]. In special cases like mountain and underwater monitoring, two-dimensional (2D) localization is insufficient, and must be supported by three-dimensional (3D) information [9-10]. The additional third dimension greatly affects the localization accuracy. Since many applications need 3D localization, it is important to develop efficient self-localization algorithms for 3D WSN nodes [11, 12].

The early localization algorithms for WSNs focus on researching and improving basic 2D localization. The localization algorithms are either range-based or range-free [13, 14], depending on the absence/presence of node distance measurement. Range-based methods, such as direction-of-arrival (DOA) [15], time-of-arrival (TOA) [16], and time-difference-of-arrival (TDOA) [17], measure the distance between nodes with physical devices. They can achieve a high measuring accuracy at a high hardware cost. Meanwhile, range-free methods, such as approximate point-in-triangulation (APIT) [18, 19], Centroid [20, 21], Amorphous, and distance vector (DV)-hop [22], drive node relationship solely based on connectivity information. Compared with range-based methods, the range-free methods are relatively inaccurate, but save hardware cost largely, because they do not need physical devices. The above methods have been widely used in 2D localization of WSNs. Nevertheless, none of them could work accurately in some 3D environments, where the additional dimension significantly influences node distance and spatial position.

Extended from 2D localization, the 3D localization of WSNs faces a high energy consumption, complex computing, and strict requirements on network connectivity. Among the few studies on 3D localization, some simply extend 2D localization methods, namely, 3D-DV-hop [23], 3D-DV-Distance [24], 3D-Centriod [25], etc. Due to the additional dimension, the extended approaches are not always suitable for 3D localization, as evidenced by their poor localization performance. To solve the problem, many researchers explored the spatial planning of 3D localization. For instance, D. Niculescu [26] proposed the APIS algorithm, which treats the partition space as the spherical shell and anchor as spherical center; together, the multiple spheres make up a 3D space; each agent finds the local spherical shell, and regards the gravity center of intersecting shells as the position. Liang [27] presented a bottom-up localization method: the sensors are clustered in all directions, and the original anchors are assumed on the bottom plane. Shu [28] combined DV-hop with artificial bee colony (ABC) into the DHA localization algorithm: the node distribution space is divided by ABC, and the position is then determined by the possible distribution space and average hop distance. Rui [29] proved that there is at least one connected anchor, whose distance from the unknown node must be farther than the virtual anchor, and improved the iterative centroid estimation by reducing the space enclosed of connected virtual anchors. Based on 3D spatial planning, the above methods only apply to specific node distributions. Given the random distribution of nodes in most real-world WSNs, these methods can only be applied to a very limited scope in 3D environments.

Mobile anchor-based localization is a trend of 3D WSNs. Zhang [30] proposed the Landscape-3D algorithm, treating it as a functional dual of target tracking: The location assistant periodically broadcasts its position, and estimates the distance by received signal strength indicator (RSSI); the information is adopted to predict node state, and iteratively update the un-scented Kalman filter, thereby achieving 3D localization. Ou [31, 32] developed a 3D localization scheme with flying anchors: Once the

sensing space is covered by flying anchors, the anchor with a GPS receiver broadcasts its location; following basic geometry principles, the flying anchors will design the flying path, and estimate the agent position. This scheme eliminates the need for sensor interactions, and remains independent of network densities and topologies. Nonetheless, the mobile anchors must design their paths carefully as to cover all the nodes. In some cases, it is impossible to realize the path design, due to limitations of environment and response time. If so, the localization would be rather inaccurate.

Some researchers explored 3D localization with the aid of artificial intelligence (AI). To solve leakage location problem, Kang et al. [33] integrated a one-dimensional (1D) convolutional neural network (CNN) with support vector machine (SVM) into a graph-based localization algorithm, in which the agents estimate positions by comparing the actual measured signals with the virtually generated signals, and developed a leakage detection system with an adaptive design. Sharma et al. [34, 35] introduced genetic algorithm (GA) to improve localization accuracy: the correction factor is updated by 3D genetic localization, the hop size is modified by distance vector hop, and coplanarity is utilized to address the coplanar anchor problem. The AI algorithms improve the accuracy of 3D localization, but their computing complexity is too high. The high computing load hinders the application of AI in 3D localization.

The existing approaches of 3D localization face problems like inflexible node distribution, high hardware cost, and low accuracy. To overcome these problems, this paper proposes the iterative virtual force (IVF) localization based on anchor selection. The contributions of this paper can be summarized as follows.

(1) Considering the inaccurate distance measurement, we propose a 3D k-means clustering (KMC) to identify the anchors with precise distance, and rank the optimal information for IVF localization.

(2) Improved by centroid of the enclosed space of selected anchors, we propose a virtual force localization, and derive the virtual force formula to determine the balance point.The balance point is analogy to the centroid of the enclosed space by drifting virtual space. The anchor with the farthest distance to agent was replaced by the balance point, forming a new set of selected anchors. In this way, the centroid of the set of selected anchors and its balance point were updated iteratively. After reaching the termination condition, the balance point was regarded as the estimated position of agent.

The rest of the paper is organized as follows. Section II presents the anchor selection method based on 3D KMC, and designs the IVF localization. Numerical results are given in Section III, and conclusions are drawn in the last section.

2 LOCALIZATION PROCESS

2.1 Anchor Selection Based on 3D KMC

Suppose there exists a WSN of $M = N_a + N_u$ nodes randomly distributed in the fixed space, including N_a anchors and N_u agents. The position of the *m*-th sensor is described as $\boldsymbol{\theta}_m = [x_m, y_m], m \in A \cup U$. The set of anchors and the set of agents are denoted by $A = \{1, \cdots, N_a\}$ and $U = \{N_a + 1, \cdots, M\}$, respectively. The wireless sensor network is assumed to be uniform distribution.

For 3D WSNs, range-free methods have a large computing load and a low accuracy in distance estimation. Meanwhile, the range-based methods incur a high cost in distance measurement, due to the necessity of using extra physical devices. The RSSI is the only computationally efficient method without needing extra devices.

Step 1: distance estimation by multi-hop method

For 3D localization in large WSNs, adequate anchors may not be within the communication range of an arbitrary agent. But the localization of one agent needs at least four anchors. To overcome the low anchor density, a multi-hop method was introduced to estimate node distance. The agent-anchor distance can be defined as the sum of hop distance in the shortest path:

$$d_{ij} \approx \sum_{l=1}^{n_h} h_l \tag{1}$$

where, d_{ij} is the distance between nodes *i* and *j*; h_l is the distance of the *l*-th hop in the shortest path between nodes *i* and *j*; n_h is the total hops of the shortest path between nodes *i* and *j*.

Step 2: multilateration by least square method

Despite solving low anchor density, the multi-hop method may result in distance estimation error for several reasons. Firstly, the current path loss model cannot characterize the relationship between node distance and RSSI, and the non-line of sight (NLOS) for signal negatively affects the distance measurement. Secondly, the shortest path is not necessarily the linear distance between two nodes, which leads to a random distance error. In addition, the anchors of the whole network are redundant to form the IVF model for localization. Therefore, 3D KMC was performed to identify the anchors with relatively accurate node distance.

We should notice that the coordinates of agents and anchors are quadratic in equations, so one more anchor should usually be utilized to determine the only position of agent. For 3D localization, at least four anchors and their distances to agents are required for multilateration (MLAT). The geometric model of 3D distribution can be expressed as:

$$\begin{cases} (x_1 - x)^2 + (y_1 - y)^2 + (z_1 - z)^2 = d_1^2 \\ \vdots \\ (x_n - x)^2 + (y_n - y)^2 + (z_n - z)^2 = d_n^2 \end{cases} \tag{2}$$

where, $(x_i, y_i, z_i), i = 1, 2 \cdots n$ are the coordinates of anchor *i*; (x, y, z) are the coordinates of an agent; $d_i, i = 1, 2, \cdots, n$ is the distance between the agent and anchor *i*.

