

Article

Decentralized Cooperative Localization with Fault Detection and Isolation in Robot Teams

Mei Wu ¹ , Hongbin Ma ^{1,*} and Xinghong Zhang ²

¹ State Key Laboratory of Intelligent Control and Decision of Complex Systems, Beijing Institute of Technology, Beijing 100081, China; 3120150402@bit.edu.cn

² Department of Automatic Control, Henan Institute of Technology, Xinxiang 453000, China; hongxing0603@163.com

* Correspondence: mathmhb@bit.edu.cn; Tel.: +86-106-891-3985

Received: 3 August 2018; Accepted: 18 September 2018; Published: 8 October 2018



Abstract: Robot localization, particularly multirobot localization, is an important task for multirobot teams. In this paper, a decentralized cooperative localization (DCL) algorithm with fault detection and isolation is proposed to estimate the positions of robots in mobile robot teams. To calculate the interestimate correlations in a distributed manner, the split covariance intersection filter (SCIF) is applied in the algorithm. Based on the split covariance intersection filter cooperative localization (SCIFCL) algorithm, we adopt fault detection and isolation (FDI) to improve the robustness and accuracy of the DCL results. In the proposed algorithm, the signature matrix of the original FDI algorithm is modified for application to DCL. A simulation-based comparative study is conducted to demonstrate the effectiveness of the proposed algorithm.

Keywords: robot localization; mobile robot; distributed cooperative localization; fault detection and isolation

1. Introduction

Robots have become increasingly prevalent in many domains, such as medical services and security [1]. Tasks assigned to robots such as rescue [2] are becoming increasingly complex for a single robot to address, making multirobot cooperation imperative. The cooperation intrinsic to a multirobot team makes it possible to execute tasks such as rescue which may be difficult or difficult with a single, highly capable robot; moreover, the presence of more than one robot makes the team fault-tolerant. The net effect is that a robot team can reliably execute its task even when its working environment changes.

Accurate position information is very important for cooperation in a multirobot team. For a single robot, the localization process is performed based only on its own sensor data, such as Global Positioning System (GPS) data [3], camera data [4], or laser scanner data [5,6]. In a multirobot team, however, sensor data collected from inter-robot observations can be considered in the process of cooperative multirobot localization. It is believed that cooperative localization (CL) methods using sensor data from inter-robot observations will outperform traditional single-robot localization methods [7].

The idea of CL was first proposed in [8], in which the robot team was considered to be divided into two groups. Each group alternately remained stationary to serve as a landmark for the other group. In [9], the premise of the stationarity of one group of robots is canceled. To date, many CL algorithms have been proposed. Among these algorithms, the simplest is centralized cooperative localization (CCL) [10]. In CCL, all information, including observation information and position information from the whole team, is transmitted to a fusion center, which maintains only one global state, as shown in Figure 1.

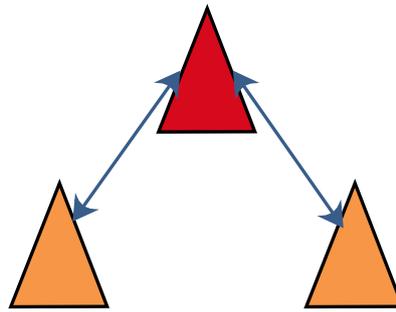


Figure 1. Communication graph for CCL.

However, CCL has clear disadvantages. First, the communication and computational costs of CCL are very high since all computational loads are concentrated at the fusion center. Second, if a fault occurs in the fusion center, then the whole CCL system will be corrupted. Finally, given the finite communication and computational capabilities of a practical system, it is difficult to continue to use only one fusion center to achieve CL for the entire team as the size of the team increases.

Decentralized cooperative localization (DCL) has proven to be a desirable solution [11] to resolve these problems, because both the computational and communication burdens placed on each robot in the team are lower. In DCL, each robot handles only its own local information. Moreover, in this paper, to calculate the interestimate correlations in a distributed manner, the split covariance intersection filter (SCIF) [12] is applied. In this algorithm, two covariance states, namely, independent covariance and dependent covariance, are maintained in each robot.

To improve the robustness of the robot team, fault detection and isolation (FDI) [13–15] are needed. In [16], an FDI algorithm was applied in CCL. However, the FDI algorithms cited above are designed for CCL and are not applicable in DCL due to the limited sensor information available. To improve the performance of fault detection, a decision algorithm that maximizes the amount of information transformation is adopted, as introduced in Section 3. To enable the application of FDI in DCL, the isolation rule is modified. The signature matrix is simplified by combining certain fault types, and a new Kullback–Leibler (KL) residual is added.

The remainder of this paper is organized as follows. The theoretical background of the DCL and FDI algorithm is introduced in Section 2. We describe the concepts of DCL with the SCIF and FDI and introduce the corresponding algorithms in Section 3. The experiments and results are introduced in Section 4. The last section concludes the paper.

2. Background

In this section, some background knowledge is introduced.

Four assumptions are made in this paper. First, each robot is assumed to be equipped with one motion sensor. The equipped sensors can report the motion data on of the robots. No single-robot localization method based on other sensors is considered in these simulations.

Second, the robots are assumed to be able to capture the relative poses of other robots within their range of perception. A robot that is perceived is called a neighboring robot in this paper. Each robot can share its relative measurements and motion states with its neighboring robots via inter-robot communication.

Third, two types of noises are considered in the simulations: additive white Gaussian noise and impulse noise. Here, additive white Gaussian noise consists of unbiased normally distributed errors with a fixed standard deviation, as shown in Figure 2a, whereas impulse noise consists of unpredictable errors that simulate certain unexpected situations, such as poor odometric measurements and noisy and distorted measurements, as shown in Figure 2b.

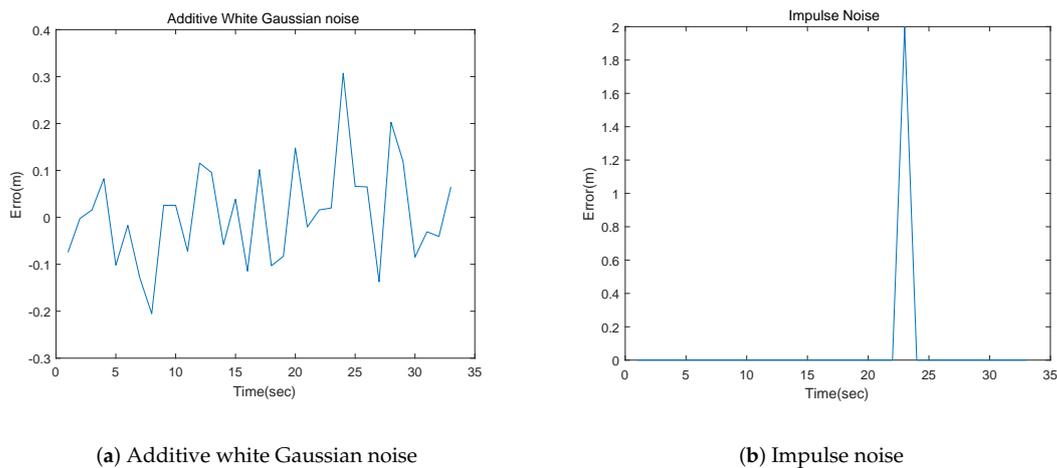


Figure 2. Additive white Gaussian noise and impulse noise.

