

Research Article

Multirobot Collaborative Navigation Algorithms Based on Odometer/Vision Information Fusion

Guohua Liu ¹, Juan Guan,¹ Haiying Liu,² and Chenlin Wang³

¹School of Mathematics, Southeast University, Nanjing 210096, China

²College of Astronautics, Nanjing University of Aeronautics and Astronautics, Nanjing 210016, China

³China COMAC Shanghai Aircraft Design and Research Institute, Shanghai 201210, China

Correspondence should be addressed to Guohua Liu; liuguohua@seu.edu.cn

Received 24 May 2020; Revised 15 June 2020; Accepted 11 July 2020; Published 27 August 2020

Guest Editor: Xiao Ling Wang

Copyright © 2020 Guohua Liu et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Collaborative navigation is the key technology for multimobile robot system. In order to improve the performance of collaborative navigation system, the collaborative navigation algorithms based on odometer/vision multisource information fusion are presented in this paper. Firstly, the multisource information fusion collaborative navigation system model is established, including mobile robot model, odometry measurement model, lidar relative measurement model, UWB relative measurement model, and the SLAM model based on lidar measurement. Secondly, the frameworks of centralized and decentralized collaborative navigation based on odometer/vision fusion are given, and the SLAM algorithms based on vision are presented. Then, the centralized and decentralized odometer/vision collaborative navigation algorithms are derived, including the time update, single node measurement update, relative measurement update between nodes, and covariance cross filtering algorithm. Finally, different simulation experiments are designed to verify the effectiveness of the algorithms. Two kinds of multirobot collaborative navigation experimental scenes, which are relative measurement aided odometer and odometer/SLAM fusion, are designed, respectively. The advantages and disadvantages of centralized versus decentralized collaborative navigation algorithms in different experimental scenes are analyzed.

1. Introduction

With the development of the age of intelligence, the application scenarios and task requirements of intelligent mobile robots are becoming more and more complex [1, 2]. A single robot has difficulty in meeting the needs of humans for highly automated robots. Therefore, the multirobot system, which fully embodies the group wisdom, has a broad prospect. The subject has attracted extensive attention of researchers [3–5]. The premise of robot collaborative navigation operation is that each robot needs to have a certain ability of navigation and localization. Therefore, it is very important for the research of robot collaborative navigation system [6].

The navigation accuracy of a single robot depends on its own navigation system, independent of other motion bodies.

This navigation method is relatively simple, but due to the computing power of its own processor, sensor quality, and sensor field of vision and other factors, it limits the working range of the navigation system to a certain extent in the process of navigation and positioning. It also affects the ability of the navigation system to suppress noise, reducing errors and adapting to the complex environment. The collaborative navigation system of robot can make up for the single robot navigation system. When there is relative measurement between each robot, the relative information can be fully used to correct the navigation results. It is also possible to establish links with each other to realize the sharing of navigation resources among robots and to obtain better navigation performance [7–9].

In the research of mobile robot navigation and location technology, the development of Simultaneous Localization

and Mapping (SLAM) technology provides a new thought for autonomous positioning of mobile robot under complex scenes [10]. At the same time, as a significant technical means to realize relative navigation, radio navigation and visual navigation are being paid much attention. It provides important technical support for further research on the collaborative navigation of multirobots [11–13]. However, the traditional combined navigation technology is far from meeting the growing demand for the high performance of collaborative navigation.

At present, most of the data fusion methods of multi-source heterogeneous sensors in integrated navigation system use centralized data fusion algorithm. This algorithm has some defects in collaborative navigation of mobile robots, high communication cost, and poor robustness. However, the current research on decentralized data fusion algorithm, which is suitable for collaborative navigation environment is not mature enough [14, 15].

Therefore, based on the information fusion of multi-source heterogeneous sensors and related SLAM data association algorithms, this paper puts forward a new concept, model, and solution of multisource heterogeneous sensor information fusion, which can significantly improve the performance of collaborative navigation, address some problems of cooperative navigation technology, explore the key technologies that need to be solved in the collaborative navigation process of multirobot platform, and provide theoretical basis and technical support for high performance of collaborative navigation application.

The structure of this paper is outlined as follows: in Section 2, in the multisource information fusion collaborative navigation system model, the principle of the module involved, and the source of error are analyzed. In Section 3, the framework of centralized and decentralized collaborative navigation framework based on odometer/vision red fusion is given, and the SLAM algorithms based on vision are presented. In Sections 4 and 5, the centralized and decentralized odometer/vision collaborative navigation algorithms are derived, respectively. In Section 6, different simulation experiments are designed to verify the effectiveness of the algorithms.

2. Multisource Information Fusion Collaborative Navigation System Model

2.1. Motion Robot Model. Suppose that the mobile robot moves according to the circular arc trajectory. Figure 1 shows the motion of a single mobile robot (see Figure 1).

2.1.1. Odometry Measurement Model. According to the principle of odometer motion calculation [16], it is assumed that the starting position of the mobile robot is (x, y, θ) , after ΔT of movement to the end. The end position is $(x + \Delta x, y + \Delta y, \theta + \Delta\theta)$, and the trajectory can be regarded as a circular arc ΔS with radius R . In this case, the pose of the robot (in discrete time) is given by

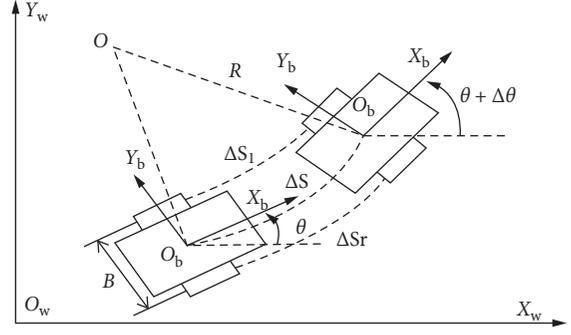


FIGURE 1: Motion model of a single mobile robot.

$$\begin{cases} \Delta S_l = \pi D \cdot \frac{\Delta N_L}{P} = \left(R - \frac{B}{2}\right) \Delta\theta, \\ \Delta S_r = \pi D \cdot \frac{\Delta N_R}{P} = \left(R + \frac{B}{2}\right) \Delta\theta, \end{cases} \quad (1)$$

where $\Delta N_L, \Delta N_R$ are increment of pulse number of odometer left and right wheel encoder. $(\Delta S_r, \Delta S_l)$ is the distance between the right wheel and the left wheel of a mobile robot. P are the resolution parameters of left and right wheel encoder. The pose increment of sampling period ΔT is obtained by solving the above equations. The discrete motion equation of the mobile robot is further obtained by

$$\begin{cases} x_k \approx x_{k-1} + \Delta S_{k-1} \cos\left(\frac{\theta_{k-1} + \Delta\theta_{k-1}}{2}\right), \\ y_k \approx y_{k-1} + \Delta S_{k-1} \sin\left(\frac{\theta_{k-1} + \Delta\theta_{k-1}}{2}\right), \\ \theta_k = \theta_{k-1} + \Delta\theta_{k-1}. \end{cases} \quad (2)$$

2.1.2. Odometer Error Model. There are systematic errors and random errors in odometer based positioning [17]. In the simulation experiment, the influence of random error is mainly considered. The corresponding covariance matrix Σ_{odom} is given by

$$\Sigma_{\text{odom}} = \begin{bmatrix} k_r |\Delta S_r| & 0 \\ 0 & k_l |\Delta S_l| \end{bmatrix}. \quad (3)$$

2.2. Lidar Environmental Detection Model. As the main module to realize the SLAM, environment detection module mainly obtains the surrounding environment information through the external sensing sensor.

2.2.1. Lidar Scanning Model. The two-dimensional lidar outputs the point cloud data from the surrounding environment and contains the angular information. The scanning schematic of the two-dimensional lidar is shown in Figure 2 (see Figure 2).

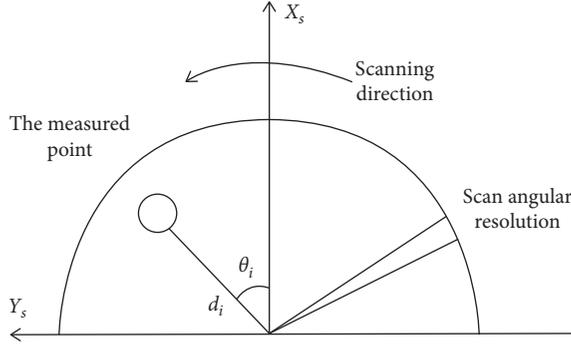


FIGURE 2: The scanning schematic diagram of the two-dimensional lidar.

As shown in Figure 2, the lidar is scanned in the sensor coordinate system along the scanning direction, with a fixed scanning angle resolution. The measured point is detected in the i th direction. Denote θ_i as the angle between this direction and the positive direction of the X_s -axis, and θ_i can be obtained by the resolution of the scanning angle with the inherent. A frame of point cloud data is obtained after a scan, which can be recorded as a set $A = \{(d_i, \theta_i) | i = 1, \dots, n\}$, where n is the number of scanned data points in this frame of data, (d_i, θ_i) is the coordinate of the measured point in the polar coordinate system. And the final point cloud data set $B = \{(x_i, y_i) | i = 1, \dots, n\}$ is obtained too.

2.2.2. Lidar Error Model. In the simulation experiment, the above error can be simplified to Gaussian white noise along the x and y directions in the carrier coordinate system of the mobile robot [18]. It is assumed that the positioning errors are in the two independent directions, so the corresponding covariance matrix is given by

$$\Sigma_{\text{rad}} = \begin{bmatrix} \sigma_x^b & 0 \\ 0 & \sigma_y^b \end{bmatrix}, \quad (4)$$

where σ_x^b and σ_y^b represent the variance along the x and y directions of the observation in the carrier coordinate system of the mobile robot, respectively.

