

Article

An Improved Game Theory-Based Cooperative Localization Algorithm for Eliminating the Conflicting Information of Multi-Sensors

Chao Tang ^{1,2} and Lihua Dou ^{1,2,*}

¹ School of Automation, Beijing Institute of Technology, Beijing 100081, China; 3120170434@bit.edu.cn

² Beijing Institute of Technology Chongqing Innovation Center, Chongqing 401135, China

* Correspondence: doulihua@bit.edu.cn

Received: 12 August 2020; Accepted: 24 September 2020; Published: 29 September 2020



Abstract: In this article, an improved game theory-based co-localization algorithm is proposed to precisely and cooperatively locate the multi-robot system in the wireless sensor network and efficiently eliminate the information conflict caused by multi-sensor. Specifically, the extended Kalman filter in the original algorithm is replaced by the unscented Kalman filter in the optimized algorithm, which contributes to lower linearization errors and higher localization precision. Then, the computational complexity is analyzed, and the derivative method is introduced to reduce the extra computation burden brought by the unscented Kalman filter. Subsequently, the stability issue resulting from the derivative method is addressed by introducing the singular value decomposition (SVD). In this context, the optimized algorithm is capable of precisely locating the multi-robot system, while maintaining the stability and not increasing the computational burden. Moreover, as demonstrated by the simulation results, the optimized algorithm has greater localization precision than the original algorithm, while they have similar computational burdens.

Keywords: Kalman filter; information conflict; SVD; game theory; co-localization

1. Introduction

The multi-robot system has been proven to have better stability, stronger robustness, and lower redundancy than the single-robot system. Consequently, it has been widely applied in large-scale searches, post-disaster searches, and alert patrols, where all robots are expected to be fully aware of their locations in order to efficiently perform cooperative tasks. In this context, precise localization is critical and requires comprehensive investigation.

There are generally three steps in the traditional autonomous localization methods when it comes to sensor information processing. The first step is to collect the information obtained by both internal and external sensors [1,2]; the second step is to process the collected information and ensure that it conforms to the filtering algorithm's format; the last step is to calculate the localization result based on the information processed by the filtering algorithm [3]. In the case of a multi-robot system, the localization methods are usually extended from those designed for the single-robot system. For example, Roumeliotis et al. [4] distributed the interaction term to the relevant robots in the extended Kalman filtering (EKF) algorithm for multi-robot co-localization. Zhang et al. [5] proposed a cooperative localization approach with communication delays, using the EKF technique based on state estimation, and they suggested reducing the state estimation error of delay filtering based on the error compensation. Wang et al. [6] implemented the co-localization for wireless sensor networks (WSNs) by using a hard decision-based method with outlier constraints in the cooperative localization phase, which helped successfully solve the issue of range-free estimation and effectively avoided large errors

in node localization. In order to provide positioning services in larger area, P. N. Pathirana et al. [7] adopted the method of multiple robots touring around the WSNs and used the robust EKF to eliminate measurement noise. However, this method has a large delay and poor performance in real-time. A. Baggio et al. [8] improved the Monte Carlo localization algorithm in WSNs. They optimized the performance of unknown nodes calling position information. H. Chen et al. [9] studied the localization issues in WSNs with physical obstacles. They adopted the convex optimization method to reduce the moving distance of mobile elements in the networks, which extended the life of the mobile elements, while ensuring high-precision localization. Guo et al. [10] suggested supporting the co-localization of unmanned aerial vehicles (UAVs) in a non-GPS environment by adopting an Ultra-Wide Band (UWB)-based EKF localization algorithm. Pirník et al. [11] investigated the performances of inertial devices in controlling wheeled mobile platforms and developed an algorithm to integrate inertial sensor-recorded data into the system control. In this context, they experimentally proved the necessity of the inertial devices to interact with other absolute sensors. Liu et al. [12] combined particle filtering (PF) with the particle swarm optimization algorithm to deal with the co-localization of heterogeneous robot systems.

However, all the dynamic models in the abovementioned studies are of simple trajectories of motion, which means that they are difficult to apply in actual situations. Accordingly, efforts have been made to fill the gap by some European scholars [13]. For instance, Božek et al. [14] used the Newton–Euler method to formulate a dynamic model of a six-axis hexacopter equipped with a robotic arm, and a dynamic model for controlling a four-wheeled highly-dynamic robot was proposed by Kilin et al. [15]. In addition, C. Yu et al. did some studies on identification of state-space in the case of unknown dynamic model [16,17] or unknown input [18]. These studies made great contributions to filtering for complex dynamic models.

Although the above-mentioned methods manage to effectively solve the co-localization problem of the multi-robot system, they are frequently vulnerable to conflicting relative observation information that would lead to divergence in the Kalman filtering (KF) and increase the computational burden. When extending the localization algorithms from single-robot systems to multi-robot counterparts, it is of critical importance to fuse the relative observation information between multiple robots. Otherwise, the possibly present conflicts among different robots in terms of the relative observation information could lead to inconsistent co-localization and even filter divergence. Taking this into consideration, Hua et al. [19] integrated the complete information static game theory into the EKF algorithm for both one-way and two-way observations in the multi-robot systems, which suppressed the observation conflicts significantly.

However, the EKF algorithm adopted by Hua et al. [19] is of low precision, and its improvement is the main focus of this study. Meanwhile, efforts are made to not incur other side effects (e.g., computational burden and instability). In this respect, the work of this study is summarized as follows:

- The unscented Kalman filter (UKF) algorithm is demonstrated to outweigh EKF in terms of precision [19]. Therefore, UKF is adopted for co-localization in this research.
- The extra computational burden brought by UKF is reduced by introducing the derivative method [20].
- Singular value decomposition (SVD) is utilized to enhance the stability of the derivative UKF.
- The performance of the proposed algorithm is demonstrated by a series of simulation examples.

Finally, a high-precision co-localization algorithm is obtained, which is more effective than the original algorithm in eliminating information conflicts caused by multiple sensors, with similar stability and computational burden. This is the major contribution of this article.

The rest of the paper is organized as follows. In Section 2, a game theory-based algorithm is introduced for co-localization, accompanied by the analysis of its filtering component. Moreover, attention is paid to investigating the KF series algorithms and the UKF-based derivative method as

well as comparing UKF with EKF. In Section 3, the dynamic model of the system is established, and an estimation fusion algorithm is proposed based on the derivative UKF in the game theory-based co-localization algorithm framework. Specifically, KF is used to simplify the calculation of the measurement update without reducing the estimation precision. In addition, SVD is applied to the derivative UKF (DUKF) algorithm to improve the stability of the algorithm and to avoid ill-posed problems during Cholesky decomposition. The game theory-based co-localization algorithm is optimized with the proposed filter algorithm and provides the specific steps of the optimized algorithm in Section 4. In Section 5, the optimized algorithm is verified by numerical simulations of the motion system. Eventually, conclusions are drawn in Section 6.

The relationship between these filtering algorithms is shown in Figure 1.

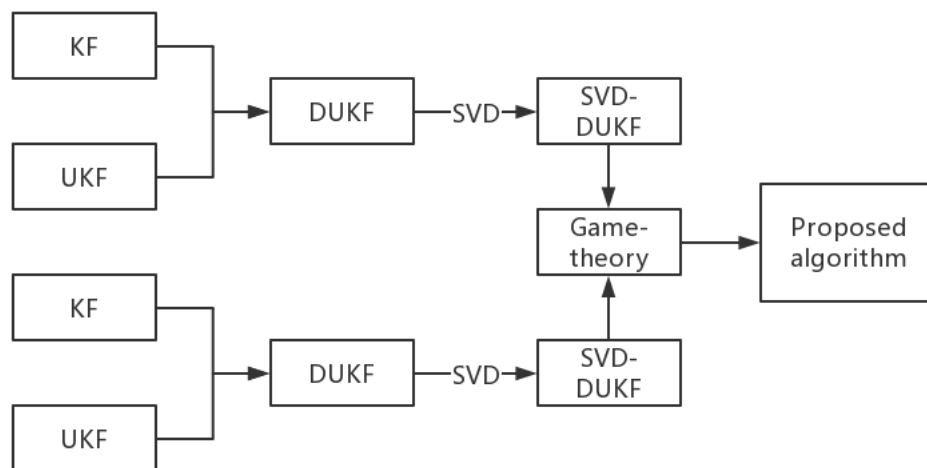


Figure 1. Schematic diagram of the relative observation.