To demonstrate the features of these two types of noise in our simulation, the localization errors of a robot under these two types of noise without filtering are shown in Figure 3a,b. As shown in these figures, the localization error of the robot increases smoothly under additive white Gaussian noise, whereas it increases suddenly under impulse noise.

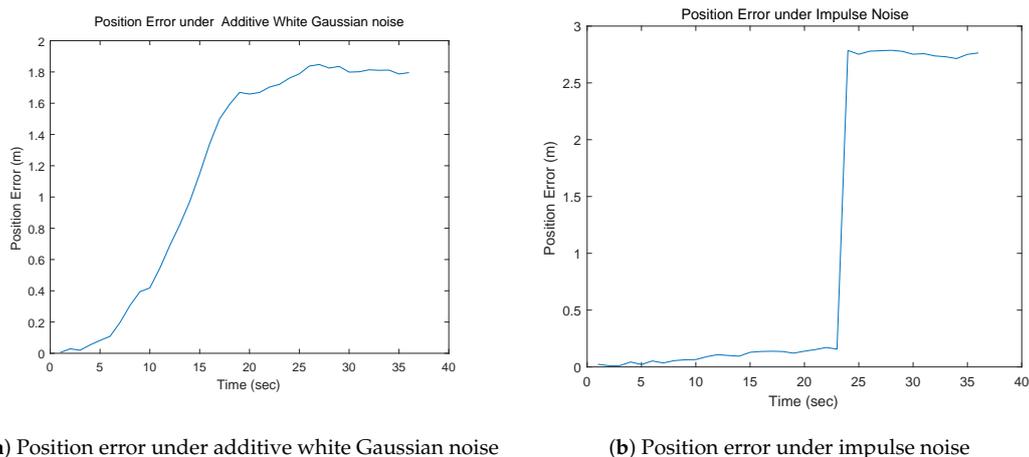


Figure 3. Position errors under additive white Gaussian noise and impulse noise.

Fourth, the members of the robot team are assumed to be subject to different levels of positioning error. When one robot changes its orientation to avoid an obstacle, the accuracy of its motion model will be lower than those of its neighboring robots because the amplitude of the noise is related to the yaw rate and the yaw rate of this robot will be larger than those of the other robots on the team.

2.1. Split Covariance Intersection Filter

As mentioned above, in DCL with the SCIF, each robot executes a localization algorithm to calculate its own position. This subsection provides a brief introduction to the concept of the SCIF, which enables the fusion of two different estimates with an unknown correlation.

Given two estimates $\{X_1, P_1\}$ and $\{X_2, P_2\}$, where X denotes the estimated state vector and P denotes the estimated covariance, the following fused estimate $\{X, P\}$ can be calculated using the Kalman filter:

$$\begin{aligned} P^{-1} &= P_1^{-1} + P_2^{-1}, \\ X &= P(P_1^{-1}X_1 + P_2^{-1}X_2). \end{aligned} \quad (1)$$

However, one premise of the Kalman filter that may not be satisfied is that the estimate and the observation are independent of each other. In other words, the estimate and the observation are correlated, Equation (1) might yield inconsistent results, and the fused result might be overconfident.

One widely used tool for handling the fusion of estimates with unknown correlations is the covariance intersection filter (CIF) [17]. In the CIF approach, a parameter w is added to the fusion formulae:

$$\begin{aligned} P^{-1} &= \left(\frac{P_1}{w}\right)^{-1} + \left(\frac{P_2}{1-w}\right)^{-1}, \\ X &= P\left(\left(\frac{P_1}{w}\right)^{-1}X_1 + \left(\frac{P_2}{1-w}\right)^{-1}X_2\right), \end{aligned} \quad (2)$$

where w , which takes values in the interval $[0, 1]$, is chosen to minimize P .

Although the CIF alleviates the overconfidence problem, it also has clear drawbacks, mainly because it treats the estimates to be fused as completely interdependent values. Thus, the fused result estimated with the CIF is pessimistic. To further mitigate this problem, the SCIF [12] has been introduced. In the algorithm of SCIF, the estimates to be fused are split into two parts: an independent part and a dependent part. Thus, the estimate $\{X_1, P_1\}$ and the observation $\{X_2, P_2\}$ to be fused are reformulated as $\{X_1, P_{1d} + P_{1i}\}$ and $\{X_2, P_{2d} + P_{2i}\}$, respectively, where P_{1d} corresponds to the dependent part of X_1 and P_{1i} corresponds to the independent part of X_1 . The formula of SCIF can be written as

$$\begin{aligned} P_1 &= \frac{P_{1d}}{w} + P_{1i}, \\ P_2 &= \frac{P_{2d}}{1-w} + P_{2i}, \\ P^{-1} &= P_1^{-1} + P_2^{-1}, \\ X &= P(P_1^{-1}X_1 + P_2^{-1}X_2), \\ P_i &= P(P_1^{-1}P_{1i}P_1^{-1} + P_2^{-1}P_{2i}P_2^{-1})P, \\ P_d &= P - P_i, \end{aligned} \quad (3)$$

where w , which takes values in the interval $[0, 1]$, is chosen to minimize P .

2.2. Statistical Decision Theory Based on an Information Theoretic Approach

In this subsection, statistical decision theory based on an information theoretic approach is introduced. In this algorithm, the decision is evaluated based on the mutual information $I(S, S_a)$. The decision rule $\delta(x)$, which is based on an observation x , can be written as shown in Equation (4).

$$\delta(x) = \arg \max_{S_a} I(S, S_a) \quad (4)$$

Here, the decision S_a is chosen to maximize $I(S, S_a)$. The input state x represents the observation of the fault state of the detection system, which is conditioned on the true state, represented by S .

The state space of S consists of two values: s_0 or s_1 , where s_0 represents the absence of fault and s_1 represents the presence of fault. Similarly the decision result S_a may only take one of these two values.

The value of $I(S, S_a)$ can be calculated using Equation (5):

$$I(S, S_a) = \sum_S \sum_{S_a} p(S, S_a) \log\left(\frac{p(S|S_a)}{p(S_a)}\right), \quad (5)$$

where the various probabilities that appear in this expression are defined as follows:

$$p(S|S_a):$$

$$p(S|S_a) = \begin{cases} p(S = s_0|S_a = s_0) = 1 - P_F, \\ p(S = s_0|S_a = s_1) = P_F, \\ p(S = s_1|S_a = s_0) = 1 - P_D, \\ p(S = s_1|S_a = s_1) = P_D. \end{cases} \quad (6)$$

$$p(S_a):$$

$$\begin{aligned} p(S_a = s_0) &= P_0(1 - P_F) + (1 - P_0)(1 - P_D), \\ p(S_a = s_1) &= P_0P_F + (1 - P_0)P_D. \end{aligned} \quad (7)$$

$$p(S, S_a):$$

$$p(S, S_a) \begin{cases} p(S = s_0, S_a = s_0) = (1 - P_F)P_0, \\ p(S = s_0, S_a = s_1) = P_FP_0, \\ p(S = s_1, S_a = s_0) = (1 - P_D)(1 - P_0), \\ p(S = s_1, S_a = s_1) = P_D(1 - P_0). \end{cases} \quad (8)$$

Here, P_D is the probability of correctly concluding that there are faulty sensors when faulty sensors are present in the robot system, P_F is the probability of incorrectly concluding that there are faulty sensors when no faulty sensors are present in the robot system, and P_0 is a priori probability of the absence of faulty sensors in the robot system, whose initial value is known.