2.3. UWB Relative Measurement Model. The relative measurement module is an important part of the cooperative navigation network in which nodes relate to the state information of the surrounding nodes. UWB (Ultra Wide Band) can measure relative distance and lidar can measure relative position which are selected as research objects in this paper. Because the scanning model of lidar has been described in the previous section, this section only establishes the corresponding ranging model and error model.

2.3.1. UWB Ranging Model. At present, there are two main ranging methods, that is, two-way ranging (TWR) method and symmetric double-sided two-way ranging (SDSTWR) method. SDSTWR method is greatly improved compared with TWR method in terms of ranging error caused by clock

synchronization. Therefore, SDSTWR method is often used for distance measurement [19] (see Figure 3).

The distance expression between transceiver A and transceiver B can be obtained from Figure 3:

$$d = c \frac{T_{A1}T_{B2} - T_{B1}T_{A2}}{T_{A1} + T_{A2} + T_{B1} + T_{B2}}, \quad (5)$$

where d is the distance between the two tested transceivers. $T_{A1} = T_{a2} - T_{a1}$ is the time interval between the first signal and the received signal of transceiver A. We can get T_{A2}, T_{B1}, T_{B2} in the same way.

2.3.2. UWB Error Model. One of the main sources of error in UWB ranging is the error caused by the clock synchronization of the two transceivers [20]:

$$e_{\text{SDS}} = T_f c \left(1 - \frac{k_A + k_B}{2} \right), \quad (6)$$

where k_A and k_B are the ratio of transceivers A and B between the actual and the expected frequency.

2.4. SLAM Model Based on Lidar Measurement. SLAM is widely used in the field of mobile robot navigation and positioning [21]. It is the key technology to solve the mapping problem at the same time, and it also provides a new idea for solving the path planning problem. SLAM technology mainly aims at the unknown location of the mobile robot. The environmental map is built in the form of an increasing perception of the surrounding strange environment according to the external sensor. At the same time, the state estimation of the mobile robot itself is obtained by using the built map information.

The SLAM localization technology is the key technology to realize the autonomous positioning of the mobile robot under the unknown environment. It is of great research value to realized more accurate navigation and localization under the condition of reducing the limitation of moving robot (such as no GPS) [22]. By applying the technology to the framework of the collaborative navigation system of the mobile robot, the navigation performance of the whole collaborative navigation system can be effectively improved (see Figure 4).

It can be seen from Figure 4 that a mobile robot's localization using a SLAM navigation system is essentially a process, which is continuous estimation to approximate the true value [23]. The triangle in the graph represents the mobile robot and the circle represents the observed landmark, where the grey circle represents the estimated landmark. The solid line connecting the triangle represents the real trajectory, and the dotted line is the estimated trajectory (see Figure 4).

3. Collaborative Navigation Framework Based on Odometer/Vision

When nodes in collaborative navigation network participate in collaborative navigation, different data fusion algorithms

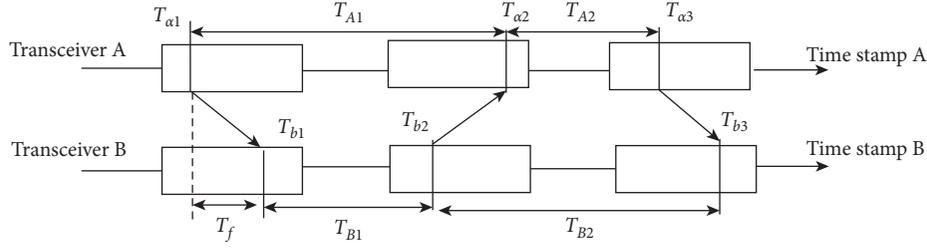


FIGURE 3: Principle of SDSTWR.

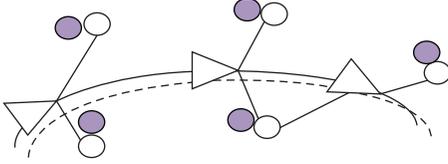


FIGURE 4: SLAM model diagram.

can be used for data fusion, which is obtained multisource heterogeneous sensors of different nodes. There are two data fusion algorithms generally used, centralized and decentralized data fusion algorithms [24].

3.1. Framework of Collaborative Navigation Algorithm. In this subsection, a corresponding data fusion framework of the centralized collaborative navigation algorithm is designed for the general odometer/vision model, and a data fusion framework of decentralized collaborative navigation algorithm is designed in the same way.

In the centralized collaborative navigation structure, the measured data obtained by each node are concentrated in a data fusion center for data fusion. In a distributed collaborative navigation structure, each node shares some information with other nodes while processing its own sensor data (see Figure 5).

According to the odometer/vision collaborative navigation model, we use the most common EKF algorithms to simulate centralized and decentralized cooperative navigation. A centralized location algorithm (CL) and a decentralized location algorithm (DCL) are designed.

As shown in Figure 6, in the CL algorithm corresponding to the centralized cooperative navigation structure, each node sends the information obtained by the sensor itself to the central server and realizes the data fusion through the EKF algorithm. Its state vector is a set of state vectors of each node, which is updated with the principle of track reckoning. After that, the measurement information obtained after data association in the SLAM process and the relative measurement information between nodes are selected. CL algorithm gathers the state and covariance information of all nodes, corrects them uniformly, and sends the corrected estimation results to each node. Because of participating in the joint measurement update process, the state information of each node is related to each other after the first update (see Figure 6).

This correlation is reasonably estimated based on the CL algorithm, and the task of data fusion is distributed to each node; then the DCL algorithm is proposed accordingly. Since the position of each node in the algorithm is equivalent, only one node needs to be discussed (see Figure 7). In order to avoid overoptimal estimation to the greatest extent, the DCL algorithm in this paper introduces the concept of separating covariance crossover filter. The covariance of each node is divided into correlation term and irrelevant term, and the time update process is basically consistent with the time update process of a single node. The measurement update process will take two steps. Firstly, according to the measurement information of the SLAM navigation system, the state estimation results are obtained by propagating the state and integrating the related information of the auxiliary navigation system. Then, the state information sent by the adjacent nodes and the relative measurement information between the nodes can obtain another state estimation of the node. Here, relevant state information sent to neighboring nodes is used to update local maps (see Figure 7).

3.2. SLAM Algorithms Based on Vision

3.2.1. Landmark State Estimation Algorithm. The key of SLAM navigation algorithm lies in the process of data association. The positioning process of this SLAM navigation system is essentially a process, which is continuous estimation to approximate the true value. This kind of probability estimation problem is usually solved by introducing appropriate filter. The most common is the EKF algorithm (see Figure 8).

Because of the high sampling frequency of the odometer selected in this paper, the lidar also has the advantages of high precision and high reliability; the EKF algorithm with better real-time performance is selected. The state estimation process of landmark information in SLAM based on EKF is described below. The observation equation of the feature information obtained by lidar is as follows:

$$z_k = h(X_{k,l}, X_{k,r}) + n_k = \begin{bmatrix} (x_{k,l} - x_{k,r}) \cos \theta_{k,r} + (y_{k,l} - y_{k,r}) \sin \theta_{k,r} \\ -(x_{k,l} - x_{k,r}) \sin \theta_{k,r} + (y_{k,l} - y_{k,r}) \cos \theta_{k,r} \end{bmatrix} + n_k, \quad (7)$$

where $X_{k,l} = (x_{k,l}, y_{k,l})^T$ is state vector of landmark at k time and $X_{k,r} = (x_{k,r}, y_{k,r}, \theta_{k,r})^T$ is state vector of mobile robot at

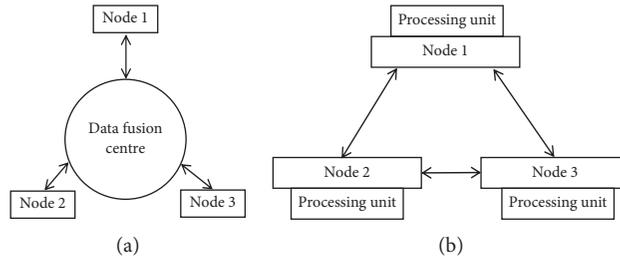


FIGURE 5: Collaborative navigation structure. (a) Centralized collaborative navigation. (b) Decentralized collaborative navigation.

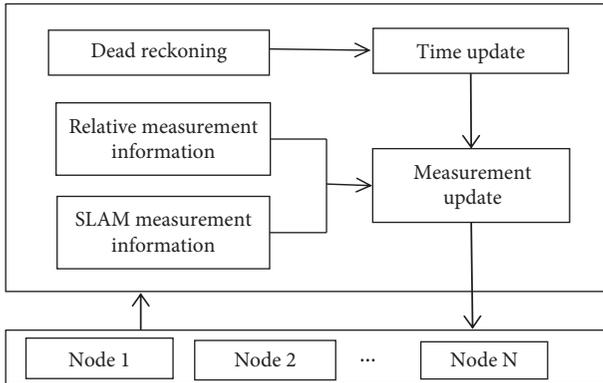


FIGURE 6: Data fusion framework of centralized collaborative navigation algorithm.

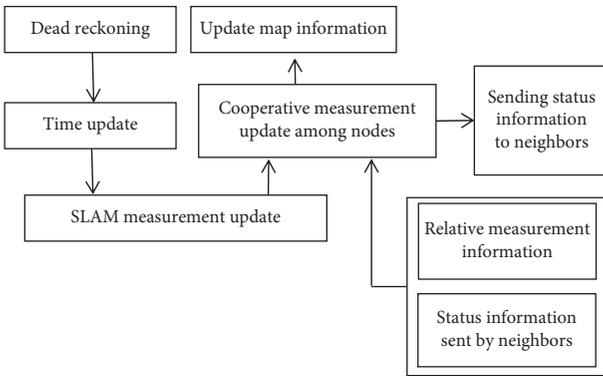


FIGURE 7: Data fusion framework of decentralized collaborative navigation algorithm.