2. Relevant Work and Problem Formulation

2.1. The Co-Localization Based on the Game Theory and the Cooperative Observation

Hua et al. [19] integrated the complete information static game theory into the EKF algorithm for both one-way and two-way observations in the multi-robot systems, which significantly relieved the computational burden and observation conflicts.

A multi-robot system generally outperforms a single-robot system in the non-GPS environment, when it comes to the relative information observation. For example, when robot *A* appears in the view range of robot *B* or two robots meet each other face to face, they will obtain the observation information concerning the location of the other. The co-localization algorithm based on the static game theory utilizes the relative observation information among multiple robots to achieve more precise localization. During the application of the algorithm, some assumptions have to be imposed on these robots:

- The location and direction of any robot can be described by a vector in a unified global coordinate system.
- Each robot is equipped with sensors to perform autonomous localization and observe other robots in the non-GPS environment.
- All the robots can instantly communicate with each other by sending their own information and observation information.
- All the robots are isomorphic, which means that they share exactly the same type of observations that can be described with the same mathematical model.

As shown in Figure 2, the vector $[\Delta d^{(ij)}, \alpha^{(ij)}]$ represents the pose information when robot A observes robot B , where $\Delta d^{(ij)}$ and $\alpha^{(ij)}$ are the distance and the relative angle between robots A and B , respectively. Similarly, the vector $[\Delta d^{(ji)}, \alpha^{(ji)}]$ represents the pose information when robot B observes robot A . In ideal conditions, the relative distances and the angles observed by the two robots are the same. However, there are inevitable noises in the environment, which result in imprecise observations. In this paper, all noises are assumed to be white noises that follow a normal distribution. Since all robots are isomorphic, σ^d is adopted to represent the distance noise, while σ^α stands for the angle noise.

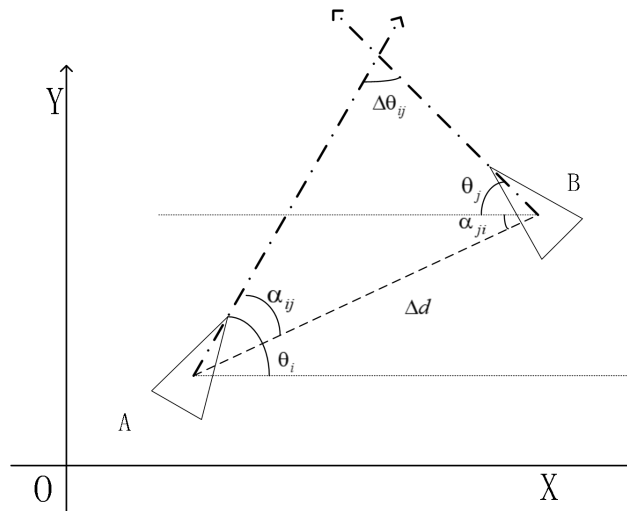


Figure 2. Schematic diagram illustrating the relative observation between two robots.

The observation by sensors is vulnerable to the effects of noises, which will lead to information conflict among the observations of robots. Accordingly, static game theory is introduced to select optimal information when there are conflicts.

Suppose that $G = \{1, 2, \dots, n\}$ is the set of all robots. Each robot corresponds to a separate number. The strategy space of the i -th robot is $s_i = \{0, 1\}$, where 1 indicates the independent location and 0 represents the co-location. In the game theory model, the benefit of an individual participant depends on the chosen localization strategy by all participants. When other robots adopt strategies $(s_1, \dots, s_{i-1}, s_{i+1}, \dots, s_n)$ and the robot i adopts strategy s_i , the benefit function is defined as

$$u_i(s_1, \dots, s_i, \dots, s_n). \quad (1)$$

When n robots perform co-localization, the benefit function can be represented by $G = \{s_1, \dots, s_n, u_1, \dots, u_n\}$ [21].

2.2. Problem Formulation

The algorithm by Hua et al. [19] can solve the observation conflicts of co-localization in multi-robot systems. However, its filtering component adopts a conservative technology, which is accompanied by a compromise in precision. As a countermeasure, UKF, instead of EKF in the original algorithm, is employed to improve the precision of the co-localization. Nevertheless, UKF is also not a perfect algorithm, because it has some inherent defects, such as high computational burden and low system stability. To conclude, the following three measures should be taken to improve the performance of the algorithm by Hua et al. [19]:

- EKF in the original algorithm should be replaced by UKF to improve localization precision.
- The computational burden of UKF should be relieved.
- The algorithm stability should be maintained.

Theoretically, the objectives of this study are mainly threefold:

- to improve the precision of the algorithm by Hua et al. [19] by introducing UKF;
- to relieve the computational burden of UKF without compromise in precision by introducing the derivative method;
- to solve the non-positive definite variance matrix caused by the derivative method by replacing the Cholesky decomposition with SVD.

3. Improvement for the UKF

3.1. Relevant Work on the Kalman Filtering

The KF series algorithms have attracted extensive attention from the engineering community since they were adopted by NASA to solve the navigation problem in the Apollo moon-landing program. However, the traditional KF algorithm is exclusively suitable for linear systems; consequently, it fails to solve problems in nonlinear systems that are ubiquitous in reality (e.g., state estimation and system identification). In this context, nonlinear filtering algorithms are urgently needed.

Recently, a large number of nonlinear filtering algorithms have been proposed, such as the Extended Kalman Filtering (EKF), the Gaussian Sum Filtering, the Gauss–Hermite Filtering, the Particle Filtering (PF), and the Unscented Kalman Filtering (UKF). Among them, EKF is the most commonly used in research on localization and navigation. Its main idea is to first approximately linearize the nonlinear functions and then apply the traditional KF algorithm to filter the linearized system. Although the EKF algorithm is very mature and widely used, it eliminates high-order terms when linearizing a system by the Taylor expansion, thereby leading to truncation errors and ultimately affecting the filtering precision. Moreover, the complexity in calculating the associated Jacobian matrix hinders the wide application of the EKF algorithm in engineering practice [20].

In order to avoid the problem of insufficient accuracy caused by the process of linear approximation, Julier and Uhlmann [22] proposed the UKF algorithm to approximate the posterior probability density function of the nonlinear system based on a set of sigma points by the unscented transformation (UT).

The UKF outperforms the EKF in the following aspects:

- UKF does not require calculating the Jacobian matrix using the nonlinear state function and the measurement function, which makes it easy to implement in the engineering and simulation.
- Regardless of the degree of nonlinearity of the system, the posterior mean and variance approximated by UT are equivalent to those of a nonlinear Gaussian system approximated by a third- or higher-order Taylor expansion [20]. In other words, UKF presents a higher precision than EKF in high-order systems.

Based on the above advantages, we believe that the accuracy of localization will be improved by using the UKF algorithm for filtering, which can also reduce information conflicts among multiple robots during the process of co-localization. The flow and step of the game theory-based UKF co-localization algorithm has been given in our previous work [23], so they are not repeated here.

3.2. Improvement of the Computational Burden Based on the Derivative Method

Compared with EKF, the high precision of UKF is unfortunately accompanied by a significantly increased computational burden. As a countermeasure, UKF is properly modified in this study by introducing the derivative method.

