Article

Cooperative Navigation for Low-Cost UAV Swarm Based on Sigma Point Belief Propagation

Mingxing Chen 1*, Zhi Xiong 1, Fengyi Song 1, Jun Xiong 2 and Rong Wang 1

1 Navigation Research Center, School of Automation Engineering, Nanjing University of Aeronautics and Astronautics, Nanjing 211106, China; cmxnuaa@nuaa.edu.cn (M.C.); syf310710@nuaa.edu.cn (F.S.); rongwang@nuaa.edu.cn (R.W.)
2 School of Internet of Things, Nanjing University of Posts and Telecommunications, Nanjing 210023, China; xiongjun@njupt.edu.cn
* Correspondence: xiongzhi@nuaa.edu.cn; Tel.: +86-138-1380-8576

Abstract: As navigation is a key to task execution of micro unmanned aerial vehicle (UAV) swarm, the cooperative navigation (CN) method that integrates relative measurements between UAVs has attracted widespread attention due to its performance advantages. In view of the precision and efficiency of cooperative navigation for low-cost micro UAV swarm, this paper proposes a sigma point belief propagation-based (SPBP) CN method that can integrate self-measurement data and inter-UAV ranging in a distributed manner so as to improve the absolute positioning performance of UAV swarm. The method divides the sigma point filter into two steps: the first is to integrate local measurement data; the second is to approximate the belief of position based on the mean and covariance of the state, and pass message by augmentation, resampling and cooperative measurement update of the state to realize a low-complexity approximation to traditional message passing method. The simulation results and outdoor flight test results show that with similar performance, the proposed CN method has a calculation load more than 20 times less than traditional BP algorithms.

Keywords: cooperative navigation; UAV swarm; sigma point filter; belief propagation

1. Introduction

In recent years, UAVs have received extensive attention in military and civil fields such as battlefield reconnaissance, aerial photography, agricultural plant protection, express transportation, disaster rescue, and power inspection, and UAV cluster technology has become a research hotspot due to its high efficiency and damage resistance [1,2]. Navigation information is crucial for reliable flight of UAVs. Especially in an intensive UAV formation flight system, the UAV swarm requires high-precision position, velocity and other information to conduct task planning, precise control, and collision avoidance [3,4]. At present, the Global Navigation Satellite System (GNSS) is the most widely used means of navigation for UAVs; the GNSS based on Real Time Kinematic (RTK) technology can provide UAVs with centimeter-level positioning precision [5]. However, in certain restricted environments such as city and forest, the GNSS receiver may be unavailable to search sufficient satellites for navigation solution, and costs are high if RTK is applied to all UAVs in a swarm [6].

In order to make full use of the advantages of UAV swarm and obtain reliable and better navigation performance, researchers began to study the cooperative navigation method that integrates relative measurements between UAVs. Compared with single UAV navigation method that merely integrates the local measurement data of the UAV, the CN method can correct its navigation result based on the geometric relationship and observation of the swarm and thus greatly improve the absolute navigation performance [7–9]. Many researchers focused on the optimization algorithm based cooperative positioning technology, created constraint equation set for the geometric relationship between all users, and established corresponding objective functions to solve the positioning result [10–14].
Due to the existence of noise and observation fault such as Non-line-of-sight (NLOS) of wireless ranging signal [15,16], the cooperative positioning problem is often non-convex and cannot be solved globally and optimally. Thereby, convex optimization technologies such as Semi-definite Programming and Cone Programming are required to relax the cooperative estimation problem [14]. The disadvantage of optimization algorithm based cooperative positioning technology is that under worse observation conditions, its performance worsens rapidly, and on-board Inertial Measurement Unit (IMU) and other navigation sensors cannot be fully used for measurement.