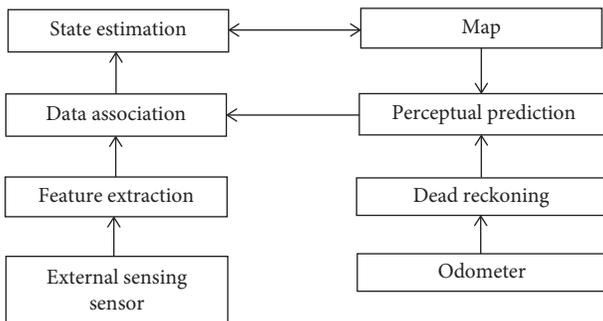


FIGURE 8: SLAM flow chart.

k time. n_k is measurement noise. Its variance matrix is R_k , which can be denoted as $n_k \sim N(0, R_k)$. Since the landmarks are static, the state estimation of $k - 1$ time landmark can be regarded as a priori estimation of k time landmark state. The measurement update process based on EKF is as follows:

Step 1: calculating the innovation and the filter gain:

$$\begin{aligned} v_k &= z_k - h(X_{k-1,l}, X_{k,r}^-), \\ K_k &= \Sigma_{k-1} H_k^T (H_k \Sigma_{k-1} H_k^T + R_k)^{-1}. \end{aligned} \quad (8)$$

Step 2: updating the state estimation and the corresponding covariance:

$$\begin{aligned} X_{k,l} &= X_{k-1,l} + K_k v_k, \\ \Sigma_k &= (I - K_k H_k) \Sigma_{k-1}, \end{aligned} \quad (9)$$

where Σ_k is the covariance matrix for state estimation of landmark at k time and H_k is the measurement matrix at k time.

Remark 1. Any observed landmark information can be position corrected by the above method, and it is noted that such position correction is limited to the landmark in the local map observed at the k time.

3.2.2. Data Association Algorithm. In a SLAM navigation system, the process of data association is an important prerequisite for state estimation. Incorrect data association is likely to lead to serious deviation of estimation results [25, 26].

At present, there are two data association algorithms commonly used in SLAM technology, that is, the nearest neighbor data association algorithm (NN, nearest neighbor) [27] and joint compatibility body & body data association algorithm (JCBB) [28]. The NN algorithm has less computation, but it is easy to form wrong data association when the density of feature information is large, which leads to the divergence of SLAM results, so it is only suitable for the environment where the density of feature information is small and the system error is small. JCBB is the improvement of the NN algorithm, which extends the association of single features in the NN to all observed feature information, which

is more binding and more reliable. The JCBB algorithm can obtain more credible association assumptions than the NN algorithm and exclude some wrong association assumptions. However, the computation amount is obviously increased, which to some extent affects the real-time performance of SLAM navigation system.

To ensure the accuracy of data association in the process of the SLAM, reduce the amount of computation as much as possible, and enhance the real-time performance of SLAM algorithm, this subsection describes an optimized data association algorithm. The classification method mentioned in [29] is used to divide the related feature information set; finally the appropriate feature information set in the local map and the preprocessed observation feature information set are selected to form the association space.

First, the collection of feature information in the local map is divided as follows:

$$\begin{cases} D[(x_m, y_m), (x_k, y_k)] \leq \Delta d, & (x_k, y_k) \in F_k, \\ D[(x_m, y_m), (x_k, y_k)] > \Delta d, & (x_k, y_k) \notin F_k, \end{cases} \quad (10)$$

where $D[(x_m, y_m), (x_k, y_k)]$ is the relative distance between the feature information (x_k, y_k) of the local map and other feature information (x_m, y_m) .

Then, the observation feature information set is preprocessed and divided. In the actual navigation process, the observation feature information obtained by lidar contains noise information. The purpose of preprocessing is to filter out some noise information, improve the accuracy of data association, and reduce the amount of computation at the same time. The judgment process is as follows:

$$f(i, j) = \begin{cases} 1, & D[(x_i, y_j), (x_i, y_j)] < \Delta D, \\ 0, & D[(x_i, y_j), (x_i, y_j)] \geq \Delta D, \end{cases} \quad (11)$$

where ΔD is the threshold, which is determined by the performance of the laser sensor. When the relative distance between the two observation feature information is less than the threshold, the observation feature information is considered to be the feature point; otherwise the noise point does not participate in the subsequent calculation.

When the set is divided, the set of observed feature information is sorted according to the order of observation. Based on the process of the local map feature information set above, the subset is divided in turn, and all points are not involved in the division repeatedly.

Finally, we select the appropriate association set to execute the data association algorithm. The subset of feature information of each local map and the subset of observed feature information at the current time are joint compatibility test, and the feature information with the best test results is selected to form a new subset as the data association object.

4. Centralized Collaborative Navigation Algorithm

4.1. Time Update. First of all, the state space model should be established. The state vector of a single mobile robot with three degrees of freedom contains position and heading angle information. Suppose the number of nodes is N , the state space of collaborative navigation system in centralized framework contains the state vectors of all mobile robots in the group, and the state vector of mobile robot i is X_k^i and the state of system is X_k . Then the state space equation of the system can be expressed as follows:

$$\begin{aligned} X_k &= \begin{bmatrix} X_k^1 \\ X_k^2 \\ \vdots \\ X_k^N \end{bmatrix} = \begin{bmatrix} f^1(X_{k-1}^1, u_{k-1}^1) \\ f^2(X_{k-1}^2, u_{k-1}^2) \\ \vdots \\ f^N(X_{k-1}^N, u_{k-1}^N) \end{bmatrix} + \begin{bmatrix} w_{k-1}^1 \\ w_{k-1}^2 \\ \vdots \\ w_{k-1}^N \end{bmatrix} \\ &\triangleq \Phi(X_{k-1}, u_{k-1}) + w_{k-1}, \end{aligned} \quad (12)$$

where the function $f^i(X, u)$ describes the kinematic characteristics of the mobile robot, $u_{k-1}^i = [\Delta S_{r,k-1} \ \Delta S_{l,k-1}]^T$ represents the input required by the mobile robot i to calculate the track at time k , w_{k-1}^i is the system noise, and $w_{k-1}^i \sim N(0, Q_{k-1}^i)$.

It is assumed that the motion of any node is not affected by any other node, and each node moves independently without being controlled by other nodes. Therefore, the state transfer matrix for centralized collaborative positioning is given by

$$F_{k-1} = \begin{bmatrix} J_{X^{(k-1)}}^1 & 0 & \cdots & 0 \\ 0 & J_{X^{(k-1)}}^2 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & J_{X^{(k-1)}}^N \end{bmatrix}, \quad (13)$$

where $J_{u^{(k-1)}}^i$ and $J_{X^{(k-1)}}^i$ are the Jacobian matrices of function f for state vectors and control inputs, respectively. The system noise variance matrix of collaborative navigation system in centralized framework is as follows:

$$Q_{k-1} = \begin{bmatrix} Q_{k-1}^1 & 0 & \cdots & 0 \\ 0 & Q_{k-1}^2 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & Q_{k-1}^N \end{bmatrix}, \quad (14)$$

where $Q_{k-1}^i = J_{u^{(k-1)}}^i \Sigma_u J_{u^{(k-1)}}^{iT}$ and Σ_u is the covariance matrix for controlling input. Then, the time update process of collaborative navigation system in centralized framework can be deduced:

$$X_k^- = \Phi(X_{k-1}^+, u_{k-1}),$$

$$P_k^- = F_{k-1}P_{k-1}F_{k-1}^T + Q_{k-1} \triangleq \begin{bmatrix} P_{11,k}^- & P_{12,k}^- & \cdots & P_{1N,k}^- \\ P_{21,k}^- & P_{22,k}^- & \cdots & P_{2N,k}^- \\ \vdots & \vdots & \ddots & \vdots \\ P_{N1,k}^- & P_{N2,k}^- & \cdots & P_{NN,k}^- \end{bmatrix}, \quad (15)$$

where

$$P_{ii,k}^- = J_{X(k-1)}^i P_{ii,k-1} J_{X(k-1)}^{iT} + Q_{k-1}^i, \quad (16)$$

$$P_{ij,k}^- = J_{X(k-1)}^i P_{ij,k-1} J_{X(k-1)}^{jT}.$$

4.2. Single-Node Measurement Update. In this section, the measurement updating process involving only one node in the centralized framework is described. The aided navigation system selected is SLAM navigation system, which integrates the landmark information of the surrounding environment measured by lidar. In this paper, a measurement model based on this navigation system is built, and the process of measurement updating based on EKF is described.

4.2.1. Measurement Model Based on SLAM. The measurement model based on SLAM is the measurement model after data association. In this paper, the position information of landmarks obtained by lidar is taken as the observation equation.

$$z_k^i = \begin{bmatrix} x_l^b \\ y_l^b \end{bmatrix}$$

$$= \begin{bmatrix} (x_l^w - x_k^i) \cos \theta_k^i + (y_l^w - y_k^i) \sin \theta_k^i \\ -(x_l^w - x_k^i) \sin \theta_k^i + (y_l^w - y_k^i) \cos \theta_k^i \end{bmatrix} + n_k^i, \quad (17)$$

where (x_l^b, y_l^b) is position information for landmarks obtained by lidar. $(x_l^w, y_l^w, \theta_k^i)$ is the coordinates of landmarks in the world coordinate system. (x_k^i, y_k^i) is the state of the mobile robot at the time of k . n_k^i is the measurement noise and its variance matrix is R_k^i , which can be denoted as $n_k^i \sim N(0, R_k^i)$. After linearization and state extension, the observation equations of the whole system can be obtained:

$$z_k^i = H_k^i X_k + h^i(X_k^{i-}) - \nabla h^i X_k^{i-} + n_k^i, \quad (18)$$

where

$$H_k^i = [0 \cdots \nabla h^i \cdots 0]_{2 \times 3N}, \quad (19)$$

and ∇h^i is Jacobian matrices of function $h^i(X_k^i)$.