The values of P_D and P_F are calculated as shown in Equation (9):

$$\begin{aligned} P_D &= \int_{\lambda}^{\infty} f(x|S = s_1)dx, \\ P_F &= \int_{\lambda}^{\infty} f(x|S = s_0)dx. \end{aligned} \quad (9)$$

Additionally, the probability density functions $f(x|S = s_0)$ and $f(x|S = s_1)$ are computed from the data distributions of x in the faulty and nonfaulty cases, respectively.

By substituting Equations (6)–(8) into Equation (5), we can obtain Equation (10):

$$\begin{aligned} I(S, S_a) &= P_0(1 - P_F) \log\left(\frac{1 - P_F}{P_0(1 - P_F) + (1 - P_0)(1 - P_D)}\right) \\ &+ P_0P_F \log\left(\frac{P_F}{P_0P_F + (1 - P_0)P_D}\right) \\ &+ (1 - P_0)(1 - P_D) \log\left(\frac{1 - P_D}{P_0(1 - P_F) + (1 - P_0)(1 - P_D)}\right) \\ &+ (1 - P_0)(P_D) \log\left(\frac{P_D}{P_0P_F + (1 - P_0)P_D}\right). \end{aligned} \quad (10)$$

The threshold t of the likelihood ration that maximizes $I(S, S_a)$ is determined as follows. First, the derivative of $I(S, S_a)$ with respect to P_F is calculated:

$$\begin{aligned} \partial I / \partial P_F &= -P_0 \log \frac{1 - P_F}{P_F} + [P_0 + t(1 - P_0)] \log(p(S_a = s_0)) \\ &- t(1 - P_0) \log \frac{1 - P_D}{P_D} \\ &- [P_0 + t(1 - P_0)] \log(p(S_a = s_1)). \end{aligned} \quad (11)$$

Set this derivative is equal to zero. We obtain that

$$t = \frac{-P_0[\log(p(S_a = s_0)/p(S_a = s_1)) - \log((1 - P_F)/P_F)]}{(1 - P_0)[\log(p(S_a = s_0)/p(S_a = s_1)) - \log((1 - P_D)/P_D)]}. \quad (12)$$

Here,

$$t = \frac{\partial P_D}{\partial P_F}. \quad (13)$$

With this threshold t , the decision rule $\delta()$ can be written as

$$\delta(x) = \begin{cases} s_1 & I(x) \geq t, \\ s_0 & I(x) < t. \end{cases} \quad (14)$$

Here, $I(x)$ is a likelihood ratio that is calculated as follows:

$$I(x) = \frac{f_x(x|s_1)}{f_x(x|s_0)}. \quad (15)$$

Likely wise Equation (14) can be expressed as

$$\delta(x) = \begin{cases} s_1 & x \geq \lambda, \\ s_0 & x < \lambda. \end{cases} \quad (16)$$

Here λ is calculated by algorithm where is chosen from the a set λ consisting of values between λ_{max} and λ_{min} (shown in Algorithm 1).

Algorithm 1 Algorithm of the thresholding calculation:

```

1:  $i \leftarrow 1$ 
2:  $I(S, S_a) \leftarrow 0$ 
3: while  $i < max$  do
4:   calculate  $P_{Fi}$  and  $P_{Di}$  with  $\lambda_i$  by Equation (9)
5:   calculate  $I(S, S_a)_i$  with  $P_{Fi}$  and  $P_{Di}$  by Equation (10)
6:   if  $I(S, S_a)_i > I(S, S_a)$  then
7:      $I(S, S_a) \leftarrow I(S, S_a)_i$ 
8:      $\lambda \leftarrow \lambda_i$ 
9:   end if
10:   $i \leftarrow i + 1$ 
11: end while

```

3. Materials and Methods

The DCL algorithm with FDI is introduced in this section. The communication graph of DCL in a robot team with three robot members, where each robot communicates only with its neighbors, is shown in Figure 4. As shown in this figure, there is no fusion center in DCL; instead, every robot executes its own localization and FDI algorithm to calculate its own position. In contrast with the fusion center in CCL, which can obtain all the information that it needs, in DCL, every robot can obtain information only from its neighbors. For this reason, the information available to every robot in DCL is limited, and the FDI algorithms designed for CCL are not applicable to DCL. To enable the application of FDI in DCL, a novel FDI algorithm is introduced in this section.

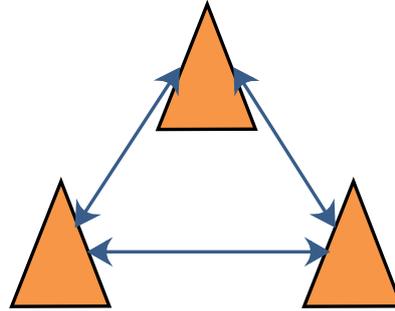


Figure 4. Communication graph for DCL.

3.1. Decentralized Cooperative Localization Algorithm

In this section, the method for achieving DCL using the SCIF (abbreviated henceforth as SCIFCL) is introduced. The SCIFCL algorithm is presented as Algorithm 2.

Algorithm 2 DCL Algorithm with the SCIF (SCIFCL):

- 1: **for** $i = 1$ to N **do**
 - 2: $X_{k/k-1}^i = f(X_{k-1}^i, u_{k-1}^i)$
 - 3: $P_{k/k-1}^i = F_{k-1}^i P_{k-1}^i F_{k-1}^{iT} + G_{k-1}^i (Qu)_{k-1}^i G_{k-1}^{iT}$
 - 4: $P_{k/k-1}^{i,in} = F_{k-1}^i P_{k-1}^{i,in} F_{k-1}^{iT} + G_{k-1}^i (Qu)_{k-1}^i G_{k-1}^{iT}$
 - 5: $P_{k/k-1}^{i,d} = P_{k/k-1}^i - P_{k/k-1}^{i,in}$
 - 6: **if** there exists a relative measurement Z_k^{ji} between robot j and robot i **then**
 - 7: $X_{k/k-1}^{i*} = h(X_{k/k-1}^j, Z_k^{ji})$
 - 8: $P_{k/k}^i = (1 - K)(P_{k/k-1}^{i,d} / \tau w + P_{k/k-1}^{i,in})$
 - 9: $X_{k/k}^i = X_{k/k-1}^i + K(X_{k/k-1}^{i*} - X_{k/k-1}^i)$
 - 10: **end if**
 - 11: **end for**
-

The state model of robot i in a multiple mobile robots team can be expressed as

$$\begin{aligned}
 X_{k/k-1}^i &= f(X_{k-1}^i, u_{k-1}^i) \\
 &= \begin{pmatrix} x_{k-1}^i \\ y_{k-1}^i \\ \theta_{k-1}^i \end{pmatrix} + \begin{pmatrix} \Delta_{k-1}^i \cos(\theta_{k-1}^i + \frac{v_{k-1}^i}{2}) \\ \Delta_{k-1}^i \sin(\theta_{k-1}^i + \frac{v_{k-1}^i}{2}) \\ v_{k-1}^i \end{pmatrix}.
 \end{aligned} \tag{17}$$

where X_k^i is the state vector of robot i :

$$X_k^i = [x_k^i, y_k^i, \theta_k^i]^T,$$

and

$$X_{k-1} = [X_{k-1}^1, X_{k-1}^2, \dots, X_{k-1}^i, \dots]^T.$$

Here, Δ_{k-1}^i and v_{k-1}^i are the elementary displacement and rotation, respectively, of robot i .