The least squares method was introduced to simplify the above equations by subtracting the last equation with first $n - 1$ equations in turn:

$$\hat{\boldsymbol{X}} = \left(\boldsymbol{H}^{\mathrm{T}} \boldsymbol{H}\right)^{-1} \boldsymbol{H}^{\mathrm{T}} \boldsymbol{B} \tag{3}$$

where, $\boldsymbol{H}$, $\boldsymbol{X}$, and $\boldsymbol{B}$ are matrices:

$$\boldsymbol{H}=\begin{bmatrix} 2x_1-2x_n & 2y_1-2y_n & 2z_1-2z_n \\ \vdots & \vdots & \vdots \\ 2x_{n-1}-2x_n & 2y_{n-1}-2y_n & 2z_{n-1}-2z_n \end{bmatrix};\boldsymbol{X}=\begin{bmatrix} x \\ y \\ z \end{bmatrix};$$
$$\boldsymbol{B}=\begin{bmatrix} x_1^2-x_n^2+y_1^2-y_n^2+z_1^2-z_n^2+d_n^2-d_1^2 \\ \vdots \\ x_{n-1}^2-x_n^2+y_{n-1}^2-y_n^2+z_{n-1}^2-z_n^2+d_n^2-d_{n-1}^2 \end{bmatrix} \quad (4)$$

Step 3: Clusting by k-means method

The MLAT regards four anchors as one group. Each group locates a position $\boldsymbol{X}=\{(x_u,y_u,z_u)\}$ of the agent. In total, there are $C_{N_a}^4$ groups for N_a anchors. Thus, the MLAT locates $C_{N_a}^4$ positions of all agents. Because the node distance is inaccurate, these $C_{N_a}^4$ positions do not coincide with each other. Fig. 1 shows the dispersion of estimated positions. The red dots in Fig. 1 represent the estimated positions of agents to be located. The axis of the coordinates is the x,y,z direction of the estimated positions. It can be observed that most estimated positions are around the true position, and a few are scattered in other places. The center of estimated positions is close to the true position. Thus, the KMC was introduced to cluster the estimated positions.

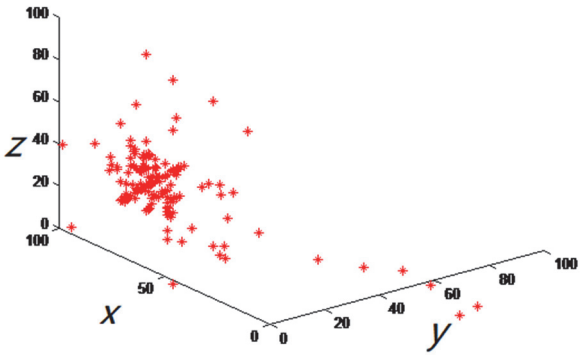

Figure 1 Distribution of estimated positions

Step 4: Ranking of the anchors for IVF localization

The dispersion of estimated locations are assessed by the Euclidean distance of K-means Clusting. Taking the $C_{N_a}^4$ estimated positions as clustering samples, it is assumed that the number of clusters is κ. Hence, there are κ cluster centers. From the $C_{N_a}^4$ estimated positions, κ initial cluster centers $C=\{(c_{tx},c_{ty},c_{tz})|t=1,2\cdots,\kappa\}$ are selected randomly, leaving $C_A^4-\kappa$ samples behind. In turn, the remaining samples calculate their Euclidean distance $\boldsymbol{Q}=\left\{q_{st}\middle|s=1,2,\cdots,\left(C_{N_a}^4-\kappa\right);t=1,2,\cdots,\kappa\right\}$ with the κ initial cluster centers, and then compare the calculated results. A sample belongs to the cluster with the shortest Euclidean distance. In this way, all samples form clusters $\boldsymbol{W}_1,\boldsymbol{W}_2,\cdots,\boldsymbol{W}_k$. Each cluster $\boldsymbol{W}_t$ consists of a cluster center and multiple samples. The mean of these samples becomes the new center C_t^* of cluster $\boldsymbol{W}_t$. Then, the Euclidean distances are calculated again, each sample finds its closest cluster center C^*, and the clustering results are updated. The above process is repeated until the cluster center changes smaller than the threshold. The threshold depends on the localization accuracy. High localization accuracy requires smaller value of the threshold. Meanwhile, convergence speed also affects the threshold, so its value also depends on the requirment of computation time.

Among the clusters, the cluster $\boldsymbol{W}_{\max}$ with most samples is the most likely to fall on the true position of the agent. Correspondingly, the cluster $\boldsymbol{W}_{\max}$ records the classified anchors. The cluster with accurate positions ranks the effective anchors by the number of occurrences. Also, the clustering κ is also affected by the accuracy of the distance measurement. Its determination depends on the environment of signal and noise. Finally, the top N_A anchors $\hat{A}_1,\hat{A}_2,\cdots,\hat{A}_{N_A}$ are selected for IVF localization.

2.2 IVF Localization

The 3D localization methods, which are extended from 2D localization, face a high complexity and a low accuracy, owing to the additional dimension. Among the various extended methods, 3D-centroid boasts the lowest computing complexity, and the slackest hardware demand. The centroid localization is hardly impacted by the additional dimension in 3D environments. Therefore, 3D-centroid is very suitable for 3D localization of large WSNs. That is why our method takes 3D-centroid as the basis [29]. The 3D-centroid replaces the farthest anchor of the enclosed space by another anchor, and shrinks the anchor space iteratively until the centroid approaches the true position. The premise is that the agent must be in a space enclosed by anchors, which can be tested through APIT. However, this premise does not always hold in reality, and APIT is difficult to implement. IVF is a method which can identify the space of agent located and avoid being beyond the boundary. Therefore, this paper replaces centroid with IVF method.

In a given 3D WSN, the true position of agent is denoted as (x,y,z), and the selected anchors as $\hat{A}_1,\hat{A}_2,\cdots,\hat{A}_{N_A}$. Hence, the distance between an agent and the *n*-th anchor can be expressed as:

$$d_{iU}=\sqrt{(x_i-x)^2+(y_i-y)^2+(z_i-z)^2},i\in\hat{A} \quad (5)$$

where, $\theta_i=(x_i,y_i,z_i)$ are the coordinates of anchor $\hat{A}_i$; $\hat{A}$ is the set of selected anchors.

Because the coordinates of the selected anchors are already known, the centroid of the space enclosed by the selected anchors can be described as:

$$\begin{aligned} x_{\hat{o}}^{k-1} &= \frac{1}{N_A}\sum_{i\in\hat{A}} x_i^{k-1} \\ y_{\hat{o}}^{k-1} &= \frac{1}{N_A}\sum_{i\in\hat{A}} y_i^{k-1} \\ z_{\hat{o}}^{k-1} &= \frac{1}{N_A}\sum_{i\in\hat{A}} z_i^{k-1} \end{aligned} \tag{6}$$

where, $\theta_{\hat{o}}^{k-1} = \left(x_{\hat{o}}^{k-1}, y_{\hat{o}}^{k-1}, z_{\hat{o}}^{k-1}\right)$ is the centroid position of the $(k-1)$-th iteration; $\theta_i^{k-1} = \left(x_i^{k-1}, y_i^{k-1}, z_i^{k-1}\right)$ is the ith selected anchor position of the $(k-1)$-th iteration; N_A is the number of selected anchors.

The agent to be located may not be in the space enclosed by anchors. If so, the centroid would be far from the true position of the agent, which impedes the application of 3D-centroid method. Without being limited by the enclosed space, the virtual force can characterize the relationship between agent and anchor, and estimate the position of the agent as initialized by centroid. With a single virtual force, the optimization problem can be depicted in the way of least squares, but the effect of node distance will be ignored. To make matters worse, a long node distance brings a large localization error, owing to the nonlinearity in the shortest path and uncertain path loss. To eliminate the negative effect of long distance, the virtual force can be derived from centroid in Eq. (6), which further reduces the enclosed space in the following iteration.