In the original algorithm, Hua et al. [19] adopted $X = [d \cos \theta, d \sin \theta]$ to describe the state of the system. Instead of this, $X = [x, y, v_x, v_y]^T$ is used for such a description in this study. In this way, the time update can be achieved by the classical KF. Nonlinear filtering is adopted for the observation update based on UT.

The derivative UKF (DUKF) algorithm has the same form as the classical KF algorithm in terms of the time update. This allows DUKF to avoid UT of the time update, which is required in the traditional UKF. As a consequence, the state estimation and its covariance can be calculated concisely, thus reducing the computational burden efficiently. In the observation update phase, the DUKF algorithm relies on the traditional UKF to update the predicted observations, thereby inheriting the excellent performance of the traditional UKF algorithm for nonlinear systems.

3.3. System Model Establishment

The considered nonlinear discrete-time system is represented by

$$X_{k+1} = \varphi X_k + w_k = \begin{bmatrix} 1 & t & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & t \\ 0 & 0 & 0 & 1 \end{bmatrix} X_k + w_k, \quad (2)$$

$$Z_k = H_k(X_k) + v_k = \begin{bmatrix} r_k^1 \\ r_k^2 \\ \dots \\ r_k^i \end{bmatrix} + v_k, \quad (3)$$

where X_k and Z_k denote the state and measurement vectors at the time k , respectively, φ is the state prediction matrix, t is the sampling time, $H_k(\cdot)$ is the nonlinear measurement function, and r_k^i is the distance between the i -th UWB sensor and the robot at time k . To increase the real-time performance of the system, we adopt the distance r instead of the coordinates x and y as the observation value in the measurement vector [24]. r_k^i is defined as

$$r_k^i = \sqrt{(x_i - x_k)^2 + (y_i - y_k)^2}. \quad (4)$$

The w_k and v_k are the uncorrelated zero-mean Gaussian white noise processes, with covariances as

$$\begin{aligned} E[w_k w_k^T] &= Q_k \\ E[v_k v_k^T] &= R_k. \end{aligned} \quad (5)$$

It can be seen from the system model that the system has a linear time update matrix and a nonlinear measurement function. Therefore, it meets the conditions required for the application of the DUKF algorithm.

To transplant the algorithm to the physical platform in the future, the values of error variance and system state are determined according to the robot and sensors we purchased. Specifications of simulated sensors and mobile robots are shown in Figure 3. Each base station is equipped with one UWB sensor to measure the distance, and each robot is equipped with three UWB sensors that form an antenna array to obtain the azimuth information during the relative observation.

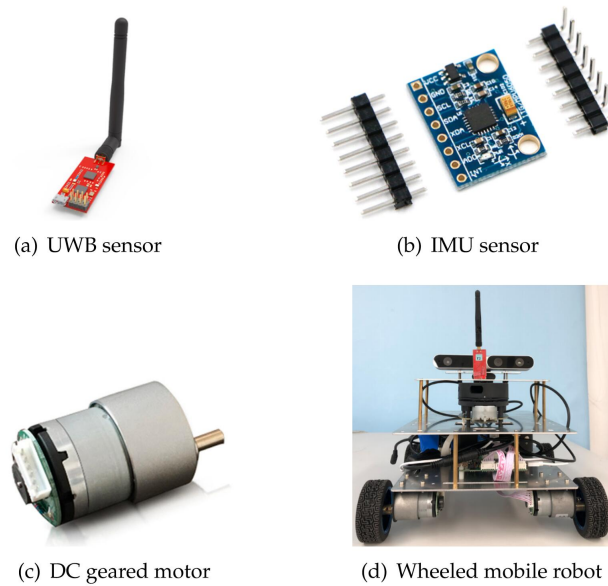


Figure 3. The sensors and the mobile robot involved in this article. (a) The Ultra-Wide Band (UWB) sensors(YCHIOT, Wenzhou, Zhejiang, China), which are commercial products provided by the company of YCHIOT, with the module of Mini3s; (b) the spatial motion sensor chip MPU6050 as the IMU sensor(Digi-Key Electronics, Thief River Falls, MN, US); (c) the odometer84 constructed by the DC gear motors MG513 with an encoder(Fenghua Transmission, Kunshan, Jiangsu, China); (d) the wheeled mobile robot platform(Ruiqu Technology, Foshan, Guangdong, China) that realizes the precise localization by carrying the above sensors.

3.4. Comparison between UKF and DUKF in Terms of Computational Complexity

To evaluate the computational burden of the identification algorithm, we count the number of multiplication and addition operations of the algorithm. Each multiplication or addition operation is called a flop, and the total number of flops represents the computational burden.

For the considered system, the computational burden of DUKF is significantly lower than that of UKF. Specifically, it is necessary for UKF to select the sigma point set to derive the nonlinear system during the time update and measurement update. Consequently, UKF brings about a large but unnecessary computational burden because the time prediction equation of the considered system is linear. In contrast, DUKF uses KF in the linear part of the system, avoiding redundant calculations for measurement updates. Table 1 comprehensively compares UKF and DUKF in terms of the computational costs, especially for the time update part since they share the same measurement updates. It is deduced, through the analysis of the computational cost, that UKF requires a total of $\frac{n^3}{3} + 17n^2 + 3n$ flops more than DUKF in each cycle, and they present comparative precision.

Table 1. The difference in Computation Cost of unscented Kalman filter (UKF) and derivative UKF (DUKF).

Step	UKF	DUKF	Flops Difference
Calculate the Sigma points (Time update)	Cholesky decomposition to the $P_{i k}$	none	$\frac{n^3}{3} + n^2$
	$\begin{cases} X_{i1 k} = X_{i k} \\ X_{iu k} = X_{i k} \pm a(\sqrt{nP_{i k}}) \end{cases}$		$3n^2$
Predict the time update of Sigma points	$\hat{X}_{iu k+1} = \varphi(x_{iu k})$	none	n^2
Estimate the time update	$\hat{X}_{k+1 k} = \sum_{u=1}^{2n+1} \omega_u \hat{X}_{iu k+1}$	$\hat{X}_{k+1 k} = \varphi \hat{X}_k$	$4n^2 + n$
	$P_{k+1 k} = \sum_{u=1}^{2n+1} \omega_u (\hat{X}_{k+1 k} - \hat{X}_{iu k+1})(\hat{X}_{k+1 k} - \hat{X}_{iu k+1})^T + Q_k$	$P_{k+1 k} = \varphi P \varphi^T + Q_k$	$8n^2 + 2n$

As shown in Figure 4, the calculation time of DUKF is in general concentrated below that of UKF. Table 2 shows that the variance of DUKF is 24.1% higher than that of UKF.

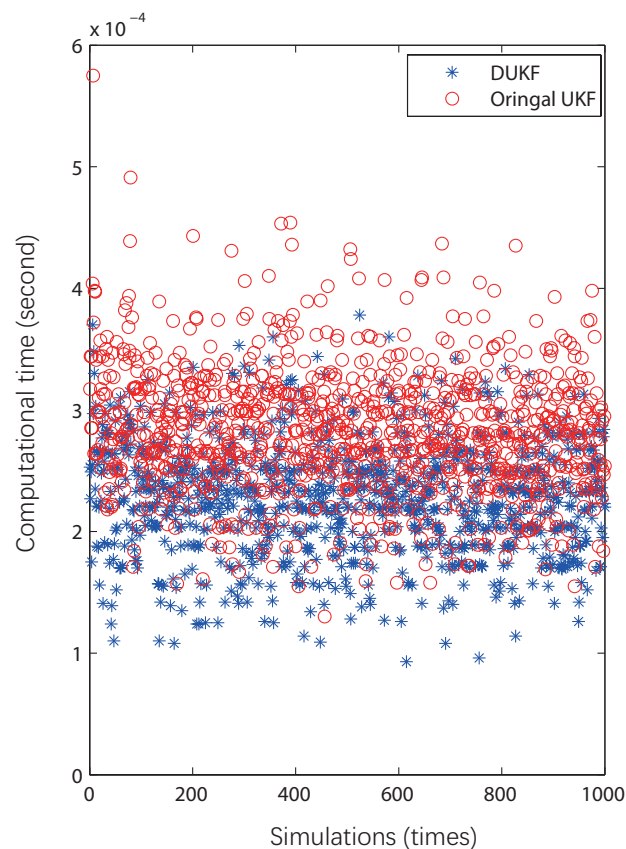


Figure 4. The computational time of UKF and DUKF during the 1000 Monte Carlo simulations of a simple model, where blue asterisks represent the calculation time of DUKF, while the red circles stand for that of UKF.