A factor graph provides an effective framework for dealing with cooperative navigation problems and is used for describing the linkage between the navigation state and observation at each vehicle of the cooperative network. To pass messages between the nodes in the factor graph, many belief propagation (BP) methods have been proposed [17–21]. Ref. [17] proposed a sum-product algorithm for wireless network (SPAWN), which realized high-precision indoor positioning on the basis of factor graph and message passing. Ref. [21] put forward a factor graph and message passing based H-SPAWN algorithm that can integrate GNSS information and peer-to-peer information to realize cooperative positioning. Ref. [22] provided a distributed positioning method based on sequential particle-based sum-product algorithm (SPSPA), which has the ability to perform online inference in factor graph with continuous variables and nonlinear local functions. Even when obtaining a good effect, these methods require a large number of particles to describe cooperative messages and have a large calculation load, making them hard to guarantee the real-time navigation performance of UAV swarm in the time of formation flight. In order to reduce the computation load of BP algorithm, Ref. [23] offered a cooperative positioning framework combining relative position estimation and optimization, which transforms a multi-dimensional BP problem into an one-dimensional problem, reduces the computation load, and can integrate Ultra-wideband (UWB), GNSS and Inertial Navigation System (INS) information to obtain precise navigation results. However, this framework requires that each UAV in the swarm can conduct GNSS calculation. Ref. [24] presents a cooperative positioning method in which the posteriori distribution of marginalized circulation factor graph is approximated by using sigma point, but the movement of the user is described using a motion model, regardless of the observation information beyond the measurement range. Hence, this method is suitable for indoor application instead of UAV flight scene. It is also unavailable to provide continuous and comprehensive navigation parameters.

In this paper, a sigma point belief propagation-based (SPBP) new CN method is proposed. In this method, sigma point filter integrates local observations from INS, GNSS, barometers, and so on; then the filter’s state vector is augmented and the cooperative measurements are updated iteratively to realize the message passing between UAVs, and further improve the absolute navigation performance of UAV swarm. Compared with the traditional BP algorithm, this paper uses sigma points to replace the particles obtained by Monte Carlo method and constructs the state equation of the cooperative navigation system based on INS, so as to replace the state equation based on motion model. Most importantly, sigma point filter and BP algorithm are combined and expanded to 3D UAV flight urban scene; moreover, IMU, GNSS, and other local multi-source navigation information are incorporated into the CN framework, improving the performance and efficiency of the CN algorithm.

The structure of this paper is as follows: Section 2 provides the preliminaries of sigma point filter and describes the CN problem based on factor graph and BP; Section 3 proposes an SPBP based CN method; the simulations are described in Section 4; and flight tests are described in Section 5; Section 6 discusses the research implications and limitations of the proposed method; and the final conclusion is presented in Section 7.
2. Preliminaries and Description of Problem

2.1. Preliminary on Sigma Point Filter

The basic idea of sigma point filter (SP filter, also known as unscented Kalman filter) is to approximate to a nonlinear density function by using a certain number of sample points and transfer the state of a real system while capturing accurate mean and variance of the state [25]. Therefore, the selection of sigma points is crucial for sigma point filter and is realized by Unscented Transformation [26]. It is assumed that an N-dimensional random vector $X$ is transformed into an $M$-dimensional random vector $Y$ after subjecting to $h(\cdot)$ nonlinear transformation, namely:

$$Y = h(X)$$  \hspace{1cm} (1)

Given the mean $\bar{X}$ and variance $P_{XX}$ of $X$, the $(2N + 1)$ sigma points of $X$ can be reproduced according to $X$:

1. $\chi^{(0)} = \bar{X}$
2. $\chi^{(i)} = \bar{X} + \left(\sqrt{(N+\lambda)P_{XX}}\right)_{(i)}$ for $i = 1, 2, \ldots, N$
3. $\chi^{(i)} = \bar{X} - \left(\sqrt{(N+\lambda)P_{XX}}\right)_{(i-N)}$ for $i = N + 1, N + 2, \ldots, 2N$

where $\left(\sqrt{(N+\lambda)P_{XX}}\right)_{(i)}$ represents the $i$-th column of the square root of the lower triangular decomposition of matrix $\sqrt{(N+\lambda)P_{XX}}$; $\lambda$ determines the distance between the sampling point and the mean. The sigma point produced by nonlinear transformation can be calculated as follows:

$$Y^{(i)} = h\left(\chi^{(i)}\right) \quad i = 0, 1, 2, \ldots, 2N$$  \hspace{1cm} (2)

Hence, the mean $\bar{Y}$, variance $P_{YY}$, and the covariance $P_{XY}$ of $X$ and $Y$ can be approximated as:

1. $\bar{Y} \approx \sum_{i=0}^{2N} W^{(m)}_i Y^{(i)}$  \hspace{1cm} (3)
2. $P_{YY} = \sum_{i=0}^{2N} W^{(c)}_i \left[Y^{(i)} - \bar{Y}\right]\left[Y^{(i)} - \bar{Y}\right]^T$  \hspace{1cm} (4)
3. $P_{XY} = \sum_{i=0}^{2N} W^{(c)}_i \left[X^{(i)} - \bar{X}\right]\left[Y^{(i)} - \bar{Y}\right]^T$  \hspace{1cm} (5)

where $W^{(m)}_i$ and $W^{(c)}_i$ are the corresponding weights, which can be calculated as:

1. $W^{(m)}_i = W^{(c)}_i = \frac{1}{2(N+\lambda)}$, for $i = 1, 2, \ldots, 2N$
2. $W^{(m)}_0 = \frac{\lambda}{N+\lambda}$, $W^{(c)}_0 = \frac{\lambda}{N+\lambda} + (1 - \alpha^2 + \beta)$

where $\alpha$ and $\beta$ are constants and nonnegative.