4.2.2. Measurement and Update Based on EKF. Combined with the basic principle of Kalman filter, the measurement and update process of the aided navigation system for a single node can be obtained as follows:

Step 1: calculating the innovation and the filter gain:

$$v = z_k^i - h^i(X_k^{i-}),$$

$$S^i = H_k^i P_k^- (H_k^i)^T + R_k^i, \quad (20)$$

$$S^i = H_k^i P_k^- (H_k^i)^T + R_k^i$$

Step 2: updating the state estimation and the corresponding covariance:

$$X_k^+ = X_k^- + K^i v, \quad (21)$$

$$P_k = P_k^- - P_k^- (H_k^i)^T (S^i)^{-1} H_k^i P_k^-.$$

4.3. Relative Measurement Update among Nodes. The standard observation model can be divided into two types: the measurement model based on the relative distance and the measurement model based on the relative position.

4.3.1. Measurement Model Based on Relative Distance. The observation of mobile robot i to mobile robot j at k time can be denoted by z_k^{ij} ; then the observation equation is given by

$$z_k^{ij} = h^{ij}(X_k^i, X_k^j) + n_k^{ij}$$

$$= \sqrt{(x_k^i - x_k^j)^2 + (y_k^i - y_k^j)^2} + n_k^{ij}, \quad (22)$$

where n_k^{ij} is the measurement noise, its variance matrix is $R_k^{ij} = \sigma_{\text{UWB}}^2$, which can be denoted as $n_k^{ij} \sim N(0, R_k^{ij})$, and σ_{UWB} is the variance for UWB ranging.

After linearization and state extension, the observation equations of the whole system can be obtained:

$$z_k^{ij} = H_k^{ij} X_k + h^{ij}(X_k^{i-}, X_k^{j-}) - \nabla h_i^{ij} X_k^{i-} - \nabla h_j^{ij} X_k^{j-} + n_k^{ij}, \quad (23)$$

where

$$H_k^{ij} = [0 \cdots \nabla h_i^{ij} \cdots \nabla h_j^{ij} \cdots 0]_{2 \times 3N}, \quad (24)$$

and ∇h_i^{ij} and ∇h_j^{ij} are Jacobian matrices of function $h^{ij}(X^i, X^j)$, respectively.

4.3.2. Measurement Model Based on Relative Position. Using lidar as the sensor to realize the relative observation among nodes can be divided into two kinds: direct method and indirect method. The direct method is to measure the relative position between the two nodes directly; the indirect method is to use lidar to observe the nearest landmark between the two nodes. The relative position between the two nodes is obtained by correlation calculation.

The state of mobile robot i is denoted by $(x_k^i, y_k^i, \theta_k^i)$, at time k and the state of mobile robot j is denoted by $(x_k^j, y_k^j, \theta_k^j)$. The coordinates of landmark L_1 adjacent to

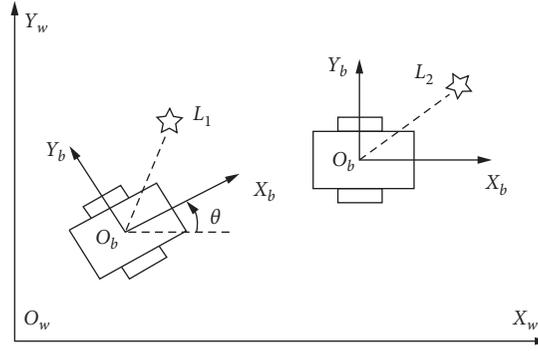


FIGURE 9: Indirect observation schematic diagram.

mobile robot i in the world coordinate system are (x_{l1}^w, y_{l1}^w) , the coordinates in the mobile robot i coordinate system are (x_{l1}^i, y_{l1}^i) , the coordinates of landmark L_2 adjacent to mobile robot j in the world coordinate system are (x_{l2}^w, y_{l2}^w) , and the coordinates in the coordinate system of mobile robot j are (x_{l2}^j, y_{l2}^j) . The specific solution process of the indirect method is as follows (see Figure 9):

$$\begin{bmatrix} x_k^j - x_k^i \\ y_k^j - y_k^i \end{bmatrix} = \begin{bmatrix} x_{l2}^j \cos \theta_k^j - y_{l2}^j \sin \theta_k^j \\ x_{l2}^j \sin \theta_k^j + y_{l2}^j \cos \theta_k^j \end{bmatrix} - \begin{bmatrix} (x_{l1}^i \cos \theta_k^i - y_{l1}^i \sin \theta_k^i) + (x_{l1}^w - x_{l2}^w) \\ (x_{l1}^i \sin \theta_k^i + y_{l1}^i \cos \theta_k^i) + (y_{l1}^w - y_{l2}^w) \end{bmatrix}, \quad (25)$$

when mobile robot i observe mobile robot j at k time, the coordinates of mobile robot j in the mobile robot i coordinate system as the observation. The observation equations are as follows:

$$z_k^{ij} = \begin{bmatrix} x_k^{j,i} \\ y_k^{j,i} \end{bmatrix} = \begin{bmatrix} (x_k^j - x_k^i) \cos \theta_k^i + (y_k^j - y_k^i) \sin \theta_k^i \\ -(x_k^j - x_k^i) \sin \theta_k^i + (y_k^j - y_k^i) \cos \theta_k^i \end{bmatrix} + n_k^{ij}, \quad (26)$$

where n_k^{ij} is the measurement noise, its variance matrix is R_k^{ij} , which can be denoted as $n_k^{ij} \sim N(0, R_k^{ij})$, and $(x_k^{j,i}, y_k^{j,i})$ is the coordinate of mobile robot j in the coordinate system of mobile robot i at k time.

4.3.3. Measurement Update Based on EKF. Similarly, we can finally get the measurement update process for the relative observation between nodes.

5. Decentralized Collaborative Navigation Algorithm

The state and covariance information of each node under the decentralized collaborative navigation algorithm is, respectively, calculated. In order to avoid overoptimal estimation to the maximum extent, the concept of the covariance intersection filter is introduced and the covariance of each node is divided into related and irrelevant items.

5.1. Covariance Intersection Filter. Given the state estimation vector X and corresponding covariance matrix P , assuming

that P^* is the covariance of the error between the state estimate X and the state real value X^* , it can be expressed as follows:

$$P^* = E[(X - X^*)(X - X^*)^T]. \quad (27)$$

Consistency is a characteristic of the covariance matrix of the estimation [30]. When the covariance matrix of the state estimation is not less than the real covariance, it is said that the estimation satisfies the consistency, that is, no overoptimal estimation is produced. Suppose two state estimates X_1 and X_2 are independent and satisfy the consistency, the corresponding covariances are P_1 and P_2 . If there is a correlation between the two estimates, the Kalman filter may produce inconsistent results; in other words, it leads to overoptimal estimation:

$$P_1 = \frac{P_{1d}}{w} + P_{1i}, \quad (28)$$

$$P_2 = \frac{P_{2d}}{w} + P_{2i}, \quad (29)$$

$$P^{-1} = P_1^{-1} + P_2^{-1}, \quad (30)$$

$$X = P(P_1^{-1}X_1 + P_2^{-1}X_2), \quad (31)$$

$$P_i = P(P_1^{-1}P_{1i}P_1^{-1} + P_2^{-1}P_{2i}P_2^{-1}), \quad (32)$$

$$P_d = P - P_i, \quad (33)$$

where the covariance corresponding to two state estimates X_1 and X_2 is $P_{1d} + P_{1i}$ and $P_{2d} + P_{2i}$, respectively. P_{1d} and P_{2d} are the correlation covariance components corresponding to the maximum correlation between the two state estimates. P_{1i} and P_{2i} are independent covariance components corresponding to absolute independence of two state estimates, W within interval $[0, 1]$. It is an optimization parameter that minimizes the covariance after fusion, and the w in this interval can ensure the consistency of the fusion results.

5.2. Time Update. Before describing the time update process of DCL algorithm, it is necessary to decompose the state information of the system in the framework of centralized collaborative navigation, which can be expressed as

$$\begin{aligned} E_G &= \{X_G, P_G\} \\ &= \{\{X_1, P_1\}; \{X_2, P_2\} \cdots \{X_N, P_N\}\} \\ &= \{\{X_1, P_{1d} + P_{1i}\}; \{X_2, P_{2d} + P_{2i}\} \cdots \{X_N, P_{Nd} + P_{Ni}\}\}, \end{aligned} \quad (34)$$

where E_G is the set of states under the centralized collaborative navigation framework, and X_G and P_G are the state space and the corresponding covariance matrix under the centralized collaborative navigation framework, respectively.

The state propagation process under the decentralized collaborative navigation framework is the state propagation process of a single node, and the propagation process of covariance can be expressed as

$$\begin{aligned} P_k^{i-} &= J_{X(k-1)}^i P_{k-1}^i J_{X(k-1)}^{iT} + J_{u(k-1)}^i \Sigma_u J_{u(k-1)}^{iT}, \\ P_{k,i}^{i-} &= J_{X(k-1)}^i P_{k-1,i}^i J_{X(k-1)}^{iT} + J_{u(k-1)}^i \Sigma_u J_{u(k-1)}^{iT}, \end{aligned} \quad (35)$$

where P_k^{i-} is the one-step prediction covariance matrix of the mobile robot i at time k and $P_{k,i}^{i-}$ is the covariance. The independent covariance components of the matrix, $J_{X(k-1)}^i$ and $J_{u(k-1)}^i$, are Jacobian matrices of function $f^i(X, u)$ for state vector and control input and Σ_u is the error matrix of the control input.