Table 2. Variance of the UKF and the DUKF.

Algorithm	Variance
UKF	0.2282
DUKF	0.2832

The 18% longer calculation time of DUKF than EKF is compensated by the much higher precision, and therefore we believe that the lengthened calculation time is worthwhile.

3.5. Improvement of the Stability of the DUKF

Compared with the traditional UKF algorithm, the DUKF algorithm adopted in this study is sensitive to the error of the initial system state estimation. Specifically, when the estimated initial system state is much different from the real counterpart, the DUKF algorithm is prone to divergence.

Due to the randomness of the covariance matrix, ill-posed problems are often encountered during the decomposition process, affecting the filtering performance and even causing divergence [25,26]. Countermeasures have been proposed in the study by Hu [20], who comprehensively analyzed the error characteristics and discussed the stability of the DUKF algorithm. According to his research, the observation covariance P_{zz} is required to be a positive definite matrix to bound the estimation error of the DUKF algorithm. In this context, the noise variance R_k needs to be a positive definite matrix as a precondition. However, R_k may be a positive semi-definite matrix, which causes P_{zz} to be a positive

semi-definite matrix as well. To stabilize the algorithm, Hu added a small positive definite matrix ΔR_k to R_k :

$$R_k^* = R_k + \Delta R_k. \quad (6)$$

However, the introduction of ΔR_k subjects the algorithm to reduced filtering precision. In the study by Geng et al. [27], the SVD is introduced into the UKF algorithm to deliver a more complex iteration of the covariance matrix, which results in higher stability. Similarly, the SVD is introduced into the derived CKF algorithm to improve the stability in our previous research [28], which achieved good performance. In this study, the SVD is integrated with the previously introduced DUKF to form an SVD-DUKF method, followed by a simple proof and numerical experiments that verify its effectiveness.

Different from the Cholesky decomposition, the SVD can decompose positive semi-definite matrices. Therefore, ΔR_k is not needed anymore in the SVD-DUKF method. In other words, the stability of the algorithm can be guaranteed without increasing the measurement errors. This method not only optimizes the iterative process and reduces the amount of calculation, but also ensures the stability of the algorithm iteration, contributing to greatly improved system convergence.

3.6. The Method of the SVD-DUKF

Matrix A , assumed to be an $m \times n$ order matrix, can be decomposed into orthogonal matrices U and V according to the singular value decomposition theory:

$$A = U\Lambda V^T, \Lambda = \begin{bmatrix} S & 0 \\ 0 & 0 \end{bmatrix} \quad (7)$$

$$S = \text{diag}(\sigma_1, \sigma_2, \dots, \sigma_p) \quad (8)$$

where σ_p is the singular value of A , and the values are changed in a descending order.

When matrix A is a positive definite matrix, Formula (7) can be rewritten as

$$A = USU^T. \quad (9)$$

In Formula (8), all of the σ_p are greater than zero.

The error covariance matrix in the DUKF algorithm is subjected to the singular value decomposition, and the decomposition matrix is iterated by the properties of the matrix, which contributes to the improved SVD-DUKF algorithm.

Specific steps of the SVD-DUKF algorithm are detailed in Algorithm 1.

To demonstrate the performance of the SVD-DUKF algorithm, a simulation is implemented in a simple model, which demonstrates that the number of convergence of the SVD-DUKF algorithm is 56.7% higher than that of the DUKF algorithm (Table 3).

Table 3. The success convergence rate of the DUKF and the singular value decomposition (SVD)-DUKF in 1000 simulations.

Algorithm	Convergence Rate
DKF	43.7%
SVD-DUKF	100%

Algorithm 1: The flow of the derivative SVD-DUKF algorithm.

Input: $\hat{x}_{k-1}, \hat{P}_{k-1}$

Output: \hat{x}_k, \hat{P}_k

1. Time update to the system based on the classical Kalman filter:

$$\begin{aligned}\hat{x}_{k/k-1} &= \varphi_{k/k-1} \hat{x}_{k-1} \\ \hat{P}_{k|k-1} &= \varphi \hat{P}_{k-1} \varphi^T + Q_k\end{aligned}$$

2. SVD of state covariance matrix $\hat{P}_{k|k-1}$ can be obtained:

$$\hat{P}_{k|k-1} = UD^2U^T$$

3. Select a set of SIGMA points as follows:

$$\begin{cases} X_{i1|k} = X_{i|k} \\ X_{iu|k} = X_{i|k} \pm a \left((\sqrt{n + \lambda} UD) \right) \end{cases}$$

4. Update the observations based on the set of sigma points as follows:

$$\begin{aligned}Z_{iu|k+1} &= H(X_{iu|k+1}) \\ Z_{i|k+1} &= \sum_{u=0}^{2n} \omega_u Z_{iu|k+1}, (u = 1, 2, 3, \dots)\end{aligned}$$

5. Calculate the covariance of the observation:

$$P_{iziz|k+1} = \sum_{u=0}^{2n} \omega_u (Z_{iu|k} - Z_{i|k+1})(Z_{iu|k} - Z_{i|k+1})^T$$

6. Calculate the cross-covariance variance of observation and state:

$$P_{ixiz|k+1} = \sum_{u=0}^{2n} \omega_u (X_{iu|k} - X_{i|k+1})(Z_{iu|k} - Z_{i|k+1})^T$$

where $\begin{cases} \omega_u = 1 - \frac{1}{a^2}, u = 0 \\ \omega_u = \frac{1}{2na^2}, u = 1, 2, 3, \dots, n \end{cases}$.

7. Obtain the Kalman gain:

$$K = P_{ixiz|k+1} P_{iziz|k+1}^{-1}$$

8. Update the system status and covariance of the system:

$$\begin{aligned}X_{i|k+1} &= X_{i|k} + K(\hat{Z}_i - Z_{i|k+1}) \\ P_{i|k+1} &= P_{i|k} - KP_{ikik|k+1}K^T\end{aligned}$$

9. end.

4. The Method of the Static Game Theory-Based SVD-DUKF Algorithm for Co-Localization

If we set robot i to be the observer and robot j to be observed in our one-way observation model, then robot i will provide the observation information, while robot j will select the strategy of localization, which depends on the information provided by robot i .