In summary, the state model of the multirobot system consists of differentially driven robots governed by a linear control law.

And the Jacobian matrices $F_{k-1}^i = \frac{\partial f}{\partial X}$ and $G_{k-1}^i = \frac{\partial f}{\partial u}$ are written as

$$F_{k-1}^i = \begin{pmatrix} 1 & 0 & -\Delta_{k-1}^i \sin(\theta_{k-1}^i + \frac{v_{k-1}^i}{2}) \\ 0 & 1 & \Delta_{k-1}^i \cos(\theta_{k-1}^i + \frac{v_{k-1}^i}{2}) \\ 0 & 0 & 1 \end{pmatrix} \quad (18)$$

and

$$G_{k-1}^i = \begin{pmatrix} \cos(\theta_{k-1}^i + \frac{v_{k-1}^i}{2}) & -\frac{1}{2} \Delta_{k-1}^i \sin(\theta_{k-1}^i + \frac{v_{k-1}^i}{2}) \\ \sin(\theta_{k-1}^i + \frac{v_{k-1}^i}{2}) & \frac{1}{2} \Delta_{k-1}^i \cos(\theta_{k-1}^i + \frac{v_{k-1}^i}{2}) \\ 0 & 1 \end{pmatrix}. \quad (19)$$

In DCL, in addition to the covariance of the robot itself, $P_{k/k-1}^i$, two additional covariances, $P_{k/k-1}^{i,in}$ and $P_{k/k-1}^{i,d}$, are also calculated as follows:

$$\begin{aligned} P_{k/k-1}^i &= F_{k-1}^i P_{k-1}^i F_{k-1}^{iT} + G_{k-1}^i (Q_u)_{k-1}^i G_{k-1}^{iT}, \\ P_{k/k-1}^{i,in} &= F_{k-1}^i P_{k-1}^{i,in} F_{k-1}^{iT} + G_{k-1}^i (Q_u)_{k-1}^i G_{k-1}^{iT}, \\ P_{k/k-1}^{i,d} &= P_{k/k-1}^i - P_{k/k-1}^{i,in}. \end{aligned} \quad (20)$$

When robot i is observed by robot j , the measurement model can be written as

Thus, the measurement of X_k^i can be written as

$$\begin{aligned} X_{k/k-1}^{i*} &= h(X_{k/k-1}^j, Z_k^j) \\ &= \begin{pmatrix} x_{k/k-1}^j + x_k^j \cos \theta_{k/k-1}^j - y_k^j \sin \theta_{k/k-1}^j \\ y_{k/k-1}^j + x_k^j \sin \theta_{k/k-1}^j + y_k^j \cos \theta_{k/k-1}^j \\ \theta_{k/k-1}^j + \theta_k^j \end{pmatrix}, \end{aligned} \quad (21)$$

where

$$Z_k^j = [x_k^j, y_k^j, \theta_k^j]^T.$$

Then, the measurement and the observation to be fused are $\{X_{k/k-1}^i, P_{k/k-1}^{i,d} + P_{k/k-1}^{i,in}\}$ and $\{X_{k/k-1}^{i*}, P_{k/k-1}^{i*,d} + P_{k/k-1}^{i*,in}\}$, respectively, where

$$\begin{aligned} P_{k/k-1}^{i*} &= C_k^{i*} P_{k/k-1}^j (C_k^{i*})^T + D_k^{i*} (D_k^{i*})^T, \\ P_{k/k-1}^{i*,in} &= C_k^{i*} P_{k/k-1}^{j,in} (C_k^{i*})^T + D_k^{i*} (D_k^{i*})^T. \end{aligned} \quad (22)$$

Here, the Jacobian matrices $C_k^{i*} = \frac{\partial X_{k/k-1}^{i*}}{\partial X_{k/k-1}^j}$ and $D_k^{i*} = \frac{\partial X_{k/k-1}^{i*}}{\partial Z_k^j}$ can be expressed as

$$\begin{aligned} C_k^{i*} &= \frac{\partial X_{k/k-1}^{i*}}{\partial X_{k/k-1}^j} \\ &= \begin{pmatrix} 1 & 0 & -x_k^j \sin(\theta_{k/k-1}^j) - y_k^j \cos(\theta_{k/k-1}^j) \\ 0 & 1 & x_k^j \cos(\theta_{k/k-1}^j) - y_k^j \sin(\theta_{k/k-1}^j) \\ 0 & 0 & 1 \end{pmatrix}, \\ D_k^{i*} &= \frac{\partial X_{k/k-1}^{i*}}{\partial Z_k^j}. \end{aligned} \quad (23)$$

Applying the SCIF to the cooperative localization problem, we obtain that

$$\begin{aligned} (P_{k/k}^i)^{-1} &= \left(\frac{P_{k/k-1}^{i,d}}{w} + P_{k/k-1}^{i,in}\right)^{-1} + \left(\frac{P_{k/k-1}^{i*,d}}{1-w} + P_{k/k-1}^{i*,in}\right)^{-1} \\ X_{k/k}^i &= P_{k/k}^i \left[\left(\frac{P_{k/k-1}^{i,d}}{w} + P_{k/k-1}^{i,in}\right)^{-1} X_{k/k-1}^i + \left(\frac{P_{k/k-1}^{i*,d}}{1-w} + P_{k/k-1}^{i*,in}\right)^{-1} X_{k/k-1}^{i*} \right]. \end{aligned} \quad (24)$$

Writing the above equation in Kalman filter form, we obtain that

$$P_{k/k}^i = (1-K) \left(\frac{P_{k/k-1}^{i,d}}{w} + P_{k/k-1}^{i,in}\right) \quad (25)$$

$$X_{k/k}^i = X_{k/k-1}^i + K(X_{k/k-1}^{i*} - X_{k/k-1}^i), \quad (26)$$

where $K = \frac{\left(\frac{P_{k/k-1}^{i,d}}{w} + P_{k/k-1}^{i,in}\right)}{\left(\frac{P_{k/k-1}^{i,d}}{w} + P_{k/k-1}^{i,in}\right) + \left(\frac{P_{k/k-1}^{i*,d}}{1-w} + P_{k/k-1}^{i*,in}\right)}$.

And

$$\begin{aligned} P_{k/k}^{i,in} &= (1-K)P_{k/k-1}^{i,in}(1-K)^T + KP_{k/k-1}^{i*,in}K^T \\ P_{k/k}^{i,d} &= P_{k/k}^i - P_{k/k}^{i,in}. \end{aligned} \quad (27)$$

Thus, the fused result is $\{X_{k/k}^i, P_{k/k}^{i,d} + P_{k/k}^{i,in}\}$.