Taking centroid as the estimated position of agent, the position would be subjected to attraction and repulsion at the same time. The two forces can be converted from node distance. If the node distance is less than true distance, the centroid is under repulsion. If the node distance is greater than true distance, the centroid is under attraction. To balance the attraction and repulsion, the centroid affected by the *i*-th anchor can be replaced by:

$$\begin{aligned} x_{\hat{s}i}^{k} &= x_{\hat{o}}^{k-1} + \varepsilon\cdot\frac{x_i^{k-1}-x_{\hat{o}}^{k-1}}{\left\|\theta_{\hat{o}}^{k-1}-\theta_i^{k-1}\right\|}\cdot\left(\left\|\theta_{\hat{o}}^{k-1}-\theta_i^{k-1}\right\|-d_{iU}\right) \\ y_{\hat{s}i}^{k} &= y_{\hat{o}}^{k-1} + \varepsilon\cdot\frac{y_i^{k-1}-y_{\hat{o}}^{k-1}}{\left\|\theta_{\hat{o}}^{k-1}-\theta_i^{k-1}\right\|}\cdot\left(\left\|\theta_{\hat{o}}^{k-1}-\theta_i^{k-1}\right\|-d_{iU}\right) \\ z_{\hat{s}i}^{k} &= z_{\hat{o}}^{k-1} + \varepsilon\cdot\frac{z_i^{k-1}-z_{\hat{o}}^{k-1}}{\left\|\theta_{\hat{o}}^{k-1}-\theta_i^{k-1}\right\|}\cdot\left(\left\|\theta_{\hat{o}}^{k-1}-\theta_i^{k-1}\right\|-d_{iU}\right) \end{aligned} \tag{7}$$

where, $\theta_{\hat{s}i}^{k} = \left(x_{\hat{s}i}^{k}, y_{\hat{s}i}^{k}, z_{\hat{s}i}^{k}\right)$ is the balance position for the *i*-th anchor; $0 \le \varepsilon \le 1$ is the balance coefficient. d_{iU} is the measured distance between the *i*-th anchor and the agent.

For all anchors, the balance position of virtual force can be expressed as:

$$\begin{aligned} x_{\hat{s}}^{k} &= \frac{1}{N_A}\sum_{i\in\hat{A}} x_{Fi}^{k-1} \\ y_{\hat{s}}^{k} &= \frac{1}{N_A}\sum_{i\in\hat{A}} y_{Fi}^{k-1} \\ z_{\hat{s}}^{k} &= \frac{1}{N_A}\sum_{i\in\hat{A}} z_{Fi}^{k-1} \end{aligned} \tag{8}$$

Eq. (8) can be transformed to:

$$\begin{aligned} x_{\hat{s}}^{k} &= \frac{1}{N_A}\sum_{i\in\hat{A}}\left(x_i^{k-1} + \varepsilon\cdot\frac{x_i^{k-1}-x_{\hat{o}}^{k-1}}{\left\|\theta_{\hat{o}}^{k-1}-\theta_i^{k-1}\right\|}\cdot\left(\left\|\theta_{\hat{o}}^{k-1}-\theta_i^{k-1}\right\|-d_{iU}\right)\right) \\ y_{\hat{s}}^{k} &= \frac{1}{N_A}\sum_{i\in\hat{A}}\left(y_i^{k-1} + \varepsilon\cdot\frac{y_i^{k-1}-y_{\hat{o}}^{k-1}}{\left\|\theta_{\hat{o}}^{k-1}-\theta_i^{k-1}\right\|}\cdot\left(\left\|\theta_{\hat{o}}^{k-1}-\theta_i^{k-1}\right\|-d_{iU}\right)\right) \\ z_{\hat{s}}^{k} &= \frac{1}{N_A}\sum_{i\in\hat{A}}\left(z_i^{k-1} + \varepsilon\cdot\frac{z_i^{k-1}-z_{\hat{o}}^{k-1}}{\left\|\theta_{\hat{o}}^{k-1}-\theta_i^{k-1}\right\|}\cdot\left(\left\|\theta_{\hat{o}}^{k-1}-\theta_i^{k-1}\right\|-d_{iU}\right)\right) \end{aligned} \tag{9}$$

Then, the shifting variable can be defined as:

$$\begin{aligned} \Delta_{xi}^{k-1} &= \varepsilon\cdot\frac{x_i^{k-1}-x_{\hat{o}}^{k-1}}{\left\|\theta_{\hat{o}}^{k-1}-\theta_i^{k-1}\right\|}\cdot\left(\left\|\theta_{\hat{o}}^{k-1}-\theta_i^{k-1}\right\|-d_{iU}\right) \\ \Delta_{yi}^{k-1} &= \varepsilon\cdot\frac{y_i^{k-1}-y_{\hat{o}}^{k-1}}{\left\|\theta_{\hat{o}}^{k-1}-\theta_i^{k-1}\right\|}\cdot\left(\left\|\theta_{\hat{o}}^{k-1}-\theta_i^{k-1}\right\|-d_{iU}\right) \\ \Delta_{zi}^{k-1} &= \varepsilon\cdot\frac{z_i^{k-1}-z_{\hat{o}}^{k-1}}{\left\|\theta_{\hat{o}}^{k-1}-\theta_i^{k-1}\right\|}\cdot\left(\left\|\theta_{\hat{o}}^{k-1}-\theta_i^{k-1}\right\|-d_{iU}\right) \end{aligned} \tag{10}$$

Then, the position can be transformed to:

$$\begin{aligned} x_{\hat{s}}^{k} &= \frac{1}{N_A}\sum_{i\in\hat{A}}\left(x_i^{k-1} + \Delta_{xi}^{k-1}\right) \\ y_{\hat{s}}^{k} &= \frac{1}{N_A}\sum_{i\in\hat{A}}\left(y_i^{k-1} + \Delta_{yi}^{k-1}\right) \\ z_{\hat{s}}^{k} &= \frac{1}{N_A}\sum_{i\in\hat{A}}\left(z_i^{k-1} + \Delta_{zi}^{k-1}\right) \end{aligned} \tag{11}$$

The effect of virtual force can be regarded as a new space enclosed by anchors, i.e., the virtual space. The anchor set corresponding to the virtual space is denoted as $S = \{1, \cdots, N_a\}$. When one vertex of virtual space is replaced by balance position, the anchor set corresponding to the virtual space is denoted as $\hat{S} = \{1, \cdots, N_a\}$. The anchor position of the virtual space can be described by $\theta_{\hat{s}i}^{k-1} = (x_{\hat{s}i}^{k-1}, y_{\hat{s}i}^{k-1}, z_{\hat{s}i}^{k-1}) = (x_i^{k-1} + \Delta_{xi}^{k-1}, y_i^{k-1} + \Delta_{yi}^{k-1}, z_i^{k-1} + +\Delta_{zi}^{k-1})$.

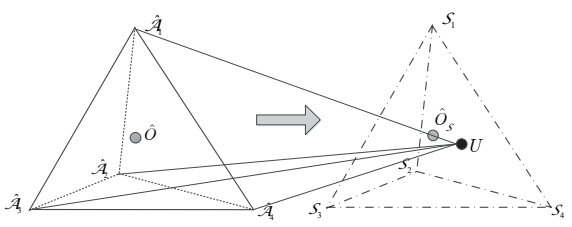

Figure 2 Enclosed space and virtual space

Correspondingly, the balance position of virtual force can be regarded as the centroid of the virtual space. This transformation is illustrated in Fig. 2, where $\hat{O}$ is the centroid of initial space enclosed by anchors, which is

shifted to the balance point by virtual force; $\hat{O}_S$, whose position is $\theta_{\hat{s}i}^k = \left(x_{\hat{s}}^k, y_{\hat{s}}^k, z_{\hat{s}}^k\right)$, is the balance point of virtual force, and also the centroid of virtual space. After the transformation, the balance point of virtual force gets closer to the agent. It can also be seen that the agent does not have to be in the space enclosed by anchors.