The observation information obtained by robot i concerning the location of robot j can be expressed as follows:

$$X_j^{(i)} = H_{ij} X_i = X_i + v_k \quad (10)$$

$$\begin{bmatrix} x_j^{(i)} \\ y_j^{(i)} \end{bmatrix} = H_{ij} \begin{bmatrix} x_i \\ y_i \end{bmatrix} = \begin{bmatrix} x_i \\ y_i \end{bmatrix} + v_k, \quad (11)$$

where the superscript (i) means that the information is provided by robot i . Other parameters are listed in Figure 1. The SVD of the state covariance matrix P_i can be obtained:

$$P_i = U_i D_i^2 U_i^T. \quad (12)$$

Based on the result of the SVD and the previous independent localization, the sigma point set can be obtained:

$$\begin{cases} \xi_{0,i} = X_i \\ \xi_{u,i} = X_i + a(\sqrt{n + \lambda} U_i D_i), u = 1, 2, \dots, n \\ \xi_{u,i} = X_i - a(\sqrt{n + \lambda} U_i D_i), u = n + 1, n + 2, \dots, 2n \end{cases} \quad (13)$$

After that, the sigma point set is transformed based on the measurement equation:

$$z_{ij} = H_{ij}(\xi_{u,i}), u = 1, 2, \dots, 2n. \quad (14)$$

The predicted measurement and its covariance are then calculated:

$$z_{ij} = \sum_{u=0}^{2n} \omega_u \xi_{u,i}, \quad (15)$$

$$P_{z,ij} = \sum_{u=0}^{2n} \omega_u (\xi_{u,i} - z_{ij})(\xi_{u,i} - z_{ij})^T + R_{ij}. \quad (16)$$

Subsequently, the cross-covariance between the state and the measurement can be calculated:

$$P_{xz,ij} = \sum_{u=0}^{2n} \omega_u (\xi_{u,i} - x_{ij})(\xi_{u,i} - z_{ij})^T, \quad (17)$$

where

$$\begin{cases} \omega_0 = 1 - \frac{1}{a^2} \\ \omega_u = \frac{1}{2na^2}, u = 1, 2, \dots, 2n \end{cases}$$

a is an adjustment parameters.

The gain matrix K can be determined by the covariance matrix $P_{z,ij}$ and the cross covariance matrix $P_{xz,ij}$:

$$K_{ij} = P_{xz,ij} P_{z,ij}^{-1}. \quad (18)$$

Finally, based on the gain matrix K_{ij} and the state covariance matrix P_i , the covariance matrix P_{ij} of co-localization by robot i to robot j can be obtained:

$$P_{ij} = P_i + K_{ij} P_{z,ij} K_{ij}^{-1}. \quad (19)$$

The difference between $X_j^{(ij)}$ and X_j is taken as the judging criteria. The eigenvalues of $\sqrt{P_{ij}}$, namely a_{ij} and b_{ij} , are the semi-major and semi-short axes of the pose uncertainty ellipse, respectively. Similarly, by calculating the eigenvalues of $\sqrt{P_{ij}}$, namely a_{ij} and b_{ij} , we have

$$\frac{f(P_{ij}, P_j)}{\|X_j^{(i)} - X_j\|} = \frac{\min \{3 \times \sqrt{a_{ij}^2 + b_{ij}^2}, 3 \times \sqrt{a_j^2 + b_j^2}\}}{\sqrt{(x_j^{(i)} - x_j)^2 + (y_j^{(i)} - y_j)^2}}. \quad (20)$$

Based on Formula (20), the following benefit function is obtained:

$$f = \frac{\min \left\{ 3 \times \sqrt{a_{ij}^2 + b_{ij}^2}, 3 \times \sqrt{a_j^2 + b_j^2} \right\}}{\sqrt{\left(x_j^{(i)} - x_j\right)^2 + \left(y_j^{(i)} - y_j\right)^2}}. \quad (21)$$

The game benefits of i and j are shown in Table 4. Moreover, in the case of the one-way observation, robot i performs autonomous localization and the benefit of the game is 1. If $f > 1$, $u_j [s_i, s_j = 1] = f > u_j [s_i, s_j = 0] = 1$, the game can reach the unique Nash equilibrium (independent localization; co-localization). If $f < 1$, $u_j [s_i, s_j = 1] = f < u_j [s_i, s_j = 0] = 1$, the game can reach the unique Nash equilibrium (independent localization; independent localization). If $f = 1$, the benefit function of the game is always equal to 1. Co-localization tends to increase the computational burden, and independent localization is thus generally selected by the robots.

Table 4. Table of game benefits.

Strategy	$s_j = 1$	$s_j = 0$
$DS_i = 1$	f_i, f_j	$f_i, 1$
$s_i = 0$	$1, f_j$	$1, 1$

After identifying the selection strategy of the game with two robots in the case of one-way relative observation, the proposed SVD-DUKF algorithm is adopted to obtain the location information of each robot. Specific steps of the SVD-DUKF algorithm are shown in Algorithm 1.

Finally, the calculation results are assigned to the corresponding robots participating in the co-localization. Specific steps of the game theory-based SVD-DUKF co-localization algorithm (with localization of robot j as an example) are shown in Algorithm 2. The flow chart of the algorithm is shown in Figure 5.

Algorithm 2: The flow of the game theory-based SVD-DUKF algorithm.

Input: $x_{i|k}, P_{i|k}, x_{j|k}, P_{j|k}$

Output: $x_{i|k+1}^{(j)}, P_{i|k+1}, x_{j|k+1}^{(i)}, P_{j|k+1}$

1. Calculate the $x_{i|k+1}, P_{i|k+1}, x_{j|k+1}, P_{j|k+1}$ by Algorithm 1.
 2. Calculate the a_{ji}, b_{ji}, a_j, b_j by the SVD.
 3. Substitute the above results into the benefit function (Formula (21)).
 4. Based on the result of Formula (21), choose the game strategy.
 5. Based on the strategy, **return** the $x_{i|k+1}^{(j)}, P_{i|k+1}, x_{j|k+1}^{(i)}, P_{j|k+1}$.
 6. **end.**
-

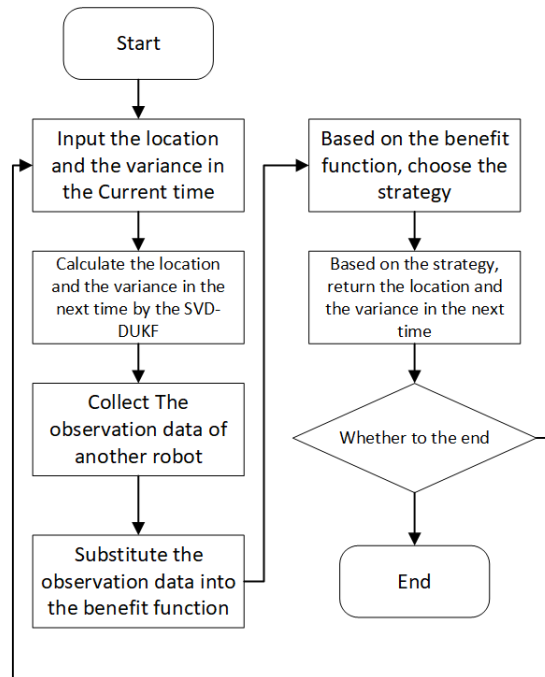


Figure 5. The flow chart of the static game theory algorithm based on SVD-DUKF.

5. Simulations

In order to demonstrate the performance of the proposed algorithm, a MATLAB simulation is implemented based on the open-source program that is provided by OpenSLAM [29].

As shown in Figure 6, the connection between the asterisk and the triangle indicates that the UWB sensor (asterisk) is measuring the distance of the robot (triangle). All robots can observe the relative angle and distance of the objects in a range of 180 degrees in front of them, with a maximum observation range of 15 m, and the paths they move are represented by green, purple, and blue lines. The simulation space is a 200 m × 200 m square. The dynamic model of the robot and the error variance of the sensor are determined by the system model proposed in Section 3.3.

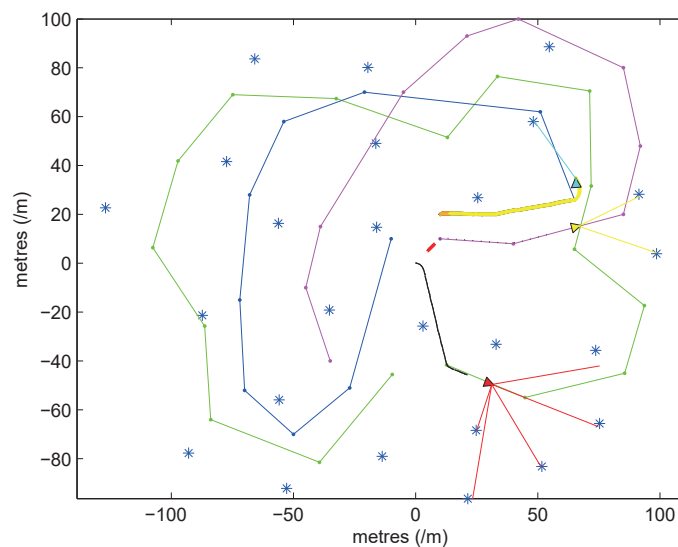


Figure 6. Simulation space.