3.2. Fault Detection and Isolation in Decentralized Cooperative Localization

After estimating the locations of the robot team via SCIFCL, the FDI algorithm is implemented. As mentioned above, in DCL, each member of the team communicates only with its neighbors, and thus, the information available to each robot for FDI is limited. To resolve this problem, an improved FDI algorithm is developed.

To evaluate the possibility of the presence of faulty sensors, the Kullback–Leibler divergence (KLD) between the results of estimation before and after the measurement update is calculated. The KLD is a measure of the extent of one probability distribution diverges from another. If $f(x)$ and $g(x)$ are two-dimensional Gaussian distributions, with respective means of μ_1 and μ_2 and covariance matrices P_1 and P_2 , then the KLD can be calculated using Equation (28):

$$KLD(f(x)||g(x)) = \frac{1}{2}[\text{trace}(P_2^{-1}P_1) + \log(\det(\frac{P_2}{P_1})) - d + (\mu_1 - \mu_2)^T P_2^{-1}(\mu_1 - \mu_2)]. \quad (28)$$

Here, the mean $X_{k/k-1}$ and the covariance matrix $P_{k/k-1}$ of the state estimate before the measurement update correspond to the prior probability distribution, while the mean $X_{k/k}$ and the covariance matrix $P_{k/k}$ of the state estimate after the measurement update correspond to the posterior probability distribution.

Since both the prior and posterior probability distributions are two-dimensional Gaussian distributions, each robot can use the information obtained from its neighbors to calculate $GKLD^i$, as an indicator of the possible presence of a fault of robot i , as shown in Equation (29).

$$\begin{aligned} GKLD^i &= KLD(X_{k/k-1}^i || X_{k/k}^i) \\ &= \frac{1}{2}[\text{trace}(P_{k/k}^{i,-1} P_{k/k-1}^i) + \log(\det(\frac{P_{k/k}^i}{P_{k/k-1}^i})) - M \\ &\quad + (X_{k/k}^i - X_{k/k-1}^i)^T P_{k/k}^{i,-1} (X_{k/k}^i - X_{k/k-1}^i)] \end{aligned} \quad (29)$$

With the threshold optimization algorithm mentioned in Section 2, the problem of fault detection in DCL is simplified to find the S_a which maximizes the value of $\delta(GKLD^i)$ in Equation (30).

$$\delta(GKLD^i) = \arg \max_{S_a} I(S, S_a) \quad (30)$$

Here the state space of S consists of two values: s_0 or s_1 , where s_0 represents the absence of faulty sensors and s_1 represents the presence of faulty sensors.

And the threshold λ can be expressed as the function of $GKLD^i$.

$$\delta(GKLD^i) = \begin{cases} s_1 & GKLD^i \geq \lambda \\ s_0 & GKLD^i < \lambda \end{cases} \quad (31)$$

Here the threshold λ is obtained by Algorithm 1.

After fault detection, if a fault in a robot is detected, then the fault isolation procedure is applied to identify the faulty robot. KL residuals are calculated to isolate the fault. The number of KL residuals calculated depends on the number of inter-robot observations. For example, when robot j is observed by robot i , the KL residual KL^{ij} is calculated as shown in Equation (32):

$$\begin{aligned} KL^{ij} &= KLD(X_{k/k-1}^i || X_{k/k}^{ij}) \\ &= \frac{1}{2} [\text{trace}(P_{k/k}^{ij}{}^{-1} P_{k/k-1}^i) + \log(\det(\frac{P_{k/k}^{ij}}{P_{k/k-1}^i})) - M + (X_{k/k}^{ij} - X_{k/k-1}^i)^T P_{k/k}^{ij}{}^{-1} (X_{k/k}^{ij} - X_{k/k-1}^i)]. \end{aligned} \quad (32)$$

Here, $X_{k/k}^{ij}$ denotes the estimation calculated by robot i based on its observation of robot j .

Each KL residual is sensitive to some faults and insensitive to others. For example, the value of KL^{ij} depends on the odometers of robots i and j and the observation of robot j as measured by robot i .

$$\begin{aligned} \delta(KL^{ij}) &= \arg \max_{S_a} I(S, S_a) \\ &= s_1 \end{aligned} \quad (33)$$

If the value of KL^{ij} satisfies Equation (33), then there may be a fault in the odometer of robot j or i or in robot i 's observation of robot j . The corresponding signal value is equal to 1. As an illustrative example, the signature matrix of robot 1 is given in Table 1.

Since the authors of [16] utilized lidar and Kinect to capture the relative position of other robot, the observation here may represent the sensor information from these two sensors.

Table 1. The signature matrix of robot 1 for DCL.

| | KL^{12} | KL^{13} |
|---------------|-----------|-----------|
| Odometer 1 | 1 | 1 |
| Odometer 2 | 1 | 0 |
| Odometer 3 | 0 | 1 |
| Observation 1 | 0 | 0 |
| Observation 2 | 1 | 0 |
| Observation 3 | 0 | 1 |

However, it is difficult to isolate the faulty sensors with signature matrices described above. To enable FDI in DCL, the signature matrices are simplified by combining the information on the observation and the odometer sensor for each robot. Here, the grouped observation information for each robot is sensitive to faults in both the observation and the odometer sensor of that robot. The corresponding signal value will equal to 1 when either the observation or the odometer sensor is

faulty. Thus, the signature matrices are modified as shown in Tables 2–4 for the case of a robot team with three members.

Table 2. The simplified signature matrix of robot 1.

| | KL^{12} | KL^{13} |
|---------|-----------|-----------|
| Robot 1 | 1 | 1 |
| Robot 2 | 1 | 0 |
| Robot 3 | 0 | 1 |

Table 3. The simplified signature matrix of robot 2.

| | KL^{12} | KL^{23} |
|---------|-----------|-----------|
| Robot 1 | 1 | 0 |
| Robot 2 | 1 | 1 |
| Robot 3 | 0 | 1 |

Table 4. The simplified signature matrix of robot 3.

| | KL^{13} | KL^{23} |
|---------|-----------|-----------|
| Robot 1 | 1 | 0 |
| Robot 2 | 0 | 1 |
| Robot 3 | 1 | 1 |

However, after this simplification, there is one fault type that cannot be isolated. If faults exist in both sensors, then the signature matrices will both be equal to (1, 1), which is same with the case that faults exist in both observations. Thus, no observation information will be accepted, even if these sensors are free of faults.

To solve this problem, we add another set of KLD entries to the signature matrix based on the measured divergence between the observations of the two robots being observed. Thus, the signature matrices are modified to those shown in Tables 5–7. The proposed signature matrices can be used to identify more fault types than is possible with the classic isolation algorithm.

$$\begin{aligned}
 KL^{ik/ij} &= KLD(X_k^{i/k} || X_k^{i/j}) \\
 &= \frac{1}{2} [\text{trace}(P_k^{i/j-1} P_k^{i/k}) + \log(\det(\frac{P_k^{i/j}}{P_k^{i/k}})) - M + (X_k^{i/j} - X_k^{i/k})^T P_k^{i/j-1} (X_k^{i/j} - X_k^{i/k})] \quad (34)
 \end{aligned}$$

Here, $X_k^{i/k}$ denotes the observation of robot k as measured by robot i .

$$\begin{aligned}
 \delta(KL^{ik/ij}) &= \arg \max_{S_a} I(S, S_a) \\
 &= s_1 \quad (35)
 \end{aligned}$$

If the value of $KL^{ik/ij}$ satisfies Equation (35), the corresponding signal value is equal to 1.