Considering that the variable $X$ and observation $Z$ satisfy $Z = Y + \xi = h(X) + \xi$ at time $k$, where $\xi$ is the additive measurement noise with variance of $R$, the posteriori probability density function of $X$ can be obtained based on $Z$:

$$p(X|Z) \propto p(Z|X)p(X)$$  \hspace{1cm} (6)

The specific steps to realize the posterior probability density estimation of Equation (8) through SP filter are as follows [26]:

Step 1: Sampling sigma points as per Equation (2) at time $k - 1$;
Step 2: Calculating the one-step predicted value at time $k$:

$$
X_{k|k-1}^{(i)} = f(X^{(i)}, u_{k-1}) \quad i = 0, 1, 2, \ldots, 2N
$$

where $f(\cdot)$ is the one-step prediction equation of the system, that is, the discrete form of the system state equation; $u_{k-1}$ is the input of the system, and $X_{k|k-1}^{(i)}$ is the one-step predicted state.$^1$

Step 3: Computing the observed value $Y_{k|k-1}^{(i)}$ and predicted value $Y_{k|k-1}$ as per Equations (3) and (4), and the corresponding covariance matrices $P_{YY_{k|k-1}}$ and $P_{XY_{k|k-1}}$ are calculated according to Equations (5) and (6);

Step 4: Calculating the posterior probability density estimation:

$$
K_k = P_{XY_{k|k-1}} \left( P_{YY_{k|k-1}} + R \right)^{-1}
$$

$$
\hat{X}_k = \hat{X}_{k|k-1} + K_k (Z_k - Y_{k|k-1})
$$

$$
P_{XX_k} = P_{XX_{k|k-1}} - K_k P_{YY_{k|k-1}} K_k^{-1}
$$

As revealed in the above process of sigma point filter, the number of sigma points depends on the number of dimensions of $X$, namely the former increases with the increase in the latter.

2.2. Belief Propagation for UAV Swarm

Figure 1 shows a flight scene of low-cost micro UAV swarm. In urban areas, the navigation system of UAV may not receive enough number of satellite signals for positioning. For this reason, all UAVs are equipped with GNSS receiver as well as some conventional onboard navigation devices, such as INS and barometer, and are also mounted with UWB to provide the ranging information between UAVs. To meet the demand for CN, it is also essential to provide a wireless communication network for exchanging CN information.

![Figure 1. Cooperative navigation for UAV swarm in urban scenario.](image)
Provided that all UAVs in the swarm belong to a set $U$, the navigation state of UAV $m$ at time $k$ is defined as $X^k_m$, the satellite observation and barometer output at time $k$ are defined as $Z^{k,\text{gnss}}_m$ and $Z^{k,\text{baro}}_m$, respectively; all the IMU measurements between $k-1$ and $k$ are defined as $Z^{k-1,\text{imu}}_m$; and all UWB observations at time $k$ are defined as:

$$Z^{k,\text{uwb}}_m = \{ r^{k,\text{uwb}}_{m,n} | n \in U_m \}$$  \hspace{1cm} (15)

where $U_m$ is the set of all UAVs that have UWB ranging links with UAV $m$. The navigation state set and observation set of all UAVs in the swarm are defined as:

$$X^k = \{ X^k_m | m \in U \}$$  \hspace{1cm} (16)

$$Z^k = \{ Z^k_m | m \in U \} = \{ Z^{k-1,\text{imu}}_m, Z^{k,\text{gnss}}_m, Z^{k,\text{baro}}_m, Z^{k,\text{uwb}}_m | m \in U \}$$  \hspace{1cm} (17)

Based on the above definitions, the joint posterior probability distribution function of the navigation state of the UAV swarm is given by:

$$p(X^k | Z^k, X^{0:k-1})$$  \hspace{1cm} (18)