The SVD-DUKF algorithm is adopted when each robot is localized individually, and the optimized game theory-based algorithm is used when a relative observation occurs.

To demonstrate the performance of the proposed algorithm, five algorithms are evaluated by the simulations, including the original EKF, the static game theory-based EKF, the static game theory-based UKF, and the static game theory-based SVD-DUKF. Moreover, another nonlinear filtering algorithm (square root cubature Kalman filter, SR-CKF) proposed by Liu et al. [30], which is also of higher precision and convergence, is assessed as the control group. The simulation results are shown in Figures 7–10.

Localization errors of the other four algorithms are all less than that of the original EKF (Figure 7). Moreover, the game theory-based UKF has fewer localization errors than the original EKF and the game theory-based EKF (Figure 8). In addition, the SVD-DUKF algorithm further reduces the localization error (Figure 9). As shown in Figure 10, the SR-CKF as a control group also presents higher performance than the game theory-based EKF, and it performs similarly well with the algorithm proposed in this research.

However, due to the introduction of the derivative method, the game theory-based SVD-DUKF proposed in this paper is subjected to a smaller computational burden than SR-CKF, which is proven in the average calculation time of 50 times of simulations (Table 5). Figures 11 and 12 show the average localization errors and variances of all the five algorithms. Consistent with the previous analyses, the game theory-based SVD-DUKF presents better performance than other original algorithms and UKF series algorithms. Its average error is almost the same as that of SR-CKF, but its variance is much smaller than that of SR-CKF. Therefore, it is safe to conclude that the algorithm proposed in this study has the best performance among the five algorithms.

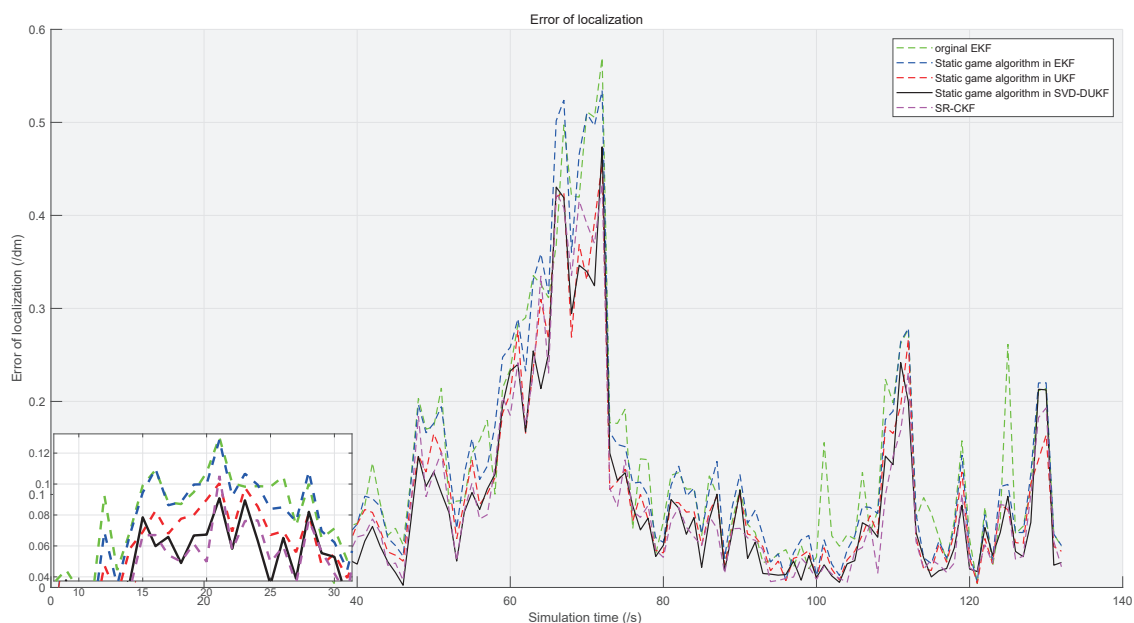


Figure 7. Localization errors of the five algorithms, with the small box in the lower left corner as a partially enlarged view. The control group and the original algorithm are represented by dashed lines in various colors, and the proposed algorithm is represented by the black solid line.



Figure 8. The error of localization on the game theory UKF, the original extended Kalman filtering (EKF), and the game theory EKF.

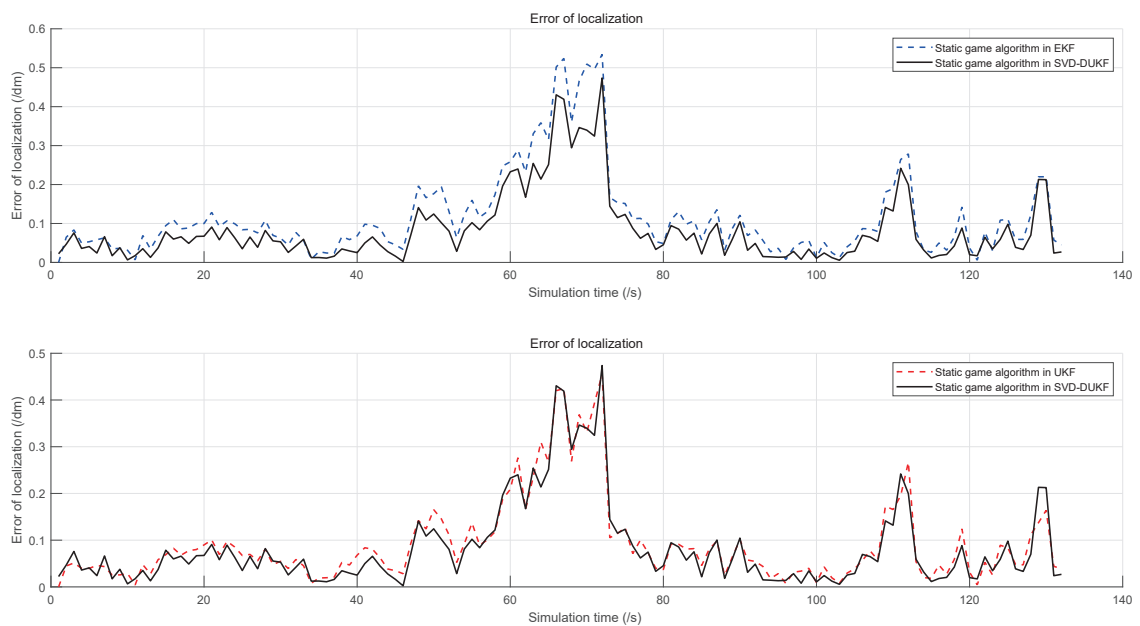


Figure 9. The error of localization on the game theory EKF, the game theory UKF, and the game theory SVD-DUKF.

Table 5. The average calculation time of SR-CKF and the static game theory-based SVD-DUKF in 50 simulations.