Hence, the distance between the centroid of the enclosed space and the agent can be expressed as:

$$d_{\hat{o}U}^{k-1} = \frac{1}{N_A}\sum_{i=1,i\in\hat{A}}^{N_A}\left(d_{iU}^{k-1}\right)^2 - \frac{1}{N_A^2}\sum_{i=1,i\in\hat{A}}^{N_A-1}\sum_{j=i+1,j\in\hat{A}}^{N_A}\left(d_{ij}^{k-1}\right)^2 \quad (12)$$

Analogically, the distance between the centroid of the virtual space and the agent can be expressed as:

$$d_{\hat{s}U}^{k} = \frac{1}{N_A}\sum_{i=1,i\in S}^{N_A}\left(d_{iU}^{k-1}\right)^2 - \frac{1}{N_A^2}\sum_{i=1,i\in S}^{N_A-1}\sum_{j=i+1,j\in S}^{N_A}\left(d_{ij}^{k-1}\right)^2 \quad (13)$$

Considering the shifting variable, the distance can be described by:

$$\begin{aligned} d_{\hat{s}U}^{k} = &\frac{1}{N_A}\sum_{i\in\hat{A}}\left(\left(x_i^{k-1}+\Delta_{xi}^{k-1}-x\right)^2 + \right.\\ &\left.+\left(y_i^{k-1}+\Delta_{yi}^{k-1}-y\right)^2+\left(z_i^{k-1}+\Delta_{zi}^{k-1}-z\right)^2\right)- \\ &-\frac{1}{N_A^2}\sum_{i=1,i\in S}^{N_A-1}\sum_{j=i+1,j\in S}^{N_A}\left(d_{ij}^{k-1}\right)^2 \end{aligned} \quad (14)$$

Eq. (14) can be transformed into:

$$\begin{aligned} d_{\hat{s}U}^{k} = &\frac{1}{N_A}\sum_{i\in\hat{A}}\left(\begin{array}{l}\left(d_{iU}^{k-1}\right)^2+\left(\Delta_{xi}^{k-1}\right)^2+\left(\Delta_{yi}^{k-1}\right)^2 \\ +\left(\Delta_{zi}^{k-1}\right)^2+2\Delta_{xi}^{k-1}\left(x_i^{k-1}-x\right)\end{array}\right. \\ &\left.+2\Delta_{yi}^{k-1}\left(y_i^{k-1}-y\right)+2\Delta_{zi}^{k-1}\left(z_i^{k-1}-z\right)\right)- \\ &-\frac{1}{N_A^2}\sum_{i=1,i\in S}^{N_A-1}\sum_{j=i+1,j\in S}^{N_A}\left(d_{ij}^{k-1}\right)^2 \end{aligned} \quad (15)$$

Then, the shifting coefficient can be defined as:

$$\lambda_i^{k-1} = \varepsilon\cdot\frac{\left(\left\|\theta_{\hat{o}}^{k-1}-\theta_i^{k-1}\right\|-d_{iU}\right)}{\left\|\theta_{\hat{o}}^{k-1}-\theta_i^{k-1}\right\|} \quad (16)$$

The distance between the centroid of the virtual space and the agent can be expressed as:

$$\begin{aligned} d_{\hat{s}U}^{k} = &\frac{1}{N_A}\sum_{i\in\hat{A}}\left(\left(d_{iU}^{k-1}\right)^2+\left(\lambda_i^{k-1}\right)^2\left(d_{i\hat{o}}^{k-1}\right)^2+\right. \\ &2\lambda_i^{k-1}\left(\left(x_i^{k-1}-x_{\hat{o}}^{k-1}\right)\left(x_i^{k-1}-x\right)+\right. \\ &\left.\left.+\left(y_i^{k-1}-y_{\hat{o}}^{k-1}\right)\left(y_i^{k-1}-y\right)+\left(z_i^{k-1}-z_{\hat{o}}^{k-1}\right)\left(z_i^{k-1}-z\right)\right)\right)- \\ &-\frac{1}{N_A^2}\sum_{i=1,i\in S}^{N_A-1}\sum_{j=i+1,j\in S}^{N_A}\left(d_{ij}^{k-1}\right)^2 \end{aligned} \quad (17)$$

where,

$$\begin{aligned} &\left(x_i^{k-1}-x_{\hat{o}}^{k-1}\right)\left(x_i^{k-1}-x\right)= \\ &=\left(x_i^{k-1}-x_{\hat{o}}^{k-1}\right)^2+\left(x_i^{k-1}-x\right)^2-\left(x-x_{\hat{o}}^{k-1}\right)^2 \end{aligned} \quad (18)$$

Thus, Eq. (17) can be simplified as:

$$\begin{aligned} d_{\hat{s}U}^{k} = &\frac{1}{N_A}\sum_{i\in\hat{A}}\left(\begin{array}{l}\left(d_{iU}^{k-1}\right)^2 \\ +\left(\lambda_i^{k-1}\right)^2\left(d_{i\hat{o}}^{k-1}\right)^2+\end{array}\right. \\ &\left.+2\lambda_i^{k-1}\left(\begin{array}{l}\left(d_{i\hat{o}}^{k-1}\right)^2 \\ +\left(d_{iU}^{k-1}\right)^2-\left(d_{\hat{o}U}^{k-1}\right)^2\end{array}\right)\right)- \\ &-\frac{1}{N_A^2}\sum_{i=1,i\in S}^{N_A-1}\sum_{j=i+1,j\in S}^{N_A}\left(d_{ij}^{k-1}\right)^2 \end{aligned} \quad (19)$$

Substituting Eq. (12) into Eq. (19), we have:

$$\begin{aligned} d_{\hat{s}U}^{k} = &\frac{1}{N_A}\sum_{i\in\hat{A}}\left(\left(d_{iU}^{k-1}\right)^2+\left(\lambda_i^{k-1}\right)^2\left(d_{i\hat{o}}^{k-1}\right)^2+\right. \\ &\left.+2\lambda_i^{k-1}\left(\left(d_{i\hat{o}}^{k-1}\right)^2+\left(d_{iU}^{k-1}\right)^2-\left(d_{iU}^{k-1}\right)^2+F_1\right)\right)- \\ &-\frac{1}{N_A^2}\sum_{i=1,i\in S}^{N_A-1}\sum_{j=i+1,j\in S}^{N_A}\left(d_{ij}^{k-1}\right)^2 \end{aligned} \quad (20)$$

where,

$$F_1 = \frac{1}{N_A^2}\sum_{i=1,i\in\hat{A}}^{N_A-1}\sum_{j=i+1,j\in\hat{A}}^{N_A}\left(d_{ij}^{k-1}\right)^2 \quad (21)$$

It can be proved that

$$F_1 = \frac{1}{N_A}\sum_{i=1,i\in\hat{A}}^{N_A}\left(d_{i\hat{o}}^{k-1}\right)^2 \quad (22)$$

Then

$$d_{\hat{s}U}^{k} = \frac{1}{N_A}\sum_{i\in\hat{A}}\left(\begin{array}{l}\left(d_{iU}^{k-1}\right)^2 \\ +\left(\lambda_i^{k-1}\right)^2\left(d_{i\hat{o}}^{k-1}\right)^2\end{array}+4\lambda_i^{k-1}\left(d_{i\hat{o}}^{k-1}\right)^2\right)-F_2 \quad (23)$$

where,

$$F_2 = \frac{1}{N_A^2}\sum_{i=1,i\in S}^{N_A-1}\sum_{j=i+1,j\in S}^{N_A}\left(d_{ij}^{k-1}\right)^2 \quad (24)$$

Thus, the distance can be obtained as:

$$\begin{aligned} d_{\hat{s}U}^{k} = &\frac{1}{N_A}\sum_{i\in\hat{A}}\left(\left(d_{iU}^{k-1}\right)^2+\right. \\ &\left.+\varepsilon^2\left(d_{iU}^{k-1}-d_{i\hat{o}}^{k-1}\right)\left(d_{iU}^{k-1}-\frac{\varepsilon+4}{\varepsilon}d_{io}^{k-1}\right)\right)-F_2 \end{aligned} \quad (25)$$

Unlike the iterative centroid, the IVF does not require the balance point to be closer to the agent. Once the balance point is determined in an iteration, it will replace the anchor the farthest from the agent, forming a new set of selected anchors. Then, further virtual force is developed under the new set.

Although the iterative force does not need the position relationship between the agent and the set of selected anchors, the distance in Eq. (25) assures faster convergence. The first term in Eq. (25) is equivalent to the first term in Eq. (12), which is about the operation of iterative centroid. The convergence could be sped up by:

$$\left(d_{iU}^{k-1}-d_{i\hat{o}}^{k-1}\right)\left(d_{iU}^{k-1}-\frac{\varepsilon+4}{\varepsilon}d_{io}^{k-1}\right)<0 \tag{26}$$

It indicates that:

$$d_{i\hat{o}}^{k-1}<d_{iU}^{k-1}<\frac{\varepsilon+4}{\varepsilon}d_{io}^{k-1} \tag{27}$$

To reduce computing complexity and computing time, at least one anchor should be within the range of $(\varepsilon+4)/\varepsilon\cdot d_{i\hat{o}}^{k-1}$. When the threshold $\varepsilon=1$, the virtual force has the greatest effect on localization result. But the range for the anchor set is small, if faster convergence is expected. When the balance coefficient ε = 0 is close to zero, the IVF degenerates to the iterative centroid. Thus, the range for the set of selected anchors becomes larger, and would be reduced through iterative centroid. In this case, the localization performance is suppressed by the absence of APIT.