Table 5. The proposed signature matrix of robot 1.

| | KL^{12} | KL^{13} | $KL^{12/13}$ |
|---------|-----------|-----------|--------------|
| Robot 1 | 1 | 1 | 0 |
| Robot 2 | 1 | 0 | 1 |
| Robot 3 | 0 | 1 | 1 |

Table 6. The proposed signature matrix of robot 2.

| | KL^{12} | KL^{23} | $KL^{12/23}$ |
|---------|-----------|-----------|--------------|
| Robot 1 | 1 | 0 | 1 |
| Robot 2 | 1 | 1 | 0 |
| Robot 3 | 0 | 1 | 1 |

Table 7. The proposed signature matrix of robot 3.

| | KL^{13} | KL^{23} | $KL^{13/23}$ |
|---------|-----------|-----------|--------------|
| Robot 1 | 1 | 0 | 1 |
| Robot 2 | 0 | 1 | 1 |
| Robot 3 | 1 | 1 | 0 |

4. Results

To validate the FDI algorithm proposed in this paper, four simulations are performed. The simulations reported in this paper were conducted in MATLAB. The background for the simulations was based on a real scenario in which a team of robots were moved forward in a certain formation while maneuvering to avoid an obstacle ahead. To maintain the formation of the team, one robot adjusted its orientation, as shown in Figure 5.

The localization results obtained for the robot team using the SCIFCL algorithm and the SCIFCL algorithm with FDI were tested in four simulations. In the first two simulations, only additive white Gaussian noise was considered, and the distance between neighboring robots was varied from 2 m to 1 m. The velocity of each robot was approximately 0.3 m/s. The additive white Gaussian noise added to the motion measurements introduced standard errors of 0.1 m/s in velocity and 0.005 rad/s in yaw rate. The absolute positioning measurement period was set to 1 s. The standard errors in relative positioning were set to 0.1 m in relative position and 0.005 rad in relative orientation. Time delay was not considered in these simulations.

The simulations consisted of two primary stages. In the first stage, one robot was randomly chosen as the leader, and each of the other robots used its relative measurements of the leader robot until its own state estimation converged. Then, in the second stage, the SCIFCL algorithm and the SCIFCL algorithm with FDI were executed simultaneously, and the robot localization errors associated with these two algorithms were recorded.

In simulation 1, the number of robots in the robot team was set to 3. The communication graph for the robot team is shown in Figure 6. The results of simulation 1 are shown in Figure 7, where each subfigure presents the localization errors of one team member with different localization methods. Here, the vertical axis represents the position error, and the horizontal axis represents the time step. The localization errors of the different methods are represented by lines of different colors. The blue line denotes the robot localization error achieved with the SCIFCL algorithm, and the red line denotes the robot localization error achieved with the SCIFCL algorithm with FDI.

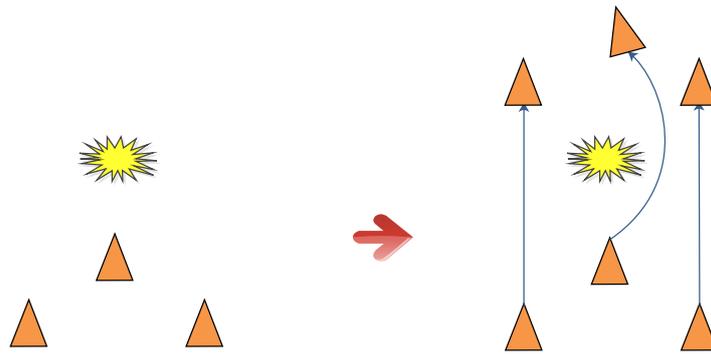


Figure 5. Simulation scenario: a team of three robots.

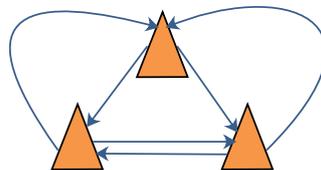


Figure 6. Communication graph for 3 robots.

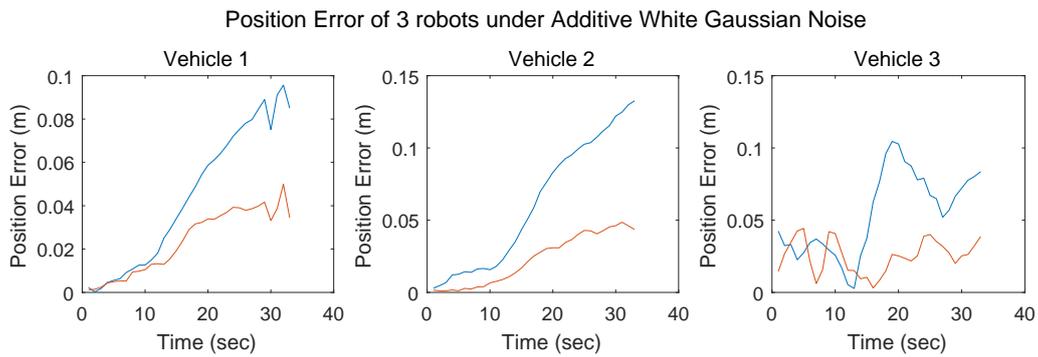


Figure 7. Position errors of 3 robots under additive white Gaussian noise.

In simulation 2, the number of robots in the robot team was set to be 9. Since at least two neighbor observations are needed for each robot to apply the SCIFCL algorithm with FDI, the communication graph for the robot team was defined as shown in Figure 8, with each robot using only the information obtained by observing the two nearest robots.

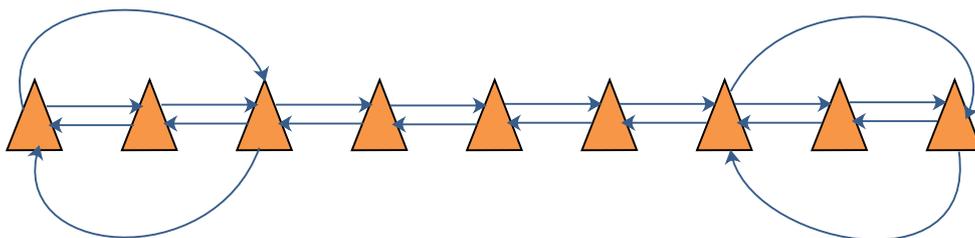


Figure 8. Communication graph for 9 robots.

The results of simulation 2 are shown in Figure 9, where each subfigure presents the localization errors for one team member as obtained using different localization methods.