Algorithm	Average Calculation Time(s)
static game theory SVD-DUKF	21.6658
SR-CKF	24.8356

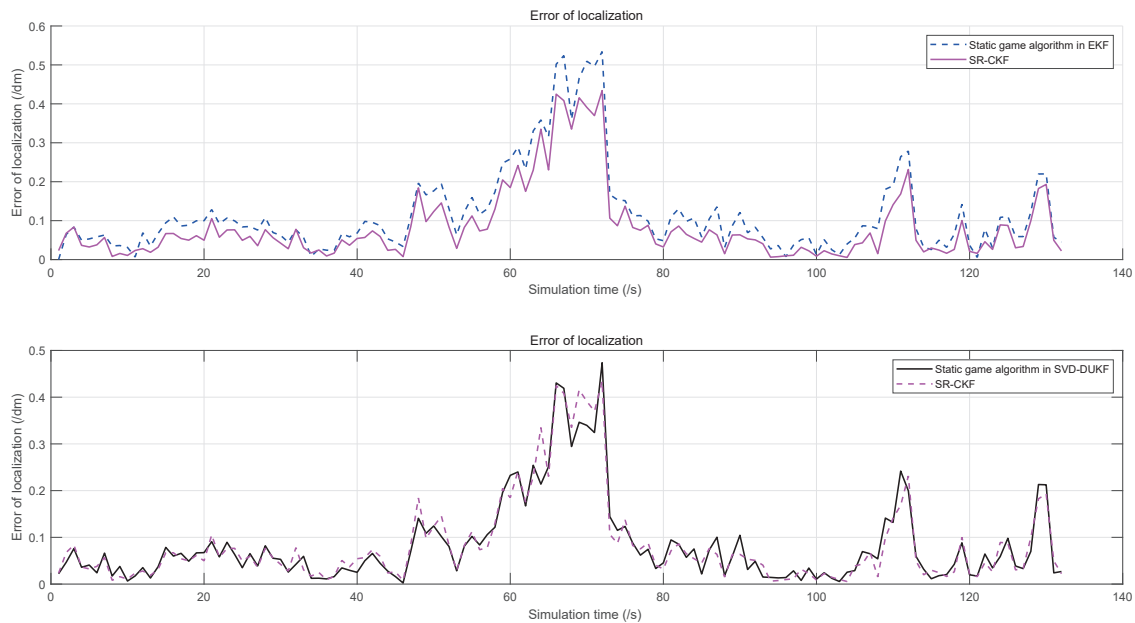


Figure 10. The error of localization on the game theory EKF, square root cubature Kalman filter (SR-CKF), and the game theory SVD-DUKF.

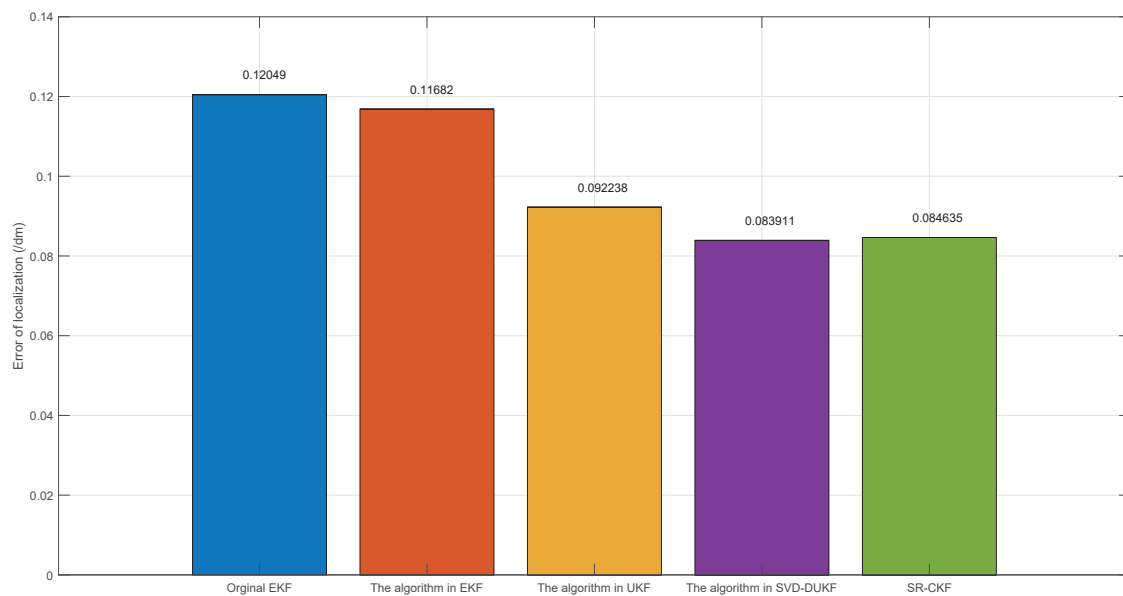


Figure 11. The mean of the localization error.

It is confirmed by the simulation results that SR-CKF and SVD-DUKF have better performances than the others. Moreover, as has been stated in Section 2, UKF has higher precision than the game theory-based EKF, which explains the poor performance of EKF in the simulation. Different from the original UKF, SVD-DUKF adopts the linear update in the time update, thus removing the linearization error. Therefore, SVD-DUKF has higher precision than the original UKF in the simulations. Although SR-CKF also presents very high precision in the simulation, it needs to calculate the volume point set during the time update phase, thereby resulting in a much heavier calculation burden than SVD-DUKF. Consequently, SVD-DUKF is believed to outperform SR-CKF, although they have almost the same precision and variance.

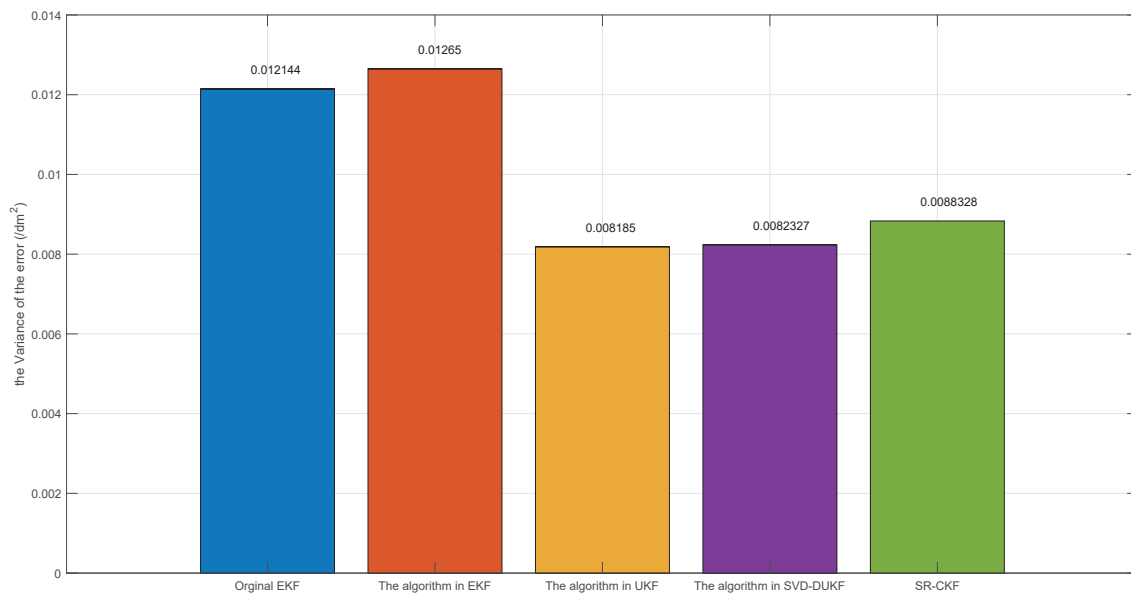


Figure 12. The variance of the localization error.

6. Conclusions

The main purpose of this study is to improve a game theory-based co-localization algorithm that is filtered by EKF. After introducing the advantages of UKF in terms of precision, the filtering part of the original game theory-based co-localization algorithm is modified. Moreover, this study manages to relieve the computational burden and to enhance the algorithm stability through combining DUKF with SVD-UKF, and the combination is utilized in the game theory-based co-localization algorithm. Simulation results show that the proposed algorithm can increase the co-localization precision, without increasing the computational burden and deteriorating the stability. Therefore, it is safe to conclude that the proposed algorithm performs well in the scenarios explored in this study, and it can effectively solve the information conflicts caused by multi-sensors in the wireless sensor network.