Let $\hat{A}_1,\hat{A}_2,\cdots,\hat{A}_{N_A}$ be the set of selected anchors in the k-th iteration. Then, the distance between the selected anchors and the agent meets the distance condition:

$$0<d_{1U}\le d_{2U}\le d_{3U}\cdots\le d_{(N_A-1)U}\le d_{N_AU} \tag{28}$$

Once the balance point $\hat{O}_S$ is determined, it will replace the anchor $\hat{A}_{N_A}$ to form a new set $\hat{A}_1,\hat{A}_2,\cdots,\hat{A}_{N_A},\hat{O}_S$. In this way, a new centroid is obtained and the balance point is updated iteratively. The iterative process goes on until meeting the termination condition of the localization algorithm:

$$d_{\hat{s}U}^{k}<\delta_1 \tag{29}$$

where, $d_{\hat{s}U}^{k}$ is the distance between the balance point and the agent; δ_1 is the threshold of the termination condition.

Because RSSI-based distance measurement is bound to have an error, the balance point cannot approach the true position of the agent. To avoid the infinite loop, the termination condition can also be defined as:

$$d_{\hat{s}U}^{k}-d_{\hat{s}U}^{k+1}<\delta_2 \tag{30}$$

where, $d_{\hat{s}U}^{k}$ and $d_{\hat{s}U}^{k+1}$ are the k-th and (k + 1)-th distance between the balance point and the agent; δ_2 is the threshold of the termination condition. When the change rate of distance falls below the threshold, the iteration ends.

2.3 Procedureof IVF Localization of 3D KMC

Procedure of IVF Localization of 3D KMC is shown as follows.

IVF Localization of 3D KMC
Input:
Agent set *A* and anchor set *U*.
Hop distance *h*.
Output:
Agent position θ.
Method:
1: Computing node distance d;
2: for each i = 1, 2, …, Na – 4 do
3: multilateration localization by least squares method
4: end for
5: KMC and Ranking
6: Initialize the centroid
7: for k = 1, 2, … do
8: Computing shifting variable
9: Computing balance position
9: Building virtual space
10: estimating distance of virtual space' vertex
11: end for
12: Return balance position

3 DISTRIBUTED COOPERATIVE LOCALIZATION SCHEME

This section evaluates the performance of the IVF method through MATLAB simulation. The simulation process and results are described in details.

3.1 Simulation Environment

The simulation was conducted in a 3D space (size: 100 × 100 × 100 m) with a low signal interference. The distribution of agents and anchors in the 3D space is illustrated in Fig. 3, where the red points represent the anchors, and the black points represent the agents. The positions of agents and anchors are randomly generated in each simulation, so the nodes are not fixed for localization. Except for eight anchors are deployed in the vertex of 3D space, it aims to make sure that all agents can be localized.

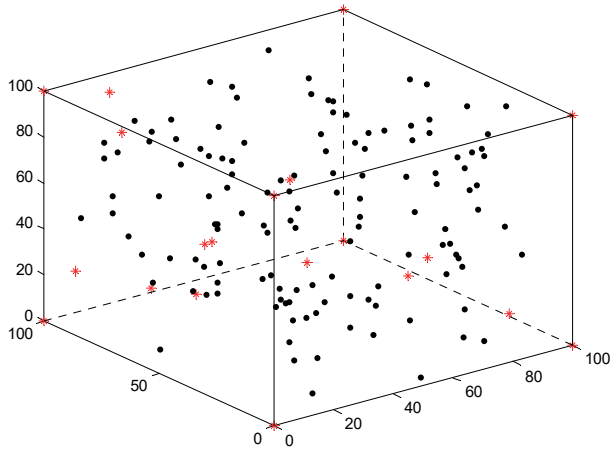

Figure 3 Distribution of sensors in the 3D space

In the basic scenario, the total number of sensors was set to 200, including 10% of anchors, and 90% of agents. The communication radius was configured as 50 m. During the simulation, the tested parameter was changed, while the simulation condition and the other parameters remained unchanged. For example, when the simulation aims to measure the influence of the anchor proportion, the proportion of anchors was adjusted from 10% to 50%, while the total number of nodes was fixed at 200, and the communication radius at 50 m. Tab. 1 lists the simulation parameters in the basic scenario. Each simulation was repeated 30 times, and the mean of the repeated simulation results was taken as the final result.

Table 1 Simulation parameters

Parameter	Value
Space size	100 × 100 × 100 m
Total number of nodes	200
Proportion of anchors	10%
Communication radius	50 m
Cluster number	5
Number of selected anchors	4

The localization performance was measured by the average localization error (ALE):

$$ALE = \frac{1}{N_u} \sum_{i=1, i \in U}^{N_u} \sqrt{(\hat{x}_i - x_i)^2 + (\hat{y}_i - y_i)^2 + (\hat{z}_i - z_i)^2} \quad (31)$$

where, $(\hat{x}_i, \hat{y}_i, \hat{z}_i)$ is the estimated position of the i-th agent; (x_i, y_i, z_i) is the true position of the i-th agent; N_u is the total number of agents.

Apart from ALE, cumulative distribution functions (CDFs) were also employed to evaluate distribution of localization errors. The faster the CDFs of ALE converge to 1, the better the localization effect.

3.2 Algorithm Parameters

The parameter configuration could affect the localization performance of the IVF method. To disclose the effect, several scenarios were simulated with different parameters: cluster number κ, number of selected anchors $\hat{A}$, and number of iterations k. The most appropriate algorithm parameters must be selected to ensure the positioning accuracy and reduce the computing load.

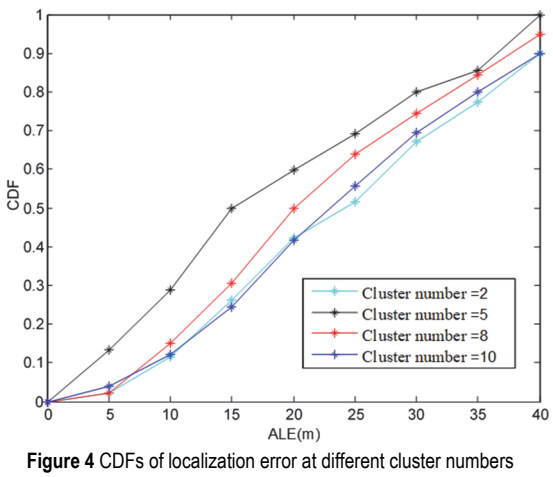

Figure 4 CDFs of localization error at different cluster numbers

For anchor selection, cluster number κ has a great impact on the localization performance. In the basic scenario of Tab. 1, the cluster number κ varies from 2 to 10: $\kappa = 2, \kappa = 5, \kappa = 8, \kappa = 10$. Under this setting, the relationship between localization accuracy and cluster number was simulated. The simulation results are displayed in Fig. 4.

As shown in Fig. 4, the localization errors were mostly distributed between 0 m and 40 m. The CDF was low and the error was large at the cluster number of 2 or 10, while the CDF was high and the error was small at the cluster number of 5. The CDF values indicate that the localization performance was improved by about 4m, as the cluster number reached 5. When there are a few clusters, the node distance is not measured accurately, and the clustering does not contribute to localization. On the contrary, when there are too many clusters, the samples would be allocated to scattered groups. This weakens the screening effect of clustering, because it is impossible to select the anchors with relatively accurate distance. In addition, over clustering will push up the computing load, which undermines the localization efficiency of WSNs. To ensure the effect of node localization, the cluster number was set to 5 for the KMC.

The number of selected anchors also affects the localization performance of the IVF algorithm. To disclose the effect, a simulation was carried out with different number of selected anchors: 4, 5, 6, and 7. The simulation results are displayed in Fig. 5.