5.3. Single Node Measurement Update. The measurement and updating process of a single node only involves the aided navigation system of a single node, so there is no need to estimate the correlation, that is, the process of saving formulas (28) and (29). Similar to the single node measurement update process in centralized collaborative navigation, the single node measurement update process in distributed collaborative navigation can be expressed as

Step 1: calculate the innovation and the filtering gain:

$$\begin{aligned} v &= z_k^i - \nabla h^i X_k^{i-}, \\ S &= \nabla h^i P_k^{i-} (\nabla h^i)^T + R_k^i, \\ K &= P_k^{i-} (\nabla h^i)^T S^{-1}. \end{aligned} \quad (36)$$

Step 2: update the state estimation and the corresponding covariance:

$$\begin{aligned} X_k^{i+} &= X_k^{i-} + Kv, \\ P_k^i &= (I - K\nabla h^i) P_k^{i-}, \\ P_{k,i}^i &= (I - K\nabla h^i) P_{k,i}^{i-} (I - K\nabla h^i)^T + KR_k^i K^T, \\ P_{k,d}^i &= P_k^i - P_{k,i}^i. \end{aligned} \quad (37)$$

5.4. Collaborative Measurement Update among Nodes. In the framework of decentralized collaborative navigation, the state estimation results of a single node aided navigation system and the state estimation results based on information sharing among nodes are integrated in the process of internode collaborative measurement updating, and the corrected state information is derived.

For the decentralized collaborative navigation framework, any node can clearly estimate the state of other nodes. In order to save the communication cost and reduce the computation of a single mobile robot platform, this paper sets up that the information exchange among the mobile robots is only taking place between the two adjacent mobile robot nodes.

Assuming that the mobile robot i performs relative observations of the mobile robot j at the k time and shares its own state and covariance information with the mobile robot j , the state of the mobile robot j can be clearly expressed with information received state of the mobile robot i and the relative measurement information between the two nodes:

$$X_k^{\text{coj}} = \begin{bmatrix} x_k^{\text{coj}} \\ y_k^{\text{coj}} \end{bmatrix} = \begin{bmatrix} x_k^i + x_k^{j,i} \cos \theta_k^i - y_k^{j,i} \sin \theta_k^i \\ y_k^i + x_k^{j,i} \sin \theta_k^i + y_k^{j,i} \cos \theta_k^i \end{bmatrix}, \quad (38)$$

where $(x_k^{\text{coj}}, y_k^{\text{coj}})$ is the partial state estimation of the mobile robot j obtained by the information sharing between the mobile robot i and the mobile robot j at k time, $(x_k^i, x_k^{j,i}, \theta_k^i)$ is the state vector shared by the mobile robot j in the k direction of the mobile robot i , and $u_{\text{rel}} = (x_k^{j,i}, y_k^{j,i})$ is the relative measurement information of the two nodes in the i coordinate system of the mobile robot.

If there is a direct relative observation between the two nodes, the relative measurement information can be obtained directly by the sensor that carries on the relative observation. If the relative observation between the two nodes is with the help of the indirect relative observation of the surrounding landmark information, then the relative measurement information needs to be solved to a certain extent, and the concrete solution method can be combined (25) and then converted into the mobile robot.

Finally, based on the principle of covariance intersection filter, the updating process of collaborative measurement among nodes in the framework of decentralized collaborative navigation can be obtained.

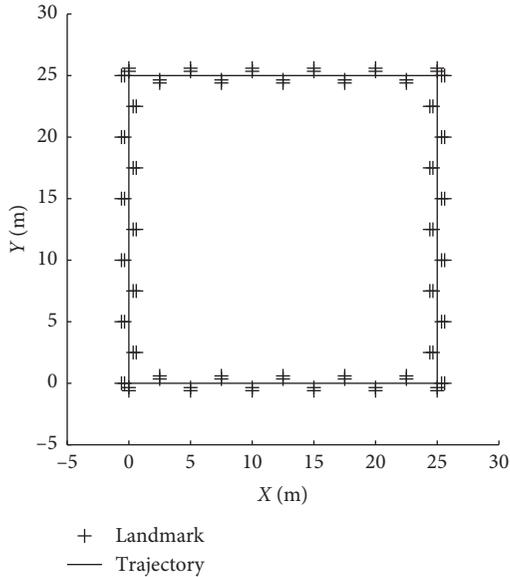


FIGURE 10: Simulation trajectory diagram.

6. Simulation Results

6.1. Simulated Experimental Environment. In this section, the nodes of the mobile robot network involved in collaborative navigation are 3. Assuming that the area of the moving 2D environment is $25\text{ m} \times 25\text{ m}$ when the mobile robot population group works together, each node in the environment is assigned to an initial position, and each node can follow random trajectory motion in this area. Assuming that all nodes follow the same simulation trajectory, only the initial position is different. The maximum speed of the mobile robot in the straight line is 0.3125 m/s and the angular velocity at the bend is $0.1^\circ/\text{s}$. It is assumed that the environment around the simulated trajectory of this rectangle can be extracted by lidar scanning to 88 landmarks (environmental feature points) for the SLAM-assisted navigation system (see Figure 10).

During this simulation, mobile robots as carriers can carry different types of sensors, including odometers, UWB, and lidar. Suitable sensors are selected according to the requirements of positioning accuracy, among which Time Domain P410 UWB sensors are used to measure the relative distance, and lidar is selected from LMS291 series 2D lidar produced by a German company. Based on the relevant parameters of these sensors, which are shown in Table 1, a simulation model for mobile robots carrying different types of sensors is built using MATLAB.

6.2. Relative Measurement Aided Odometer Collaborative Navigation. In the experiment, all three mobile robots are equipped with odometer capable of moving monitoring, UWB capable of measuring relative distance, or lidar capable of measuring relative position.

From Figure 11, it can be seen that the collaborative navigation system which realizes relative information sharing has significant advantages over the case of not

TABLE 1: Relevant parameters of sensors.

| Type | Measure | Frequency (Hz) | Error |
|----------|-----------------------------------|----------------|--------|
| Odometer | Linear velocity of the two wheels | 20 | 4 cm/s |
| UWB | Relative distance | 10 | 3 cm |
| Lidar | Relative position of X direction | 10 | 2 cm |
| | Relative position of Y direction | 10 | 2 cm |

sharing any information in positioning accuracy. Besides, the improvement of group navigation performance of mobile robots is affected by the type of shared relative information. When the relative position information is shared, the growth of the error can be effectively limited; relatively speaking, when the relative distance information is shared, the position error is still growing slowly, which only reduces the growth rate of the error (see Figure 11).

The analysis shows that the relative distance information is weakly constrained, so sharing this information cannot effectively realize the navigation and localization of mobile robots. In contrast, the sharing of relative position information includes the solution to mobile robot navigation and information. Information accuracy is significantly improved. It can even be increased by more than 60% at some time. This difference is more obvious in the angle error diagram (see Figure 11).

In this paper, two observation methods, direct relative measurement and indirect relative measurement, are mentioned in the description of the measurement model based on relative position. Based on this experimental scene, scenario I is three mobile robots to observe the relative position information directly through lidar. Scenario II is three mobile robots to extract the surrounding landmark information through lidar, and based on this solution, the relative position information is calculated. In the above experimental scenario, the centralized collaborative navigation algorithm is used to solve the navigation problem. The two relative position measurement methods are compared through the above simulation experimental scenarios. The comparison results are shown in Figure 12 (see Figure 12).

Through Figure 12, it is clear that the collaborative navigation and positioning accuracy of relative position measurement using the direct method are better than those of indirect method. However, cost calculation cannot be ignored while navigation performance is considered. The direct method requires that the measurement range of lidar includes the activity range of the whole mobile robot population group while the measurement range of lidar required by indirect method only needs to include the surrounding landmarks. This greatly reduces the cost calculation. Considering that the accuracy of collaborative navigation and positioning using the two relative position measurement methods is not much different, it is obvious that the indirect method is more suitable for practical application (see Figure 12).

The difference of the decentralized collaborative navigation framework compared with the centralized collaborative navigation framework is that the correlation among

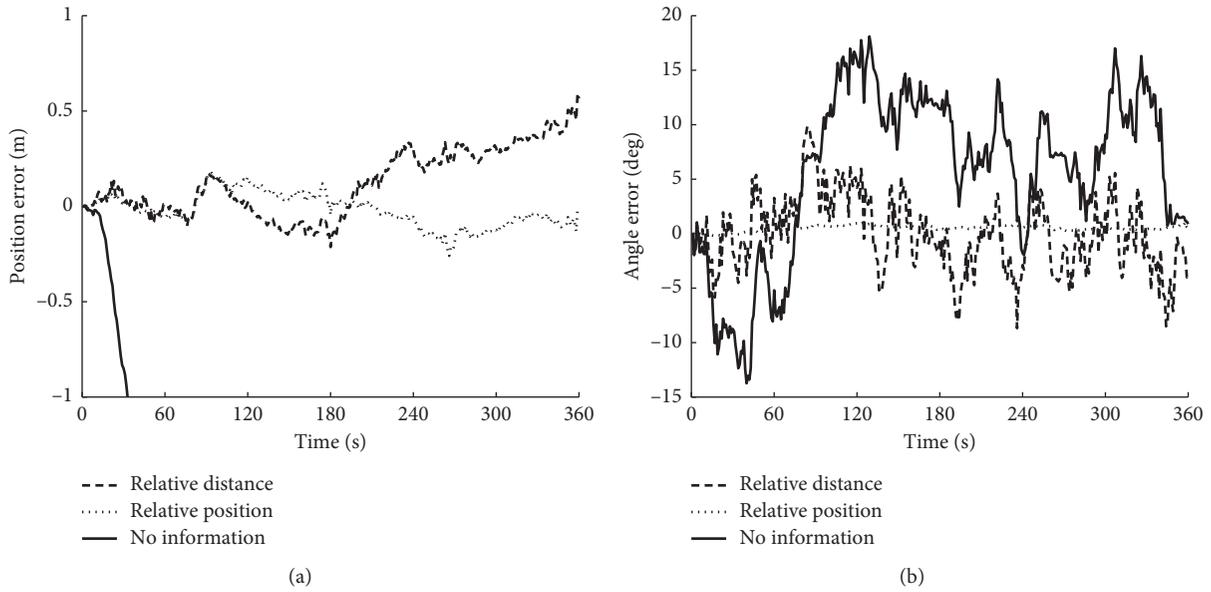


FIGURE 11: Comparative diagram of navigation error under the condition of sharing different relative information. (a) Position error. (b) Angle error.