We aim to perform the following in the future:

- test the performance of the proposed algorithm on the physical platform, rather than only through numerical simulation;
- since the CKF series algorithms have better stability and precision theoretically, apply similar improvements to the CKF series algorithms in subsequent research;
- apply the proposed algorithm to SLAM;
- testing the performance of the optimized algorithm on a complex dynamic model.

Author Contributions: Conceptualization, C.T.; data curation, C.T.; writing, C.T.; supervision, L.D.; funding acquisition, L.D. All authors have read and agreed to the published version of the manuscript.

Funding: This work was supported by the National Natural Science Foundation of China (NSFC) under Grant 61991414 and Grant 61873301.

Conflicts of Interest: The authors declare that there is no conflict of interest.

References

1. Thrun, S.; Fox, D.; Burgard, W.; Dellaert, F. Robust Monte Carlo Localization for mobile robots. *Artif. Intell.* **2001**, *128*, 99–141. [[CrossRef](#)]
2. Filliata, D.; Meyerb, J.A. Map-based navigation in mobile robots: I. A review of localization strategies. *Cogn. Syst. Res.* **2003**, *4*, 243–282. [[CrossRef](#)]
3. Frese, U. A discussion of simultaneous localization and mapping. *Auton. Robot.* **2006**, *20*, 25–42. [[CrossRef](#)]

4. Roumeliotis, S.I.; Bekey, G.A. Distributed multirobot localization. *IEEE Trans. Robot. Autom.* **2002**, *18*, 781–795. [[CrossRef](#)]
5. Zhang, S.; Cao, Y. Cooperative Localization Approach for Multi-Robot Systems Based on State Estimation Error Compensation. *Sensors* **2019**, *19*, 3842. [[CrossRef](#)] [[PubMed](#)]
6. Wang, Z.; Jin, X.; Wang, X.; Xu, J.; Bai, Y. Hard Decision-Based Cooperative Localization for Wireless Sensor Networks. *Sensors* **2019**, *19*, 4665. [[CrossRef](#)] [[PubMed](#)]
7. Pathirana, P.N.; Bulusu, N.; Savkin, A.V.; Jha, S. Node localization using mobile robots in delay-tolerant sensor networks. *IEEE Trans. Mob. Comput.* **2005**, *4*, 285–296. [[CrossRef](#)]
8. Baggio, A.; Langendoen, K. Monte Carlo localization for mobile wireless sensor networks. *Ad Hoc Netw.* **2008**, *6*, 718–733. [[CrossRef](#)]
9. Chen, H.; Shi, Q.; Tan, R.; Poor, H.V.; Sezaki, K. Mobile element assisted cooperative localization for wireless sensor networks with obstacles. *IEEE Trans. Wirel. Commun.* **2010**, *9*, 956–963. [[CrossRef](#)]
10. Guo, K.; Qiu, Z.; Miao, C.; Zaini, A.H.; Chen, C.L.; Meng, W.; Xie, L. Ultra-Wideband-Based Localization for Quadcopter Navigation. *Unmanned Syst.* **2016**, *4*, 23–34. [[CrossRef](#)]
11. Pirník, R.; Hruboš, M.; Nemeč, D. Integration of Inertial Sensor Data into Control of the Mobile Platform. In Proceedings of the 2015 Federated Conference on Software Development and Object Technologies, Žilina, Slovakia, 19–20 November 2015; Springer International Publishing: Berlin/Heidelberg, Germany, 2015.
12. Limei, L.; Zixing, C. Multi-robot cooperative localization based on particle swarm optimization. *J. Cent. South Univ. Sci. Technol.* **2011**, *42*, 682–687.
13. Turygin, Y.; Bozek, P.; Abaramov, I.; Nikitin, Y. Reliability determination and diagnostics of a mechatronic system. *Adv. Sci. Technol. Res. J.* **2018**, *12*, 274–290. [[CrossRef](#)]
14. Kilin, A.; Bozek, P.; Karavaev, Y.; Klekovkin, A.; Shestakov, V. Experimental Investigations of a Highly Maneuverable Mobile Omniwheel Robot. *Int. J. Adv. Robot. Syst.* **2017**, *14*, 1729881417744570. [[CrossRef](#)]
15. Božek, P.; Al Akkad, M.A.; Blištan, P.; Ibrahim, N.I. Navigation Control and Stability Investigation of a Mobile Robot Based on a Hexacopter Equipped with an Integrated Manipulator. *Int. J. Adv. Robot. Syst.* **2017**, *14*, 1729881417738103. [[CrossRef](#)]
16. Yu, C.; Chen, J.; Verhaegen, M. Subspace identification of individual systems in a large-scale heterogeneous network. *Automatic* **2019**, *109*, 108517. [[CrossRef](#)]
17. Yu, C.; Ljung, L.; Wills, A.; Verhaegen, M. Constrained Subspace Method for the Identification of Structured State-Space Models. *IEEE Trans. Autom. Control* **2019**, *99*, 1. [[CrossRef](#)]
18. Yu, C.; Chen, J.; Li, S.; Verhaegen, M. Identification of affinely parameterized state-space models with unknown inputs. *Automatic* **2020**, *122*, 109271. [[CrossRef](#)]
19. Chenghao, H.; Lihua, D.; Automation, S.O. A new algorithm merging static game with complete information into EKF for multi-robot cooperative localization. *J. Cent. South Univ. Technol.* **2013**, *44*, 4534–4541.
20. Gao, H. *Extension Research on UKF Algorithm and Data Fusion Technology for Integrated Navigation*; Northwestern Polytechnical University: Xi'an, China, 2016.
21. Gibbons, R. *A Primer in Game Theory*; Pearson Prentice Hall: Harlow, UK, 1992.
22. Julier, S.; Uhlmann, J.; Durrantwhyte, H.F. A new method for the nonlinear transformation of means and covariances in filters and estimators. *IEEE Trans Autom. Control* **2001**, *45*, 477–482. [[CrossRef](#)]
23. Tang, C.; Dou, L. A game-theory based UKF algorithm for multi-robots cooperative localization. In Proceedings of the IEEE 15th International Conference on Control and Automation (ICCA), Edinburgh, UK, 16–19 July 2019; pp. 899–904.
24. Li, J.; Bi, Y.; Li, K.; Wang, K.; Lin, F.; Chen, B.M. Accurate 3D Localization for MAV Swarms by UWB and IMU Fusion. In Proceedings of the 2018 IEEE 14th International Conference on Control and Automation (ICCA), Anchorage, AK, USA, 12–15 June 2018.
25. Crassidis, J.L.; Markley, F.L. Unscented filtering for spacecraft attitude estimation. *J. Guid. Control Dyn.* **2003**, *26*, 536–542. [[CrossRef](#)]
26. Merwe, R.V.D.; Wan, E.A. The square-root unscented Kalman filter for state and parameter-estimation. In Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing, Salt Lake City, UT, USA, 7–11 May 2001; pp. 3461–3464.
27. Dusheng, G.; Jianchao, W.; Yalin, J. Adaptive SVD-UKF algorithm and application to integrated navigation. *J. Chin. Inert. Technol.* **2010**, *18*, 737–741.

28. He, C.; Tang, C.; Yu, C. A Federated Derivative Cubature Kalman Filter for IMU-UWB Indoor Positioning. *Sensors* **2020**, *20*, 3514. [[CrossRef](#)] [[PubMed](#)]
29. Bailey, T.; Nieto, J. SLAM Package of Tim Bailey. Available online: <http://openslam.org/bailey-slam.html> (accessed on 15 March 2013).
30. Liu, K.; Liu, B.; Wang, Y.; Zhao, Y.; Cui, S.; Feng, X. Adaptive square-root CKF with application to DR/LBL integrated heading estimation for HOV. In Proceedings of the 27th Chinese Control and Decision Conference (2015 CCDC), Qingdao, China, 23–25 May 2015; pp. 1851–1855.



© 2020 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/>).