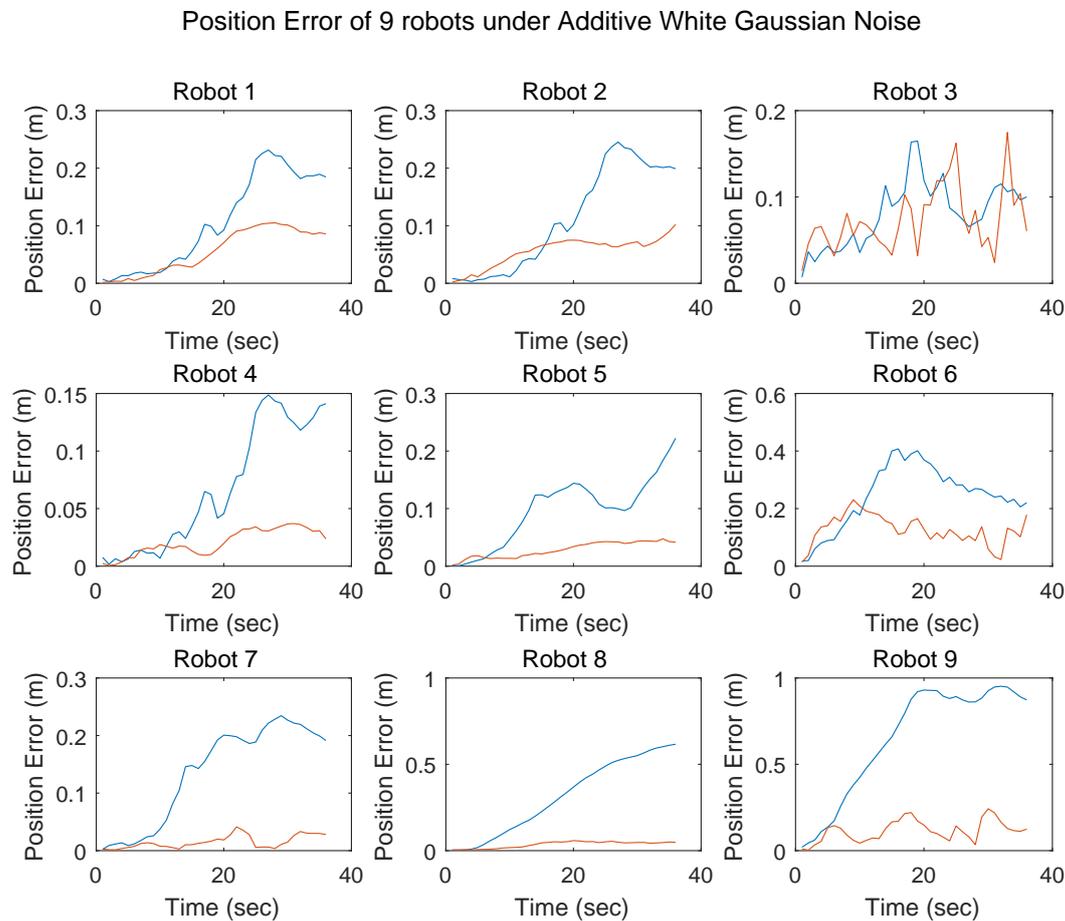


Figure 9. Position errors of 9 robots under additive white Gaussian noise.

The SCIFCL results will be improved if some team members achieve better positioning accuracy. However, the localization results achieved by robots with better accuracy will be “contaminated” by the robots with less position accuracy. This is mainly because the SCIFCL procedure does not consider the accuracy of the observations being fused. We divide the team members into two groups based on their localization accuracy, where the accuracy of the first group of robots is better than that of the second group. Updating the localization result for a robot in either the first or the second group with the observation from the first group of robots will improve the accuracy of its localization result. However, updating the localization result for a robot in either the first or the second group with an observation from the second group of robots will decrease the accuracy of the result being updated.

Since the SCIFCL algorithm is applied to all observations from neighboring robots, some observations from robots in the second group will be utilized to adjust the states of robots in the first group. Consequently, the localization results for robots in the first group will be “contaminated” by robots in the second group. However, with the FDI algorithm, it is possible to select the robots that belong to the second group and isolate them. If the observations from robots in the second group are excluded from the measurement update, then the accuracy achieved for the entire robot team will be enhanced.

In simulations 3 and 4, impulse noise was added in addition to additive white Gaussian noise. In simulation 3, the number of robots in the team was three, while in simulation 4, the number of team members was 9. The results of these two simulations are shown in Figures 10 and 11, respectively.

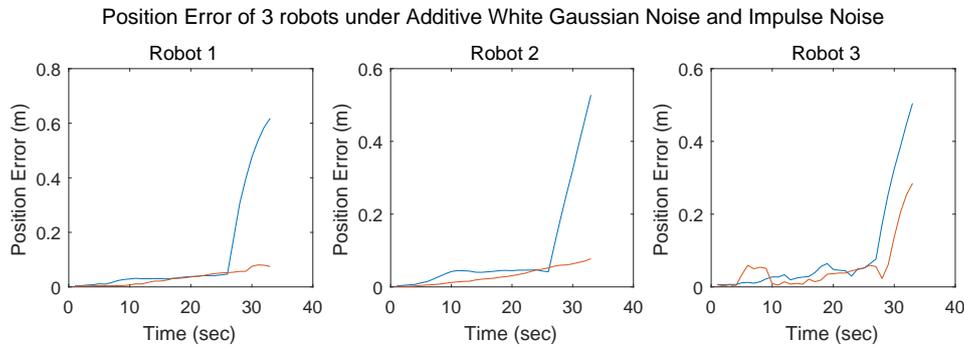


Figure 10. Position errors of 3 robots under additive white Gaussian noise and impulse noise.

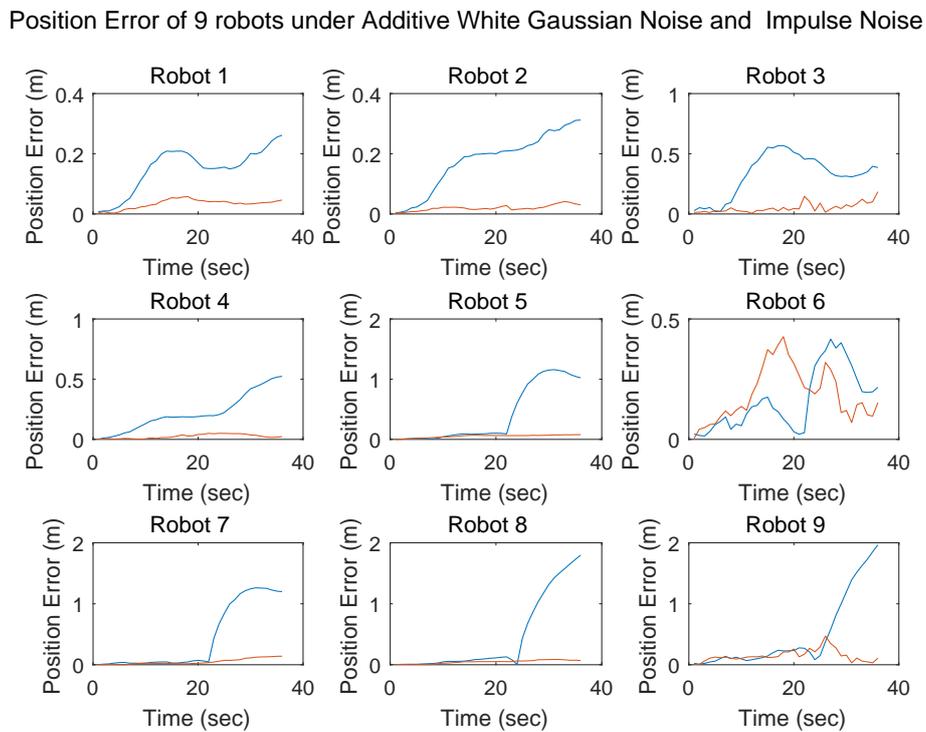


Figure 11. Position errors of 9 robots under additive white Gaussian noise and impulse noise.