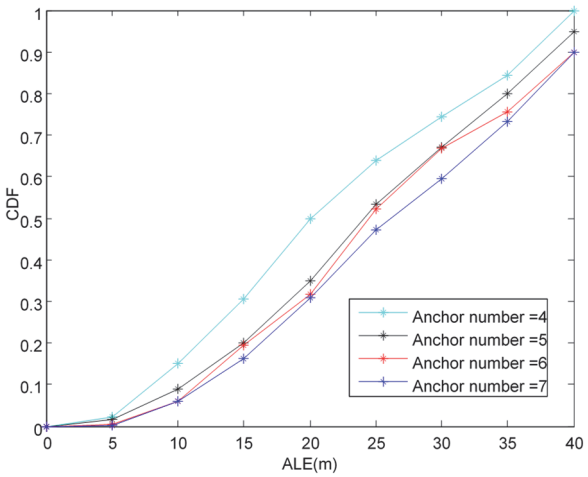

Figure 5 CDFs of localization error at different anchor numbers

For 3D localization, at least 4 anchors are required to form a set of selected anchors, and locate the agent by the IVF method. As shown in Fig. 5, the localization accuracy peaked at the anchor number of 4. With the growing number of selected anchors, the localization became less and less accurate. This means, unlike single centroid localization, a large anchor number does not benefit the performance of the IVF method. In single centroid localization, the centroid is calculated only once, and treated as the estimated position of the agent. After all, single centroid localization requires the agent to be in the space enclosed by anchors, and takes the mean of the anchors as the centroid. A large number of anchors helps to avoid large deviation, and balance the mean value. On the contrary, the IVF algorithm analogizes the iterative

centroid method. The initial centroid of the enclosed space or the initial balance point is not regarded as the estimated position of the agent. As a result, the mean function of the anchors has little to do with the localization. Rather, the measuring error of node distance may impact on the APIT accuracy or the drifting variable, and play a key role in the localization result of iterative centroid or IVF. Since more selected anchors leads to extra comparison and redundant computation, the anchor number for 3D localization was optimized as 4.

Since the agent position is estimated based on the iterative balance points, it is important for the IVF algorithm to converge iteratively. Therefore, another simulation was performed on the convergence speed at different number of iterations. The simulation results are reported in Figs. 6 and 7. As shown in Fig. 6, the balance point changes, i.e., the distance between the agent and the balance point, decreased with the growing number of iterations. When the balance point replaces the anchor in the set of selected anchors, which is the farthest from the agent, the distance between the agent and the balance point becomes smaller. Iteratively, the balance point gets closer to the agent, which guarantees the algorithm convergence. As shown in Fig. 7, the ALE decreased with the growing number of iterations, and tended to be stable at the 6th iteration. The final result is not the optimal solution, but the near-optimal solution. This is because the error in distance estimation is inherent and cannot be avoided. Overall, the ALE of the IVF algorithm is also convergent like balance point change. Unlike balance point change, the ALE does not always decrease, i.e., does not change monotonically.

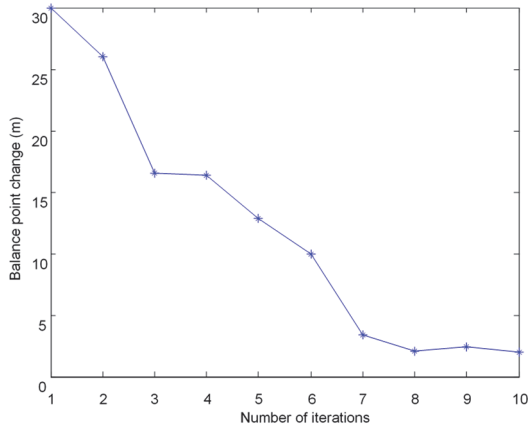

Figure 6 The balance point change with the growing number of iterations

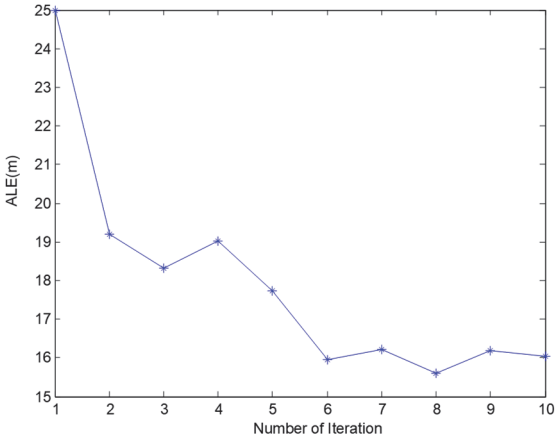

Figure 7 The ALE with the growing number of iterations

3.3 Distribution Parameters

The localization performance may vary with distribution parameters like communication radius and anchor proportion. This subsection aims to reveal the influence of these parameters.

The distribution parameter of communication radius depends on hardware technique. Based on the basic scenario in Tab. 1, the communication radius was changed from 20 m, 30 m, 40 m, to 50 m, and the CDFs with different communication radii were simulated. The simulation results are displayed in Fig. 8.

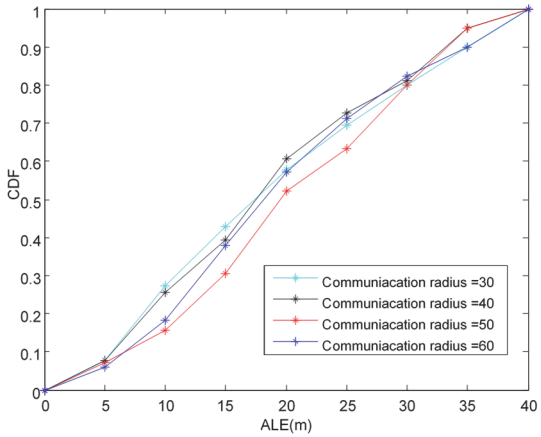

Figure 8 The CDFs with different communication radii

As shown in Fig. 8, the CDFs with respect to different communication radii were close. No regularity was observed as the radius changed from 20 m, 30 m, 40 m, to 50 m. Therefore, the communication radius has little influence on localization performance. However, the communication radius might be configured carefully, such that every node can communicate with at least one node, when the node density is low. If the radius is too short, it may not be possible for a node in a sparse to communicate with another node, not to mention node localization. In other words, the communication radius should guarantee the connectivity among nodes.

Anchor proportion is defined as the percentage of anchors in all nodes. Based on the basic scenario in Tab. 1, the anchor proportion was changed from 10%, 20%, 30% to 40%, and the CDFs with different anchor proportions were simulated. The simulation results are displayed in Fig. 9.

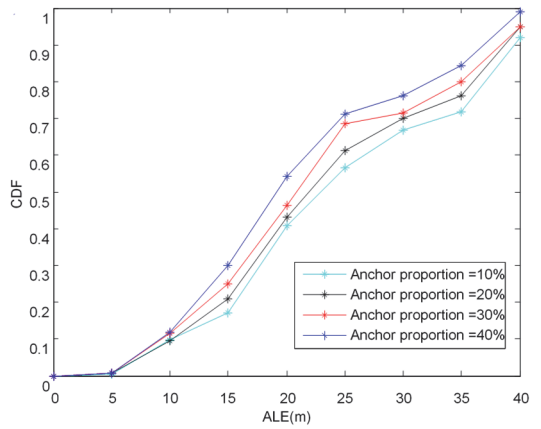

Figure 9 The CDFs with different anchor proportions

As shown in Fig. 9, the localization accuracy increased with the anchor proportion, indicating that anchor proportion promotes localization effect. A large anchor proportion provides many references for localization, and mainly improves the accuracy in distance estimation. When the anchor proportion is small, the improving effect is limited, but the accuracy is still acceptant. Our method still has a good localization result, even if the anchors only take up small proportion of all nodes. To a certain extent, the iterative algorithm lowers the hardware cost of anchors. In addition, a small anchor proportion lowers the computing load and speeds up the operation, thereby broadening the application of our localization method.

The simulations about environmental parameters prove that the localization effect is not affected by communication radius and is little affected by anchor proportion. Therefore, our iterative algorithm is suitable for various applications, especially those with a short communication radius and a few anchors. With a low hardware cost, the algorithm can be applied effectiveness in complex 3D environments.

3.4 Performance Comparison

The above simulations prove the effectiveness of the IVF method in complex and harsh scenarios. More comparative simulations were designed to evaluate its superiority over the other localization methods, such as 3D-DV-hop, Ou-3D in [31], and 3D-centroid [29]. The simulations were conducted in the basic scenario. The simulation results are reported in Fig.10.

As shown in Fig. 10, the IVF method achieved basically the same accuracy and CDFs as Ou-3D. Meanwhile, 3D DV-hop ended up with the lowest accuracy, and the accuracy of 3D-centroid fell between that of the IVF and 3D DV-hop. The ALE results show that the IVF method improves the localization accuracy by 6 m from the level of 3D DV-hop. This improvement is valuable for large WSNs.