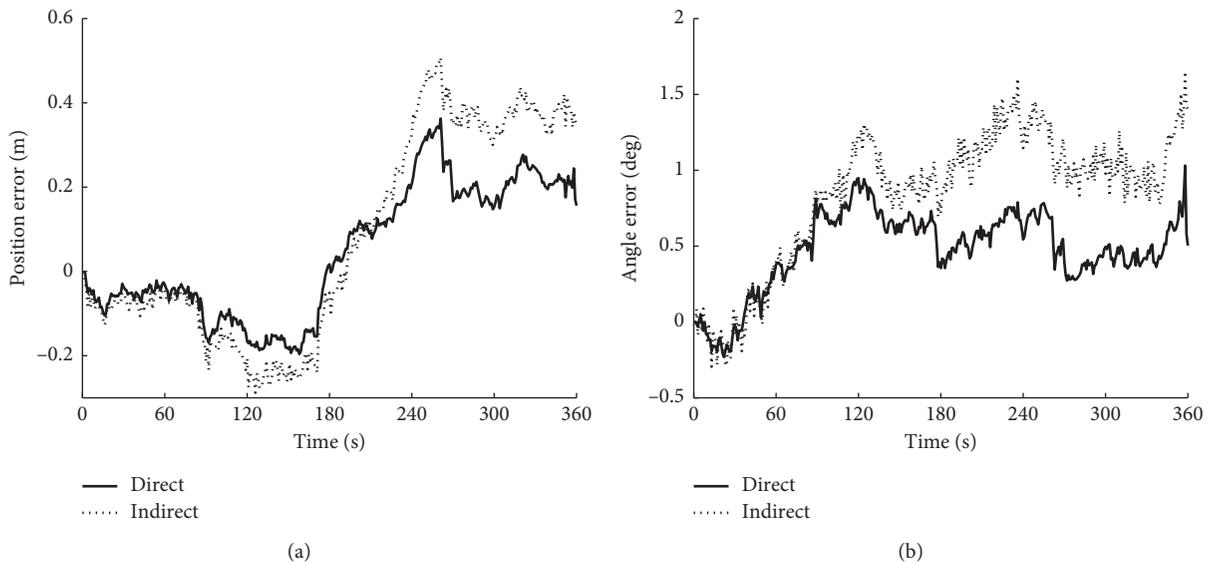


FIGURE 12: Comparison of navigation errors under different relative position measuring methods. (a) Position error. (b) Angle error.

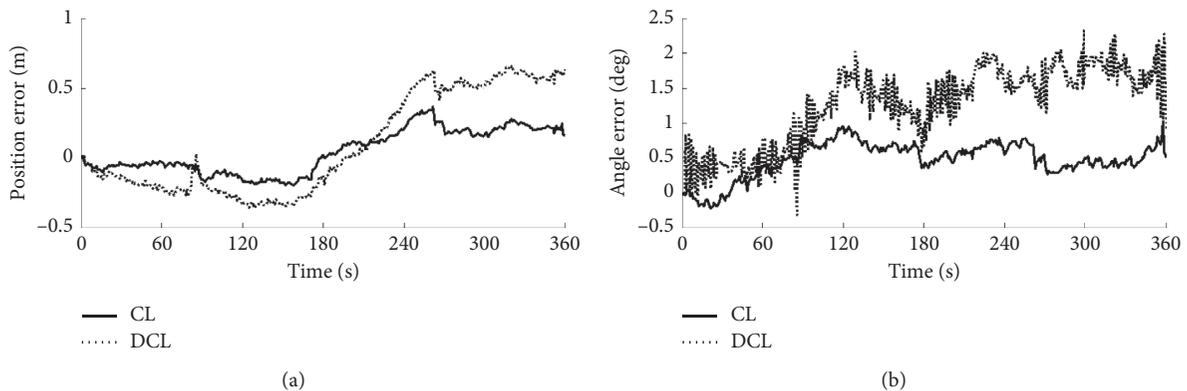


FIGURE 13: Comparison of navigation errors under different collaborative navigation algorithms. (a) Position error. (b) Angle error.

the different node states is accurately calculated in the centralized collaborative navigation framework, and this correlation cannot be used in the decentralized collaborative navigation framework. In order to better reflect the impact of this correlation, the navigation errors of the two collaborative navigation algorithms in the odometer collaborative navigation system are shown in Figure 13 (see Figure 13).

To compare the two algorithms, 20 experiments are carried out in this paper, and the root mean square error (RMS) and formulas of the two collaborative navigation algorithms are calculated as shown in the following formula:

$$\text{RMS} = \sqrt{\frac{1}{n} \sum_{i=1}^n (x_i - \bar{x}_i)^2}, \quad (39)$$

where n is the total number of samples, x_i is the actual value, and \bar{x}_i is the estimated value. RMS parameters for the odometer collaborative navigation are shown in Table 2.

As can be seen from Figure 13 and Table 2, the error of the centralized collaborative navigation algorithm is smaller than that of the decentralized collaborative navigation algorithm. This can be predicted because the correlation among node states in the centralized collaborative navigation algorithm can be calculated accurately, which is estimated in the decentralized collaborative navigation algorithm. However, the improved accuracy of the navigation is at the expense of high computing power and high quality data communication. Therefore, although the performance of centralized collaborative navigation framework is better than that of distributed collaborative navigation framework, the centralized collaborative navigation framework is not applicable in some practical scenarios (see Figure 13).

6.3. Odometer/Vision SLAM Collaborative Navigation. In the odometer/vision collaborative navigation model, scenario I is designed that all the mobile robots are equipped with an odometer which can monitor the motion. One of the mobile robots is equipped with SLAM-aided navigation system and can work properly.

Firstly, the mobile robot with SLAM-aided navigation system is studied, and it only runs its own integrated navigation algorithm without sharing the relative position information. Using the centralized collaborative navigation algorithm, the navigation error of nodes with SLAM-aided navigation system is shown in Figure 14 (see Figure 14).

Figure 14 fully verifies the correctness of a centralized collaborative navigation algorithm based on the odometer/vision collaborative navigation model. The SLAM-assisted navigation system is based on the relative observation. The position estimation of the node itself and the position estimation of the landmark have the error accumulation, but the association algorithm of the SLAM is combined, the centralized collaborative navigation algorithm makes the position estimation of the landmark closer to the real value while the positioning accuracy of the mobile robot is

TABLE 2: Odometer collaborative navigation RMS parameters.

| Algorithm type | Position error (m) | Angle error (°) |
|----------------|--------------------|-----------------|
| CL | 0.1629 | 0.74625 |
| DCL | 0.36342 | 1.3762 |

improved, the data association is more reliable, further correcting the state estimation of the mobile robot itself. Therefore, the algorithm is very obvious to improve the navigation accuracy of mobile robot (see Figure 14).

Then, the mobile robots without the SLAM-aided navigation system in the experiment are studied. In order to fully reflect the influence of the SLAM-aided navigation information on the navigation performance of other nodes, Scenario II is designed that all mobile robots are equipped with odometer which can monitor the motion, and two of them are equipped with SLAM-aided navigation system and can work properly. The navigation error of other nodes without SLAM-aided navigation system is shown in Figure 15 (see Figure 15).

As shown in Figure 15, the mobile robot with SLAM-aided navigation system performs loop-back detection in about 320 seconds and data associated with the local map created at the initial location, thus eliminating most of the accumulated errors. The unique superior performance of the SLAM-aided navigation system is transmitted to other nodes in the group through the process of information sharing in the process of collaborative navigation, so that it can also eliminate most of the accumulated errors in the vicinity of the time, which is an important advantage of the collaborative navigation system (see Figure 15).

To verify the NN algorithm, JBCC algorithm, and the optimized data association algorithm on the navigation performance of nodes without SLAM-aided navigation system, the experimental scene is designed that all mobile robots are equipped with odometer which can carry out motion monitoring. One of the mobile robots is equipped with SLAM-aided navigation system and can work normally, and the CL algorithm is run. The navigation error of nodes without SLAM-aided navigation system is shown in Figure 16 (see Figure 16).

The performance of the centralized collaborative navigation algorithm under the three SLAM data association algorithms is shown in Table 3.

From Figure 16 and Table 3, it can be seen that the navigation performance of nodes without SLAM-aided navigation system is affected by the SLAM data association algorithm used by nodes carrying SLAM-aided navigation system. Running the NN algorithm, the matching accuracy of feature information is not high, so that the navigation accuracy is poor. Running the JCBB algorithm, the correct rate of data association is the highest, but the operation time is the longest. Running optimization data association algorithm, the navigation accuracy is slightly reduced, but the operation time is less, which can meet the real-time requirements (see Figure 16).