According to the flight characteristics and sensor characteristics of UAV swarm, the following two conditions can be assumed [23]: a. The navigation state of each UAV at time $k$ is only related to time $k-1$, and obeys the standard Markov assumption; b. The UWB, GNSS and barometric measurements obtained by UAV are independent and only related to the state at the present time. Based on these two assumptions, the joint posterior probability distribution function can be factorized in terms of a priori information and individual process and measurement models. This factorization can be written as

$$p(X^k | Z^k, X^{0:k-1}) \propto p(X^0) \prod_{t=1}^k \left[ p(X^t | Z^{t-1,\text{imu}}, X^{t-1}) p(Z^{t,\text{uwb}} | X^t) \right]$$

$$= p(X^0) \prod_{t=1}^k \left[ \prod_{m \in U} p(X^t_m | Z^{t-1,\text{imu}}_m, X^{t-1}_m) \prod_{m \in U} p(Z^{t,\text{gnss}}_m | X^t_m) \prod_{m \in U} p(Z^{t,\text{baro}}_m | X^t_m) \prod_{r_{m,n} \in \mathcal{D}} p(r^{t,\text{uwb}}_{m,n} | X^t_m, X^t_n) \right]$$  \hspace{1cm} (19)

where $p(X^0)$ represents the priori factor of the navigation state. This factorization can be described by the factor graph composed of factor nodes and variable nodes as presented in Figure 2, where factor node $t^{k-1,\text{imu}}_m = p(X^t_m | Z^{k-1,\text{imu}}_m, X^{k-1}_m)$ represents the motion of UAV obtained on the basis of IMU output; $t^{k,\text{gnss}}_m = p(Z^{k,\text{gnss}}_m | X^k_m)$ and $t^{k,\text{baro}}_m = p(Z^{k,\text{baro}}_m | X^k_m)$ are respectively the functions between UAV $m$ and GNSS data, and between UAV $m$ and barometric height; $t^{k,\text{uwb}}_{m,n} = p(r^{k,\text{uwb}}_{m,n} | X^k_m, X^k_n)$ represents the function of mutual ranging between UAV $m$ and UAV $n$. 


Based on Figure 2 and Equation (19), the approximated marginal posteriori probability density function $b^{(r)}(X_m^k) \approx p(X_m^k \mid Z_m^k)$ of variable node $X_m^k$ is called as the belief, which can be calculated by iteration of BP message passing on the factor graph. At the $\tau$-th iteration at time $k$, the belief of $X_m^k$ can be obtained as follows:

$$b^{(\tau)}(X_m^k) \propto q_{k-1,imu}^{X_m^k} \to X_m^k \cdot q_{k,\text{gnss}}^{X_m^k} \to X_m^k \prod_{n \in U_m} q_{l_n,\text{nwb}}^{X_m^k} \to X_m^k$$  \quad (20)

The forms of messages involved in the above equation are as follows:

1. The prediction message passed from factor node $l_m^{k,imu}$ to variable node $X_m^k$

$$q_{k-1,imu}^{X_m^k} \to X_m^k = l_m^{k-1,imu}(X_m^{k-1})$$  \quad (21)

2. The GNSS correction message passed from factor node $l_m^{k,\text{gnss}}$ to variable node $X_m^k$

$$q_{l_m,\text{gnss}}^{X_m^k} \to X_m^k = l_m^{k,\text{gnss}}$$  \quad (22)

3. The height correction message passed from barometer factor $l_m^{k,\text{nwb}}$ to variable node $X_m^k$

$$q_{l_m,\text{nwb}}^{X_m^k} \to X_m^k = l_m^{k,\text{nwb}}$$  \quad (23)

4. The cooperative correction message passed from UWB factor $l_{n,m}^{k,\text{uwb}}$ to variable node $X_m^k$

$$q_{l_{n,m}^{k,\text{uwb}}}(X_m^k) = \int l_{nm}^{k,\text{uwb}}(X_m^k, X_n^m) \ dt X_n^m$$  \quad (24)

where $b^{(\tau)}(X_m^k)$ is the belief passed by UAV $n$ to factor node $l_{n,m}^{k,\text{uwb}}$ at the $\tau$-th iteration:

$$b^{(\tau)}(X_m^k) = q_{k-1,imu}^{X_m^k} \to X_m^k \cdot q_{k,\text{gnss}}^{X_m^k} \to X_m^k \prod_{m' \in U_n \setminus \{m\}} q_{l_{nm'}^{k,\text{uwb}}}(X_m^k) \to X_m^k$$  \quad (25)

where $m' \in U_n \setminus \{m\}$ represents the set of all UAVs that have ranging links with UAV $n$, except UAV $m$. For the SPAWN message passing scheme, $b^{(\tau)}(X_m^k)$ can be directly replaced with $b^{(\tau)}(X_n^k)$ without recalculation [17].
3. Sigma Point Belief Propagation