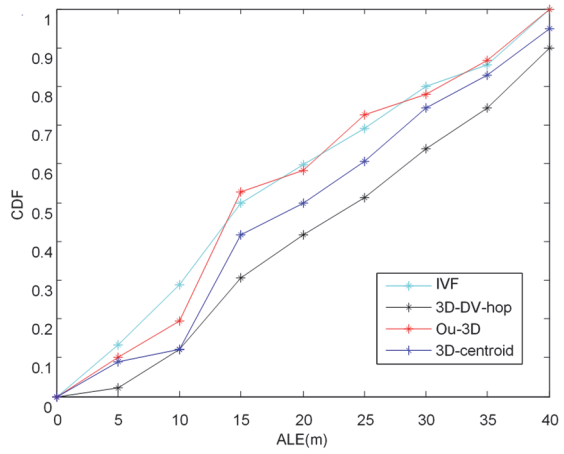

Figure 10 Simulation results of different methods

The above methods were also compared in terms of hardware cost, node distribution, and computing complexity.

On hardware cost, the IVF has a more relaxed hardware demand than Ou-3D, which needs many mobile anchors with localization devices. The IVF has the same hardware demand as 3D DV-hop: both need some anchors with localization devices. The IVF can work effectively with fewer anchors, while 3D DV-hop will become less accurate, if fewer anchors are available. The hardware demand is the same between 3D-centroid and the IVF, but the former is relatively low in accuracy. Overall, the IVF method outperforms in accuracy and hardware cost among the contrastive methods.

On node distribution, the IVF and 3D-DV-hop have no requirement on anchor distribution, and can work on randomly distributed anchors. However, 3D-centroid requires the agent to fall in the space enclosed by anchors, which is hard to achieve in unknown environments (e.g., battlefield, and gas detection). Ou-3D requires mobile anchors to plan paths that cover all agents. Compared with the other methods, the IVF has a very relaxed requirement on node distribution. As a result, the IVF can achieve a high robustness in special cases with complex unknown environments, while the other methods cannot.

On computing complexity, Tab. 2 compares the contrastive methods in details.

Table 2 Computing complexity of different methods

Algorithm	Complexity
IVF	$O(N_u(M_1 + M_2))$
3D-DV-hop	$O(N_u)$
Ou-3D	$O(N_u M_3 + A)$
3D-centroid	$O(N_u M_4)$

Note: Nu is the total number of agents; M_1 and M_2 are the iterative computation in anchor selection and IVF, respectively; M_3 is the localization computation of Ou-3D method; M_4 is the localization computation of 3D-centroid method.

As shown in Tab. 2, the computing loads of M_1 and M_2 are limited. The IVF has a similar computing load as 3D-centroid. The least computing load belongs to 3D-DV-hop, while Ou-3D needs to make the extra computation A of path planning. In a complex environment, path planning will significantly add to the computing load. Overall, the IVF has an acceptable computing load, although not the lowest among the methods.

3.5 Application Analysis

The proposed localization method aims to adapt to various node distributions and application requirements. To explore the application scope of our method, the localization accuracy, computing complexity, and hardware cost of the IVF method were compared with those of 3D genetic algorithm based improved distance vector hop (3D-GAIDV) method [35], 3D underwater localization (3DUL) method [36], and multidimensional scaling (MDS) method [37]. The performance of these methods is compared in Tab. 3.

Table 3 Comparison of application performance of different methods

Algorithm	Localization accuracy	Computing complexity	Hardware cost
IVF	10 ~ 15 m	$O(N_u(M_1 + M_2))$	Low
3D-GAIDV	6 ~ 12 m	$O(N_u N_a^3)$	Low
3DUL	10 ~ 15 m	$O(N_u)$	High
MDS	6 ~ 12 m	$O(2^{Nu})$	Low

As shown in Tab. 3, the IVF and 3DUL achieved the same level of localization accuracy, higher than that of 3D-

GAIDV and MDS. Similarly, the IVF and 3DUL realized a relatively low computing complexity, because 3D-GAIDV seriously depends on anchor number, while MDS runtime increases exponentially with the node number. Hence, our method can localize targets accurately without increasing computing complexity. In aspect of hardware cost, the IVF, 3D-GAIDV and MDS all achieved a relatively low cost. These range-free methods need no extra measurement devices, and do not consider time sync or angle of arrival. By contrast, 3DUL must estimate the propagation delay based on two-way message exchange, and derive inter-node distances from estimated sound speed. To estimate the sound speed, it is necessary to equip conductivity, temperature, and depth (CTD) sensors.

In general, the IVF provides an excellent algorithm for accurate localization in 3D environments, and strikes a balance between hardware cost and computing complexity. Although it is not as accurate as 3D-GAIDV and MDS, the IVF method makes up the gap by the low hardware cost, and limited computing load. When it comes to applications like forest fire warning, lots of low-cost sensors are needed. To reduce the energy cost, these sensors are usually in sleep state, and will be discarded whenever the energy runs out. Once a fire is detected, the sensors must be awakened to sense temperature, and send the information to the administration center. With the aid of the IVF, the fire can be localized by each sensor with a low computing cost, and the localization accuracy is enough to pinpoint the fire area. Therefore, our approach can be extensively and flexibly applied in complex and volatile environments.

4 CONCLUSIONS

This paper proposes the IVF method for 3D localization, aiming to overcome the effect of the additional dimension. Through the KMC, the anchor outliers were screened, and a set of selected anchors were formulated. Then, a drifting coefficient was defined, and the balance point was deduced iteratively to estimate the agent positions at a low hardware cost and low computing complexity. In practical engineering, the IVF method faces very few limitations, and provides a suitable localization tool for complex and volatile environments. Extensive simulations show that the IVF method achieved higher efficiency and better accuracy than traditional approaches.

Acknowledgments

Project supported by Science and Technology Nova Plan of Beijing City, China (Grant No. Z201100006820122).

5 REFERENCES

[1] Boubrima, A., Bechkit, W., & Rivano, H. (2019). On the optimization of WSN deployment for sensing physical phenomena: Applications to urban air pollution monitoring. *Mission-Oriented Sensor Networks and Systems: Art and Science*, 99-145. https://doi.org/10.1007/978-3-319-91146-5_4

[2] Lalitha, A. & Reddy, G. H. (2021). An integrated signal allocation model with effective collision resolution model for performance enhancement of wireless sensor networks. *Traitement du Signal*, *38*(5), 1369-1375. https://doi.org/10.18280/ts.380512

[3] Saeed, N., Celik, A., Al-Naffouri, T. Y., & Alouini, M. S. (2019). Localization of energy harvesting empowered under water optical wireless sensor networks. *IEEE Transactions on Wireless Communications*, *18*(5), 2652-2663. https://doi.org/10.1109/TWC.2019.2906309

[4] Dou, Z., Wang, X., & Li, Y. (2019). Addressing for 6LoWPAN. *International Journal of Internet Protocol Technology*, *12*(1), 51-60.

[5] Ranjan, A., Sahu, H. B., & Misra, P. (2020). Modeling and measurements for wireless communication networks in underground mine environments. *Measurement*, *149*, 106980. https://doi.org/10.1016/j.measurement.2019.106980

[6] Alouache, L., Nguyen, N., Aliouat, M., & Chelouah, R. (2020). HSDN-GRA: A hybrid software-defined networking-based geographic routing protocol with multi-agent approach. *International Journal of Communication Systems*, *33*(15), e4521. https://doi.org/10.1002/dac.4521

[7] Pandey, O. J., Gautam, V., Jha, S., Shukla, M. K., & Hegde, R. M. (2020). Time synchronized node localization using optimal h-node allocation in a small world wsn. *IEEE Communications Letters*, *24*(11), 2579-2583. https://doi.org/10.1109/LCOMM.2020.3008086

[8] Benaissa, B. E., Lahfa, F., Naima, K., Lorenzini, G., Inc, M., & Menni, Y. (2021). Detection and cooperative communications for deployment sensor networks. *Traitement du Signal*, *38*(3), 555-564. https://doi.org/10.18280/ts.380303

[9] Guo, Y., Han, Q., & Kang, X. (2020). Underwater sensor networks localization based on mobility-constrained beacon. *Wireless Networks*, *26*(4), 2585-2594. https://doi.org/10.1007/s11276-019-02023-5