In this subsection, to compare the performance of collaborative navigation systems in odometer/vision collaborative navigation systems under centralized and

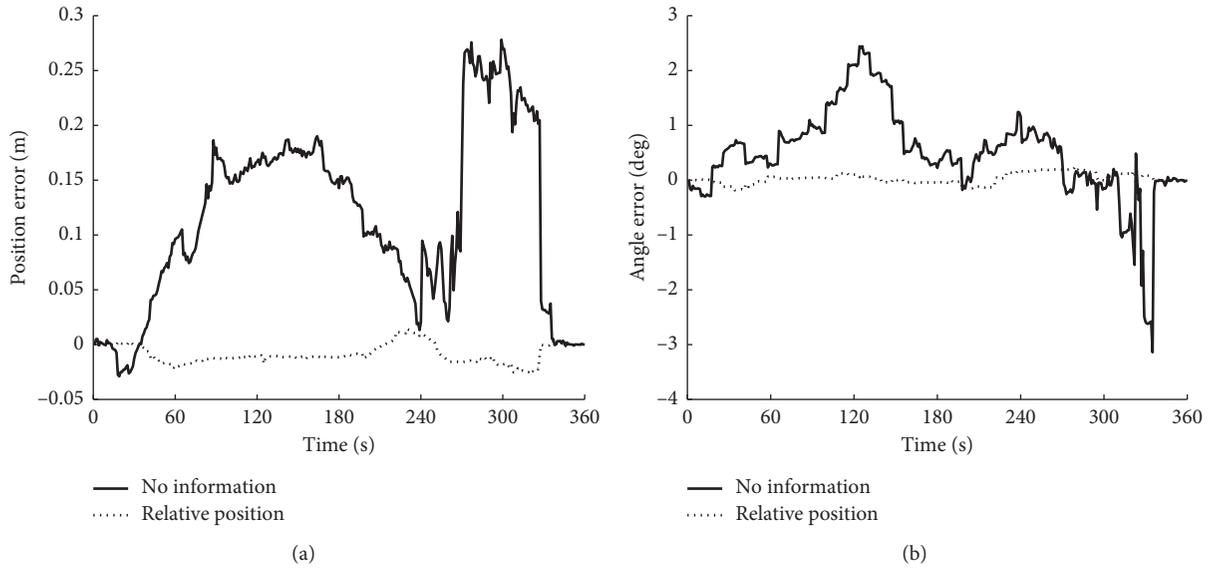


FIGURE 14: The navigation error map of the node with the SLAM-aided navigation system. (a) Position error. (b) Angle error.

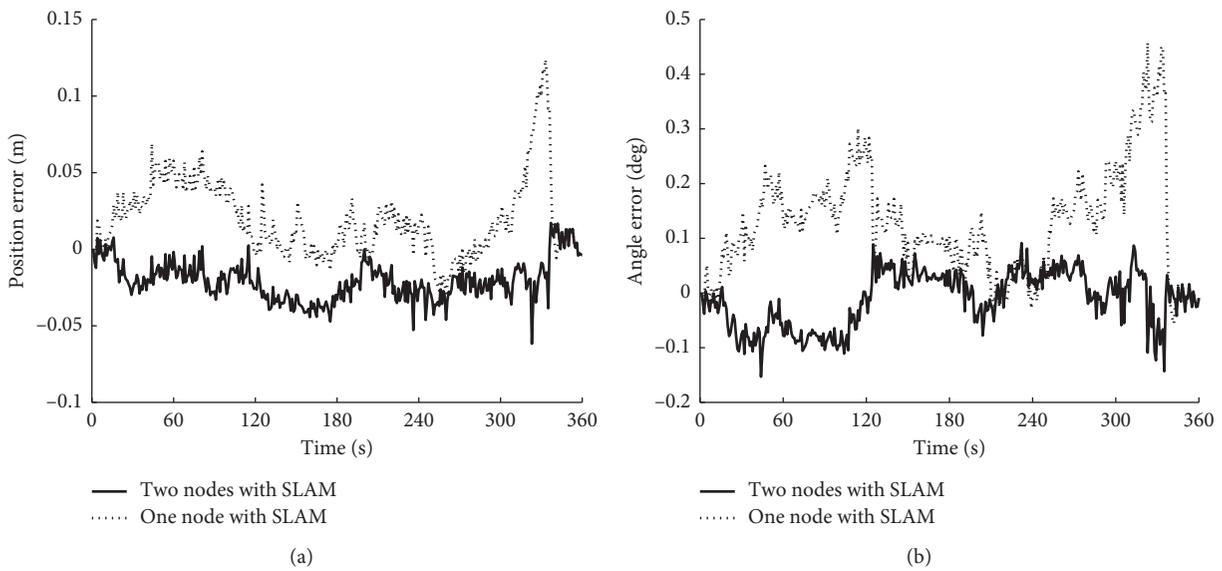


FIGURE 15: Some nodes are equipped with SLAM-aided navigation system. (a) Position error. (b) Angle error.

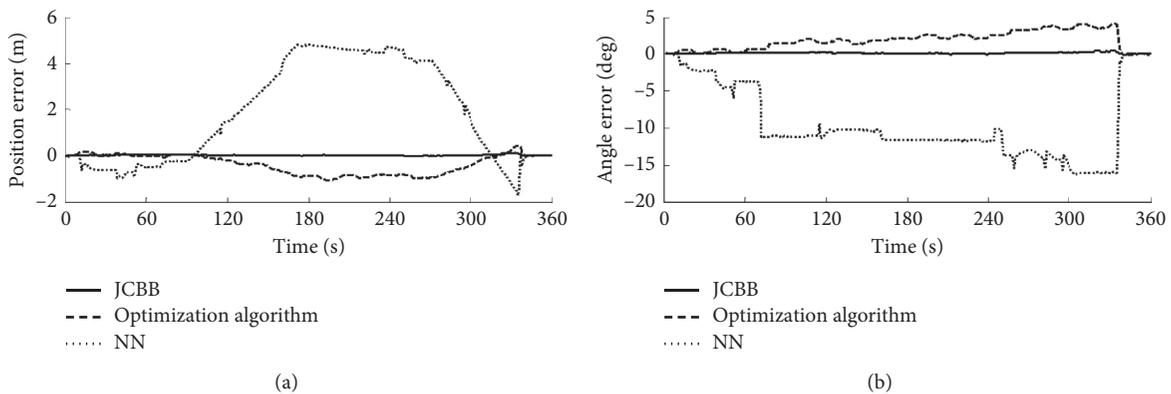


FIGURE 16: Comparison diagram of navigation error for fusion of single node SLAM information under different SLAM data association algorithms. (a) Position error. (b) Angle error.

TABLE 3: Performance comparison of centralized collaborative navigation algorithms under different SLAM data association algorithms.

| Algorithm type | Position error (m) | Angle error (°) | Relative time |
|----------------|--------------------|-----------------|---------------|
| NN | 2.8323 | 10.7919 | 4 |
| JCBB | 0.0322 | 0.1623 | 12 |
| Optimization | 0.5587 | 2.2476 | 1 |

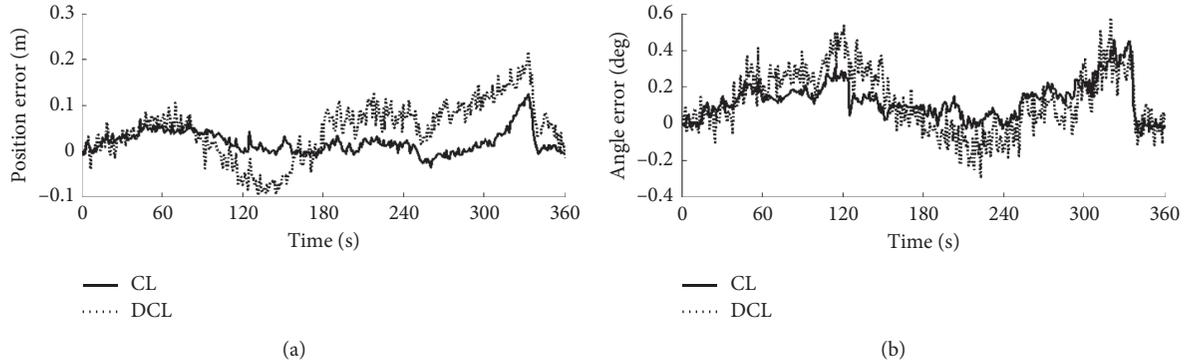


FIGURE 17: Comparative diagram of navigation error for fusion of single node SLAM information under different collaborative navigation algorithms. (a) Position error. (b) Angle error.

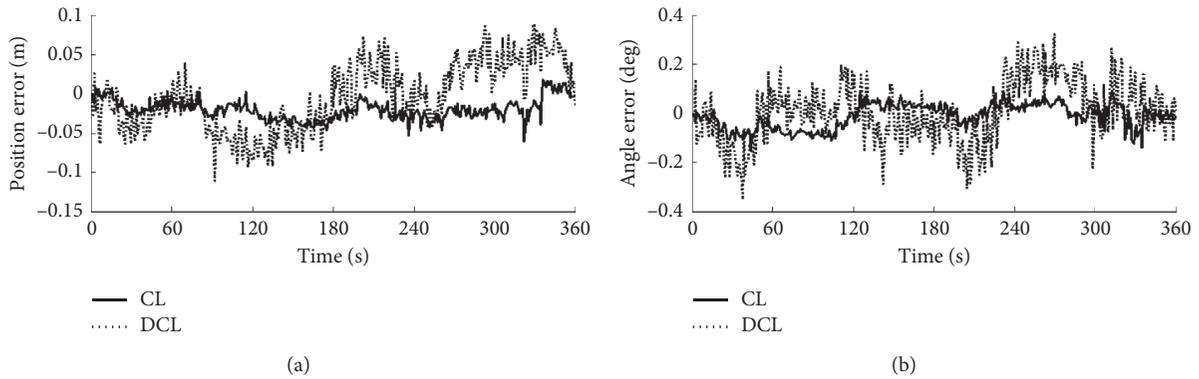


FIGURE 18: Comparison diagram of navigation error for fusion of multinode SLAM information under different collaborative navigation algorithms. (a) Position error. (b) Angle error.