Section 2.1 shows an SP filter based method for posteriori probability estimation of variables. According to a comparison among Equations (8), (19) and (20), the expression of belief is similar to Equation (8), and the belief of navigation state can also be calculated by SP. Using a mean vector and covariance matrix to approximate the belief of each UAV’s navigation state is conducive to reducing communication cost and calculation load, and low-complexity approximation to the traditional BP algorithm. This section will expound the SPBP method in detail.

3.1. Navigation State Model and Time Update

Taking Earth-fixed frame as the navigation coordinate frame, the motion of UAV from $k - 1$ to $k$ is modeled. Based on the measurements of IMU, the time update of the motion state of UAV is obtained as follows [27]:

$$\begin{align*}
Q &= 0.5Q \circ (\tilde{\omega} - \omega_{ie} - \omega_{en} - \omega_r - \omega_e) \\
\dot{v} &= C_n^b(\tilde{a} - \nabla_r - (2\omega_{ie} + \omega_{en}) \times v) + g \\
p &= v
\end{align*}$$

(26)

where $Q$ is the attitude quaternion, including the one-dimensional scalar $Q_0$ and the three-dimensional vectors $Q_1$, $Q_2$ and $Q_3$, $\circ$ represents the quaternion multiplication, $v$ is the three-dimensional velocity, $p$ is the three-dimensional position, $\tilde{\omega}$ is the measurement of the gyroscope, $\omega_{ie}$ is the angular velocity of the Earth’s rotation, $\omega_{en}$ is the rotational angular velocity of the carrier relative to the Earth; $\omega_r$ is the random walk error of the gyroscope, $\omega_e$ is the measurement noise of the gyroscope, $C_n^b$ is the rotation matrix from body frame to navigation frame, $\tilde{a}$ is the accelerometer measurement, $\nabla_r$ is the random walk error of the accelerometer, $\times$ represents the cross multiplication of vectors, $G$ is the gravity vector, and $r$ is the position vector with respect to the Earth Centered Earth fixed frame. The random walk error update equation of the gyroscope and accelerometer is:

$$\begin{align*}
\dot{\omega}_r &= \omega_n \\
\nabla_r &= \nabla_n
\end{align*}$$

(27)

where $\omega_n$ and $\nabla_n$ are the random walk drive noises of the gyroscope and accelerometer, respectively, which are usually supposed to be zero mean white noises. The covariance of noise can be obtained by ALLAN variance method or other IMU error analysis methods.

Since the scalar part of the normalized quaternion can be obtained from the vector part, in order to reduce the dimension of the SP filter, the initial state vector is created on the basis of vectors $Q_1$, $Q_2$ and $Q_3$, as well as the position vector, velocity vector, gyro random walk error $\omega_r$ and accelerometer random walk error $\nabla_r$:

$$x = \begin{bmatrix} Q_1 & Q_2 & Q_3 & v^T & p^T \end{bmatrix}^T (\omega_r)^T (\nabla_r)^T$$

(28)

The system noise is:

$$W = \begin{bmatrix} (\omega_n)^T & (\omega_r)^T & (\nabla_n)^T \end{bmatrix}^T$$

(29)

When the sigma point filter is updating the time, the system noise should be augmented to the system state, and the augmented system state vector is $X = \begin{bmatrix} x^T & W^T \end{bmatrix}^T$. By combining and discretizing Equations (26) and (27), the one-step prediction equation of the navigation state can be obtained, that is, Equation (9). Then, the attitude, velocity and position represented by each sigma point can be updated in combination with the input of IMU as per Equations (9)–(11). It should be noted that for observations from satellites, barometers and UWB, if the observation error does not belong to additive white noise (e.g., error parameters such as the receiver clock error of equivalent pseudo range observa-
tion), the corresponding error vector should also be augmented to the system state vector. Apparently, the IMU measurement based time update of sigma point filter corresponds to the passing of message $q_{k-1,imu} \rightarrow X_{m}^{k-1}$ in BP. At time $k-1$, the belief message passed by variable node $X_{m}^{k-1}$ to IMU factor $f_{k-1,imu}$ is approximated to the mean $\hat{X}_{m}^{k-1}$ and variance $P_{XX_{m}}^{k-1}$ of the state.