As in the case of simulation 1, the localization errors achieved using different methods are represented by lines of different colors. The blue lines denote the localization errors of the SCIFCL method, and the red lines denote the localization errors of the SCIFCL method with FDI.

As shown in Figures 10 and 11, the phenomenon of “contamination” is more evident when impulse noise is present. The SCIFCL results for the robot team diverge, while for the SCIFCL algorithm with FDI, the localization results for other robots in the robot team are not affected by robots that experience extreme localization failure.

The results of these four simulations demonstrate that the SCIFCL algorithm with FDI exhibits better localization performance than that of the SCIFCL algorithm alone.

5. Discussion

In this paper, an FDI algorithm is applied in DCL for a multirobot system. In the proposed algorithm, the signature matrix used in the original FDI algorithm is modified to enable the application of FDI in DCL, in which each robot communicates only with its neighbors. In the proposed algorithm, certain fault types in the signature matrix are combined, and a new KL residual is added. With this modified signature matrix, fault isolation can be achieved in DCL. Simulations have been conducted to test the proposed FDI algorithm for DCL. The experimental results demonstrate that the proposed DCL algorithm with FDI achieves better accuracy and robustness than the DCL algorithm without FDI does.

The purpose of the algorithm introduced in this paper is to serve as an augmentation approach to compensate for poor odometric measurements and noisy and distorted measurements from other sensors. The proposed localization method may become a valuable tool for the application of FDI in DCL.

Possible further studies are concluded below:

1. We plan to test the performance of the proposed method in a real experiment.
2. Our algorithm is designed to be an augmentation system to some localization algorithm. A localization algorithm with map for example AMCL and SCIFCL with FDI would be part of our further work.
3. In our plan, the communication graph may change but the signature matrices will not change for the algorithm FDI runs in each robot only need two nearest neighbor information. The further discussion is needed about this issue.
4. When every robot loses their position at the same time a re-location algorithm with sensors like GPS or anchor node will be needed.

Author Contributions: H.M. conceived the idea, provided financial support to this work and finalized this paper. M.W. designed the simulations, implemented the algorithm, analyzed the data and drafted the paper. X.Z. provided constructive and useful comments to the manuscript. All authors read and approved the manuscript.

Funding: This work is partially supported by the National Key Research and Development Program of China [2017YFF0205300], the National Natural Science Foundation of China [61473038, 91648117] and Beijing Natural Science Foundation [4172055].

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Chi, W.; Wang, J.; Meng, M.Q. A Gait Recognition Method for Human Following in Service Robots. *IEEE Trans. Syst. Man Cybern. Syst.* **2018**, *48*, 1429–1440. [[CrossRef](#)]
2. Ghedini, C.; Ribeiro, C.H.; Sabattini, L. Toward efficient adaptive ad-hoc multi-robot network topologies. *Ad Hoc Netw.* **2018**, *74*, 57–70. [[CrossRef](#)]
3. Cooper, S.B.; Durrantwhyte, H.F. A Kalman filter model for GPS navigation of land vehicles. In Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems, Munich, Germany, 12–16 September 1994; pp. 157–163. [[CrossRef](#)]
4. Fuentespacheco, J.; Ruizascencio, J.; Rendonmancha, J.M. Visual simultaneous localization and mapping: A survey. *Artif. Intell. Rev.* **2015**, *43*, 55–81. [[CrossRef](#)]
5. Fox, D. KLD-Sampling: Adaptive Particle Filters. *Adv. Neural Inf. Process. Syst.* **2001**, *14*, 713–720.
6. Grisetti, G.; Stachniss, C.; Burgard, W. Improved Techniques for Grid Mapping With Rao-Blackwellized Particle Filters. *IEEE Trans. Rob.* **2007**, *23*, 34–46. [[CrossRef](#)]
7. Kia, S.S.; Rounds, S.F.; Martinez, S. Cooperative Localization for Mobile Agents: A Recursive Decentralized Algorithm Based on Kalman-Filter Decoupling. *IEEE Control Syst.* **2015**, *36*, 86–101. [[CrossRef](#)]
8. Kurazume, R.; Nagata, S.; Hirose, S. Cooperative Positioning with Multiple Robots. *Proc. IEEE Int. Conf. Rob. Autom.* **1994**, *2*, 1250–1257. [[CrossRef](#)]
9. Rekleitis, I.; Dudek, G.; Miliotis, E. Multi-robot collaboration for robust exploration. *Ann. Math. Artif. Intell.* **2001**, *31*, 7–40. [[CrossRef](#)]

10. Howard, A.; Matarić, M.J.; Sukhatme, G.S. Mobile Sensor Network Deployment using Potential Fields: A Distributed, Scalable Solution to the Area Coverage Problem. In *Distributed Autonomous Robotic Systems 5, Proceedings of the International Symposium on Distributed Autonomous Robotic Systems, Fukuoka, Japan, 25–27 June 2002*; Springer: Tokyo, Japan, 2002; pp. 299–308. [[CrossRef](#)]
11. Kia, S.S.; Rounds, S.F.; Martinez, S. A centralized-equivalent decentralized implementation of Extended Kalman Filters for cooperative localization. In *Proceedings of the International Conference on Intelligent Robots and Systems, Chicago, IL, USA, 14–18 September 2014*; pp. 3761–3766. [[CrossRef](#)]
12. Li, H.; Nashashibi, F. Cooperative multi-vehicle localization using split covariance intersection filter. *IEEE Intell. Transp. Syst. Mag.* **2013**. [[CrossRef](#)]
13. Tinos, R.; Navarroserment, L.E.; Paredis, C.J.J. Fault tolerant localization for teams of distributed robots. In *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems, Maui, HI, USA, 29 October–3 November 2001*; pp. 1061–1066.
14. Hozo, I.; Djulbegovic, B.; Luan, S.; Tsalatsanis, A.; Gigerenzer, G. Towards theory integration: Threshold model as a link between signal detection theory, fast-and-frugal trees and evidence accumulation theory. *J. Eval. Clin. Pract.* **2017**, *23*, 49–65. [[CrossRef](#)] [[PubMed](#)]
15. Harmouche, J.; Delpha, C.; Diallo, D. Incipient fault detection and diagnosis based on Kullback–Leibler divergence using Principal Component Analysis: Part I. *Signal Process.* **2015**, *109*, 334–344. [[CrossRef](#)]
16. Hage, J.A.; Najjar, M.E.B.E.; Pomorski, D. Multi-sensor fusion approach with fault detection and exclusion based on the Kullback–Leibler Divergence: Application on collaborative multi-robot system. *Inf. Fusion* **2017**, *37*, 61–76. [[CrossRef](#)]
17. Julier, S.J.; Uhlmann, J.K. A non-divergent estimation algorithm in the presence of unknown correlations. In *Proceedings of the American Control Conference, Albuquerque, NM, USA, 6 June 1997*; Volume 4, pp. 2369–2373. [[CrossRef](#)]



© 2018 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<http://creativecommons.org/licenses/by/4.0/>).