TABLE 4: Collaborative navigation RMS parameters for fusion of single node SLAM information.

| Algorithm type | Position error (m) | Angle error (°) |
|----------------|--------------------|-----------------|
| CL | 0.0322 | 0.1623 |
| DCL | 0.0669 | 0.2094 |

decentralized collaborative navigation algorithms, we run the CL and DCL algorithms separately under the experimental scenario I. The navigation errors of the two collaborative navigation algorithms are compared as shown in Figure 17. Under the experimental scenario II of this subsection, we run the CL algorithm and the DCL algorithm, respectively. The navigation errors of the two collaborative navigation algorithms are compared as shown in Figure 18 (see Figures 17 and 18).

TABLE 5: Collaborative navigation RMS parameters for fusion of multinode SLAM information.

| Algorithm type | Position error (m) | Angle error (°) |
|----------------|--------------------|-----------------|
| CL | 0.0243 | 0.0524 |
| DCL | 0.0438 | 0.1265 |

After 20 experiments, the RMS parameters of collaborative navigation with single node SLAM information are shown in Table 4.

The RMS parameters of the coordinated navigation with the fused multinode SLAM information are shown in Table 5.

As can be seen from Figures 17 and 18 in conjunction with Tables 4 and 5, in the odometer/vision collaborative navigation system, the error of the centralized collaborative

navigation algorithm is less than the distributed collaborative navigation algorithm; after the landmark information collected by the single node or the multinode is fused, there is a small gap between the two algorithms. In other words, the distributed collaborative navigation algorithm based on the odometer/vision collaborative navigation model can well estimate the correlation of the internode information (see Figures 17 and 18).

Considering the high requirement of the centralized collaborative navigation algorithm to the computing power and the communication level, the application scenarios of the two algorithms are analyzed in combination with the abovementioned collaborative navigation experiment: the centralized collaborative navigation algorithm is suitable for the case that there are few nodes and the nodes are not equipped with additional aided navigation system, the decentralized collaborative navigation algorithm is suitable for the large number of nodes and the large amount of information shared, and some nodes are equipped with additional aided navigation systems, especially in the case of SLAM-aided navigation system.

7. Conclusion

In order to improve the performance of cooperative navigation system, a multirobot collaborative navigation algorithm based on odometer/vision multisource information fusion is studied. On the basis of establishing the multisource information fusion collaborative navigation system model, the centralized collaborative navigation of odometer/vision fusion, the decentralized collaborative navigation framework, and the vision-based SLAM are given, and the centralized and decentralized odometer/vision collaborative navigation algorithms are derived, respectively. The effectiveness of the proposed algorithm is verified by the simulation experiments, which has some theoretical value and application value in high performance collaborative navigation applications.

Data Availability

The data used to support the findings of this study are available from the corresponding author upon request.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

References

- [1] K. N. Olivier, D. E. Griffith, G. Eagle et al., "Randomized trial of liposomal amikacin for inhalation in nontuberculous mycobacterial lung disease," *American Journal of Respiratory and Critical Care Medicine*, vol. 195, no. 6, pp. 814–823, 2017.
- [2] M. Schwarz, M. Beul, D. Droeschel et al., "DRC team nimbro rescue: perception and control for centaur-like mobile manipulation robot momaro," *Springer Tracts in Advanced Robotics*, Springer, Berlin, Germany, pp. 145–190, 2018.
- [3] M. Long, H. Su, and B. Liu, "Group controllability of two-time-scale discrete-time multi-agent systems," *Journal of the Franklin Institute*, vol. 357, no. 6, pp. 3524–3540, 2020.
- [4] T. Fukuda, S. Nakagawa, Y. Kawauchi, and M. Buss, "Structure decision method for self organising robots based on cell structures-CEBOT," in *Proceedings of the 1989 International Conference on Robotics and Automation*, Scottsdale, AZ, USA, May 1989.
- [5] H. Asama, A. Matsumoto, and Y. Ishida, "Design of an autonomous and distributed robot system: actress," in *Proceedings of the IEEE/RSJ International Workshop on Intelligent Robots and Systems "(IROS '89) the Autonomous Mobile Robots and its Applications*, September 1989.
- [6] J. Zhou, Y. Lv, G. Wen, X. Wu, and M. Cai, "Three-dimensional cooperative guidance law design for simultaneous attack with multiple missiles against a maneuvering target," in *Proceedings of the 2018 IEEE CSAA Guidance, Navigation and Control Conference (CGNCC)*, August 2018.
- [7] H. Su, J. Zhang, and Z. Zeng, "Formation-containment control of multi-robot systems under a stochastic sampling mechanism," *Science China Technological Sciences*, vol. 63, no. 6, pp. 1025–1034, 2020.
- [8] H. Park and S. Hutchinson, "A distributed robust convergence algorithm for multi-robot systems in the presence of faulty robots," in *Proceedings of the 2015 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pp. 2980–2985, IEEE, Hamburg, Germany, September–October 2015.
- [9] K. Petersen and R. Nagpal, "Complex design by simple robots: a collective embodied intelligence approach to construction," *Architectural Design*, vol. 87, no. 4, pp. 44–49, 2017.
- [10] L. Chaimowicz, T. Sugar, V. Kumar, and M. F. M. Campos, "An architecture for tightly coupled multi-robot cooperation," in *Proceedings 2001 ICRA. IEEE International Conference on Robotics and Automation (Cat. no. O1CH37164)*, vol. 3, pp. 2992–2997, IEEE, Seoul, Korea, May 2001.
- [11] H.-X. Hu, G. Chen, and G. Wen, "Event-triggered control on quasi-average consensus in the cooperation-competition network," in *Proceedings of the IECON 2018—44th Annual Conference of the IEEE Industrial Electronics Society*, October 2018.
- [12] A. Amanatiadis, K. Charalampous, I. Kostavelis et al., "The avert project: autonomous vehicle emergency recovery tool," in *Proceedings of the 2013 IEEE International Symposium on Safety, Security, and Rescue Robotics (SSRR)*, pp. 1–5, IEEE, Linköping, Sweden, October 2013.
- [13] R. Kurazume, S. Hirose, T. Iwasaki, S. Nagata, and N. Sashida, "Study on cooperative positioning system," in *Proceedings of the IEEE International Conference on Robotics and Automation*, Minneapolis, MN, USA, August 1996.
- [14] Z. Fu, Y. Zhao, and G. Wen, "Distributed continuous-time optimization in multi-agent networks with undirected topology," in *Proceedings of the 2019 IEEE 15th International Conference on Control and Automation (ICCA)*, November 2019.
- [15] Y. Zhao, Y. Liu, and G. Wen, "Finite-time average estimation for multiple double integrators with unknown bounded inputs," in *Proceedings of the 2018 33rd Youth Academic Annual Conference of Chinese Association of Automation (YAC)*, May 2018.
- [16] S. Mao, *Mobile robot localization in indoor environment*, Zhejiang University, Hangzhou, China, Ph.D. dissertation, 2016.
- [17] J. Yang, "Analysis approach to odometric non-systematic error uncertainty for mobile robots," *Chinese Journal of Mechanical Engineering*, vol. 44, no. 8, pp. 7–12, 2008.
- [18] J. Kang, F. Zhang, and X. Qu, *Angle Measuring Error Analysis of Coordinate Measuring System of Laser Radar*, vol. 40, no. 6, pp. 834–839, 2016.

- [19] J. Zhang, P. Orlik, Z. Sahinoglu, A. Molisch, and P. Kinney, "UWB systems for wireless sensor networks," *Proceedings of the IEEE*, vol. 97, no. 2, pp. 313–331.
- [20] D. Kaushal and T. Shanmuganatham, "Design of a compact and novel microstrip patch antenna for multiband satellite applications," *Materials Today: Proceedings*, vol. 5, no. 10, pp. 21 175–21 182, 2018.
- [21] J. Xiucui, *Data association problem for simultaneous localization and mapping of mobile robots*, National University of Defense Technology, Changsha, China, Ph.D. dissertation, 2008.
- [22] Z. Yuan, "Research of mobile robot's slam based on binocular vision," Master's thesis, Tianjin University of Technology, Tianjin, China, 2016.
- [23] F. Bellavia, M. Fanfani, F. Pazzaglia, and C. Colombo, "Robust selective stereo slam without loop closure and bundle adjustment," in *Proceedings of the International Conference on Image Analysis and Processing*, pp. 462–471, Springer, Naples, Italy, 2013.
- [24] H. Fourati, *Multisensor Data Fusion: From Algorithms and Architectural Design to Applications*, CRC Press, Boca Raton, FL, USA, 2015.
- [25] S. Jia, X. Yin, and X. Li, "Mobile robot parallel pf-slam based on openmp," in *Proceedings of the 2012 IEEE International Conference on Robotics and Biomimetics (ROBIO)*, pp. 508–513, IEEE, Guangzhou, China, December 2012.
- [26] W. Zhou, E. Shiju, Z. Cao, and Y. Dong, "Review of slam data association study," in *Proceedings of the 2016 International Conference on Sensor Network and Computer Engineering*, Atlantis Press, Shanghai, China, 2016.
- [27] R. Singer and R. Sea, "A new filter for optimal tracking in dense multitarget environments," in *Proceedings of the Annual Allerton Conference on Circuit and System Theory*, pp. 201–211, Monticello, MN, USA, 1972.
- [28] J. Neira and J. D. Tardós, "Data association in stochastic mapping using the joint compatibility test," *IEEE Transactions on Robotics and Automation*, vol. 17, no. 6, pp. 890–897, 2001.
- [29] L. Yanju, X. Yufeng, G. Song, H. Xi, and G. Zhengping, "Research on data association in slam based laser sensor," *Microcomputer & Its Application*, vol. 36, no. 2, pp. 78–82, 2017.
- [30] O. Hlinka, O. Slučiak, F. Hlawatsch, and M. Rupp, "Distributed data fusion using iterative covariance intersection," in *Proceedings of the 2014 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 1861–1865, IEEE, Florence, Italy, May 2014.