3.2. Measurement Model and Measurement Update
3.2.1. Local Measurement Model and Measurement Update

In this paper, two measurements, GNSS information and barometric altitude, are considered in addition to IMU. Regarding whether the satellites searched by GNSS receiver can be positioned directly or not, the GNSS measurement can be divided into position measurement and pseudo-range measurement. The position measurement can be modeled as follows [28]:

$$p_{m}^{G} = p_{m}^{G} + \nu_{m}^{G}$$ (30)

where $\nu_{m}^{G}$ is the GNSS position observation noise of UAV $m$ and can be modeled as a white noise with covariance of $R_{m}^{G}$. For position measurement, the SP filter does not need to augment the system state.

The pseudo range measurement can be modeled as [29]:

$$\hat{\rho}_{i}^{S_{m}} = \rho_{i}^{S_{m}} + \delta t_{m}^{\rho} + \zeta_{i}^{S_{m}} + \Delta \rho_{i}^{S_{m}} + \nu_{i}^{S_{m}}$$ (31)

where $\rho_{i}^{S_{m}}$ is the true distance between UAV $m$ and satellite $S_{i}$; $\delta t_{m}^{\rho}$ is the distance offset corresponding to the equivalent clock error of the receiver; $\zeta_{i}^{S_{m}}$ is the error offset caused by the ionosphere and troposphere; $\Delta \rho_{i}^{S_{m}}$ is the error caused by multipath or NLOS; and $\nu_{i}^{S_{m}}$ is the receiver noise with covariance of receiver noise. The errors caused by ionosphere and troposphere can be eliminated by using the corresponding correction model or the information broadcasted by Satellite-Based Augmentation System (SBAS); the error caused by multipath or NLOS can be detected in some integrity methods [30–32]. When the detected error caused by multipath or NLOS is large, the pseudo range measurement can be deleted directly. For pseudo range measurement, the equivalent receiver clock error $\delta t_{m}^{\rho}$ should be augmented to system state. The time update equation of $\delta t_{m}^{\rho}$ is shown below:

$$\delta t_{m}^{\rho} = \delta t_{m}^{\rho} + \nu_{m}^{\rho}$$ (32)

where $\nu_{m}^{\rho}$ is the equivalent clock error deviation noise.

The barometric height can be modeled as:

$$h_{m}^{\text{baro}} = h_{m} + \nu_{m}^{\text{baro}}$$ (33)

where $h_{m}$ is the true height of UAV $m$; and $\nu_{m}^{\text{baro}}$ is the barometer measurement noise with variance of $R_{m}^{\text{baro}}$.

After the local measurements are modeled, the system state should be augmented again as per the noise form when there is non-additive noise in the measurement model and the dimension of the augmented state is $N$. The error augmented to system state vector in local measurement also requires time update. In face of upcoming measurement, the measurement can be updated as per Equations (3)–(6) and (12)–(14). $b^{(0)} (X_{m}^{k}) \propto q_{k-1,imu} \rightarrow X_{m}^{k} \rightarrow q_{k,\text{gnss}} \rightarrow X_{m}^{k} \rightarrow q_{k,\text{baro}} \rightarrow X_{m}^{k}$ is the belief corresponding to the mean $X_{m}^{k}$ and covariance $P_{XX_{m}}^{k}$ of the local measurement state updated at time $k$. 
3.2.2. Cooperative Measurement Model and Measurement Update

The UWB measurement within the visual range based on Time of Flight (TOF) can be modeled as [33,34]:

\[
\tilde{d}_{m,n} = \|p_m - p_n\| + u_{uwb}^{m,n} \tag{34}
\]

where \(\tilde{d}_{m,n}\) is the UWB ranging value between UAV \(m\) and UAV \(n\); \(\| \cdot \|\) represents the Euclidean distance; and \(u_{uwb}^{m,n}\) is the ranging noise with variance of \(R_{uwb}\).