[10] Dong, M., Li, H., Yin, R., Qin, Y., & Hu, Y. (2021). Scalable asynchronous localization algorithm with mobility prediction for underwater wireless sensor networks. *Chaos, Solitons & Fractals*, *143*, 110588. https://doi.org/10.1016/j.chaos.2020.110588

[11] Huang, M., Liu, A., Zhao, M., & Wang, T. (2019). Multiworking sets alternate covering scheme for continuous partial cover agein WSNs. *Peer-to-Peer Networking and Applications*, *12*(3), 553-567. https://doi.org/10.1007/s12083-018-0647-z

[12] Karupusamy, S., Maruthachalam, S., Mayilswamy, S., Sharma, S., Singh, J., & Lorenzini, G. (2021). Efficient computation for localization and navigation system for a differential drive mobile robot in indoor and outdoor environments. *Revue d'Intelligence Artificielle*, *35*(6), 437-446. https://doi.org/10.18280/ria.350601

[13] Dou, R., Fang, X., & Huang, D. (2021). An Efficient range-free multi-hop localisation algorithm for irregular wireless sensor networks. *International Journal of Security and Networks*, *16*(1), 37-48. https://doi.org/10.1504/IJSN.2021.112839

[14] Dai, Z., Wang, G., Jin, X., & Lou, X. (2020). Nearly optimal sensor selection for TDOA-based source localization in wireless sensor networks. *IEEE Transactions on Vehicular Technology*, *69*(10), 12031-12042. https://doi.org/10.1109/TVT.2020.3011118

[15] Thazeen, S., Mallikarjunaswamy, S., Siddesh, G. K., & Sharmila, N. (2021). Conventional and subspace algorithms for mobile source detection and radiation formation. *Traitement du Signal*, *38*(1), 135-145. https://doi.org/10.18280/ts.380114

[16] Song, B., Li, S. L., Tan, M., & Zhong, W. (2019). An adaptive approach for ultra-wideband positioning in complex environment. *Instrumentation Mesure Métrologie*, *18*(5), 435-443. https://doi.org/10.18280/i2m.180502

[17] He, Y., Li, J., & Zhang, X. (2020). Adaptive cascaded high-resolution source localization based on collaboration of multi-UAVs. *China Communications*, *17*(4), 165-179. https://doi.org/10.23919/JCC.2020.04.015

[18] Prashar, D. & Jha, N. (2021). Review of secure distributed range-freehop-based localization algorithms in the wireless sensor networks. *Multimedia Security*, 283-302. https://doi.org/10.1007/978-981-15-8711-5_15

[19] Han, S., Zhang, B., & Chai, S. (2021). A novel auxiliary hole localization algorithm based on multidimensional scaling for wireless sensor networks in complex terrain with holes. *Ad Hoc Networks*, *122*, 102644. https://doi.org/10.1016/j.adhoc.2021.102644

[20] Kaur, A., Kumar, P., & Gupta, G. P. (2019). A weighted centroid localization algorithm for randomly deployed wireless sensor networks. *Journal of King Saud University-Computer and Information Sciences*, *31*(1), 82-91. https://doi.org/10.1016/j.jksuci.2017.01.007

[21] Gutiérrez, S. & Ponce, H. (2019). An intelligent failure detection on a wireless sensor network for in door climate conditions. *Sensors*, *19*(4), 854. https://doi.org/10.3390/s19040854

[22] Xu, Y., Luo, X., Wang, W., & Zhao, W. (2017). Efficient dv-hop localization for wireless cyber-physical social sensing system: A correntropy-based neural network learning scheme. *Sensors*, *17*(1), 135. https://doi.org/10.3390/s17010135

[23] Cai, X., Wang, P., Du, L., Cui, Z., Zhang, W., & Chen, J. (2019). Multi-objective three-dimensional DV-hop localization algorithm with NSGA-II. *IEEE Sensors Journal*, *19*(21), 10003-10015. https://doi.org/10.1109/JSEN.2019.2927733

[24] Kumari, S. & Gupta, G. P. (2019). Target Localization Algorithm in a Three-Dimensional Wireless Sensor Networks. *International Conference on Computer Networks and Communication Technologies*, 33-42. https://doi.org/10.1007/978-981-10-8681-6_5

[25] Chuku, N. & Nasipuri, A. (2021). RSSI-Based localization schemes for wireless sensor networks using outlier detection. *Journal of Sensor and Actuator Networks*, *10*(1), 10. https://doi.org/10.3390/jsan10010010

[26] Wang, S. & Jiang, X. (2020). Three-dimensional cooperative positioning in vehicular ad-hoc networks. *IEEE Transactions on Intelligent Transportation Systems*, *22*(2), 937-950. https://doi.org/10.1109/TITS.2019.2961452

[27] Lin, C., Guo, C., Dai, H., Wang, L., & Wu, G. (2019). Near optimal charging scheduling for 3-D wireless rechargeable sensor networks with energy constraints. *2019 IEEE 39th International Conference on Distributed Computing Systems (ICDCS)*, 624-633. https://doi.org/10.1109/ICDCS.2019.00068

[28] Kumari, J., Kumar, P., & Singh, S. K. (2019). Localization in three-dimensional wireless sensor networks: a survey. *The Journal of Supercomputing*, *75*(8), 5040-5083. https://doi.org/10.1007/s11227-019-02781-1

[29] Liu, X. (2020). Research on WSN Node Localization Algorithm Based on RSSI Iterative Centroid Estimation. *Tehnički vjesnik*, *27*(5), 1544-1550. https://doi.org/10.17559/TV-20190827114252

[30] Singh, P., Mittal, N., Srivastava, R., Kautish, S., & Tiwari, R. (2020). A Review on Localization in Wireless Sensor Networks for Static and Mobile Applications. *Green Information and Communication Systems for a Sustainable Future*, 1-28. https://doi.org/10.1201/9781003032458-1

[31] Sorbelli, F. B., Das, S. K., Pinotti, C. M., & Rigoni, G. (2021). A comprehensive investigation on range-free localization algorithms with mobile anchors at different altitudes. *Pervasive and Mobile Computing*, *73*, 101383. https://doi.org/10.1016/j.pmcj.2021.101383

[32] Khalaf-Allah, M. (2021). Novel Solutions to the Three-Anchor To A - Based Three-Dimensional Positioning Problem. *Sensors*, *21*(21), 7325. https://doi.org/10.3390/s21217325

[33] Patalas-Maliszewska, J., & Halikowski, D. (2019). A model for generating work place procedures using a CNN-SVM architecture. *Symmetry*, *11*(9), 1151. https://doi.org/10.3390/sym11091151

[34] Guan, Y., Ma, L., Sun, X., & Yan, J. (2021). Research on multi-sensor mission planning in target tracking system based on SVM. https://doi.org/10.1049/icp.2021.0650

[35] Kanwar, V. & Kumar, A. (2021). Range Free Localization for Three Dimensional Wireless Sensor Networks Using Multi Objective Particle Swarm Optimization. *Wireless Personal Communications*, *117*(2), 901-921.

[36] Lin, Y., Tao, H., Tu, Y., & Liu, T. (2019). A node self-localization algorithm with a mobile anchor node in underwater acoustic sensor networks. *IEEE Access*, *7*, 43773-43780. https://doi.org/10.1109/ACCESS.2019.2904725

[37] Saeed, N., Nam, H., Al-Naffouri, T. Y., & Alouini, M. S. (2019). A state-of-the-art survey on multidimensional scaling-based localization techniques. *IEEE Communications Surveys & Tutorials*, *21*(4), 3565-3583. https://doi.org/10.1109/COMST.2019.2921972

Contact information:

Yun HONG
School of Modern Post (School of Automation),
Beijing University of Posts and Telecommunications,
Beijing 100876, China
E-mail: HongYun805@bupt.edu.cn

Song WANG
(Corresponding author)
School of Modern Post (School of Automation),
Beijing University of Posts and Telecommunications,
Beijing 100876, China
E-mail: wongsang@bupt.edu.cn

Huibing KANG
School of Modern Post (School of Automation),
Beijing University of Posts and Telecommunications,
Beijing 100876, China
E-mail: Khbing@bupt.edu.cn

Yanzhu HU
School of Modern Post (School of Automation),
Beijing University of Posts and Telecommunications,
Beijing 100876, China
E-mail: bupt_automation_safety_yzhu@bupt.edu.cn