In order to effectively integrate the cooperative measurement of UWB, the position vector of adjacent UAV should be augmented to the system state. After the augmentation, the state vector of SP filter for UAV \(m\) is:

\[
X_m^0 = \left( X_m^T \left( p_{U_m} \right) \right)^T \tag{35}
\]

where \(p_{U_m} = \left[ \left( p_{n_1} \right)^T \left( p_{n_2} \right)^T \ldots \left( p_{n_{\nu_{\text{U}}}} \right)^T \right]^T\) is a vector composed of the positions of all UAVs that have ranging links with UAV \(m\) after the augmentation, the dimension of the state is \(N^u\). Assuming that the mean and covariance of the state variable corresponding to the message passed by UAV \(n\) to factor node \(r_{U_m}^{k,n}\) after the \(\tau\)-th iteration at time \(k\) are \(X_n^{(\tau)k}\) and \(P_{XX}^{(\tau)k}\), respectively, the positions \(p_n\) and \(P_{PP}^{(\tau)k}\) can be obtained from \(X_n\) and \(P_{XX}^{(\tau)k}\). Hence, the mean and covariance of the augmented state vector of UAV \(m\) are as presented below:

\[
\begin{align*}
\hat{X}_{XX}^{(\tau)k} &\triangleq \begin{bmatrix} \hat{(\tau)k}_1 \ldots \hat{(\tau)k}_{\nu_{\text{U}}} \end{bmatrix}^T \\
P_{XX}^{(\tau)k} &\triangleq \text{diag} \left( P_{XX}^{(\tau)k}, P_{PP}^{(\tau)k}, \ldots, P_{PP}^{(\tau)k}_{\nu_{\text{U}}} \right)
\end{align*} \tag{36}
\]

where diag(\cdot) represents the block diagonal matrix whose diagonal blocks are the listed matrices. A sigma point can be regenerated as per the mean and covariance of Equation (36), followed by measurement update and completion of the \((\tau + 1)\)-th iteration, obtaining the belief \(b^{(\tau + 1)} \left( X_m^k \right)\). Noteworthy, the position vector is augmented without time update.

3.3. Algorithm Description

Taking the navigation state of UAV \(m\) as an example, the proposed method is described in detail as shown in Algorithm 1 and Figure 3.

**Step 1** (Algorithm 1: 1) is to initialize the state vector \(X_m^0\) of SP filter, and set the initial mean \(X_m\) and initial covariance \(P_{XX}^0\), which are the bases for the time recursion of the SP filter.

**Step 2** (Algorithm 1: 3–6) is to sample sigma points as per the mean \(X_m^k\) and covariance \(P_{XX}^{k-1}\) of the state at the previous time, and complete time update of the system based on the IMU measurements, obtain the mean \(X_m^k\) and covariance \(P_{XX}^{k-1}\) of the state quantity.

**Step 3** (Algorithm 1: 7–9) is to complete local measurement update as per GNSS measurements, barometric height and other local measurements, obtain the mean \(X_m^k\) and covariance \(P_{XX}^{k-1}\) of the state. This mean and covariance correspond to belief \(b^{(0)} \left( X_m^k \right) \propto q_{Xm}^{k-1,0} \rightarrow X_m^k q_{Xm}^{k-1,0} \rightarrow X_m^k q_{Xm}^{k-1,0} \rightarrow X_m^k q_{Xm}^{k-1,0} \rightarrow X_m^k q_{Xm}^{k-1,0} \cdot \)
Algorithm 1: The Detailed Recursion Process of Navigation State Based on SPBP (Example of UAV m)

1: INPUT: Initial system state $X^0_m$, including the initial mean $X^0$ and the initial covariance $P^0_{XXm}$
2: Measurements of IMU, GNSS, barometer and UWB
3: OUTPUT: Navigation results of UAV $m$, including position, velocity and attitude
4: CALCULATE:
5: for time step $t = 1$ to $T$ do
6: for UAV $m \in U$ in parallel do
7: Sample $2N + 1$ sigma points according to Equation (2)
8: Time update as per Equations (26), (27) and (9)–(11), calculate the message passed by IMU factor $f_{m_{imu}}^t$ to variable node $X^t_{m-1}$ at time $k$, and obtain the mean $X^t_{m}$ and covariance $P^t_{XXm}$ of the state
9: end parallel
10: for UAV $m \in U$ in parallel do
11: while ($\|p_m^{(t+1)}\|_2^2 - P_m^{(t+1)} \parallel / \|p_m^{(t)}\|_2^2 < \alpha$) do
12: Propagate the mean $p_m$ and covariance $p_m^{(t+1)}$ of the position corresponding to message $b^{(t)}(X^t_m)$
13: Receive the mean and covariance matrix of the position sent by adjacent UAV
14: Augment the state as per Equation (35) and obtain the mean $X^t_m$ and covariance $P^t_{XXm}$ of the new state
15: Resample $2N^\alpha + 1$ sigma points according to Equation (2)
16: Complete the cooperative measurement update of UWB observation as per Equations (3)–(6) and (12)–(14), and obtain the mean $X^t_m$ and covariance $P^t_{XXm}$ of the state quantity corresponding to belief $b^{(t)}(X^t_m)$
17: end while
18: end parallel
19: for UAV $m \in U$ in parallel do
20: Extract the navigation states from the final iteration results $X^t_m$ and $P^t_{XXm}$, and output the navigation results
21: end parallel
22: end for

Step 4 (Algorithm 1: 10–17) is to iterate the cooperative messages. The iterative loop is divided into 4 stages:

1. Propagate the mean $p_m^{(t)}$ and covariance $P_m^{(t)}$ of position corresponding to the belief $b^{(t)}(X^t_m)$, receive the mean and covariance of the position of adjacent UAV, and then augment the state to obtain the mean $X^t_{XXm}$ and covariance $P^t_{XXm}$ of a new state;
2. Resample $2N^\alpha + 1$ sigma points;
3. Update the cooperative measurement and obtaining the mean $X^t_{m}$ and covariance $P^t_{XXm}$ of the state quantity corresponding to belief $b^{(t+1)}(X^t_m)$;
Judge whether the convergence condition \( \| p_m^{(\tau+1)} t - p_m^{(\tau)} t \| / \| p_m^{(\tau+1)} t \| < \alpha \) is met or not; if yes, exiting the loop; if not, skipping to stage 1). \( \alpha \) can be selected as per the empirical value.

Step 5 (Algorithm 1: 19–21) is to extract and output the navigation results \( \hat{X}_m^t \) and \( P_{XX}^t \), and then go to Step 2 to start the next epoch.

3.4. Complexity Analysis of SPBP

For the SPBP method proposed in this paper, the main calculation load is to calculate the square root of the covariance matrix when selecting sigma points as per Equation (2). An effective calculation can be realized by Cholesky decomposition [35], with complexity being cube of the dimension of the state and also cube of particle number \( N_{spbp} \).

Within an epoch, the computation load of this method is \( O \left( (N)^3 + (N^a + 3M_{link})^3 I_{sp} \right) \), where \( M_{link} \) is the number of adjacent UAVs and \( I_{sp} \) is the number of iterations. In contrast, for traditional NBP method, its computation complexity is linear in terms of \( M_{link} \) and quadratic in terms of particle number. Hence, the computation complexity of traditional NBP method is \( O \left( I_{nbp} (N_{nbp})^2 M_{link} \right) \) regarding the state estimation requiring \( I_{nbp} \) iterations and \( N_{nbp} \) particles. However, the particle number for SPBP is much less than that for NBP in general. Therefore, SPBP method has far higher calculation efficiency than NBP method.

In terms of communication traffic, the SPBP method takes the mean \( \hat{p}_m^{(\tau)} t \) and covariance \( P_{pp}^{(\tau)} t \) of the estimated position as the belief of the position. Compared with the NBP

Figure 3. Procedure for the proposed SPBP algorithm.
method that requires transferring sub-particles, the SPBP method obviously has a small communication load.

4. Simulations
4.1. Simulation Configuration

Based on a flight setting in urban scene, a UAV swarm containing 12 UAVs is simulated to verify the proposed SPBP method. Figures 4 and 5 display the initial position and horizontal trajectories of the UAV swarm, respectively. It is assumed that every UAV can interact with the UAVs within the working distance of the UWB transceiver, 60 m. In order to fully evaluate the performance of this method, all the results shown in this section are based on 200 Monte Carlo simulation experiments where each simulation takes 1000 s.

![Figure 4. The initial positions of the UAV swarm.](image)

![Figure 5. Two-dimensional trajectories of the UAV swarm.](image)

The parameters used for sensor configuration and simulation are as listed in Table 1 [36]. Each UAV is equipped with low-cost IMU, barometer and GNSS receiver, and can conduct inter-UAV ranging via a UWB transceiver. It is set that all UAVs realize the transmission of cooperative information through broadcasting, and UAVs that receive cooperative information and are within the ranging range can use this information for cooperation. Two UAVs in the swarm can obtain the position observation of GNSS directly, the remaining UAVs at a
flight height above 30 m can receive pseudo-range observations of three satellites, while that at a flight height below 30 m cannot receive the pseudo-range observations.

Table 1. Sensor configuration and parameters.

| Sensor   | Parameter                                | Value         |
|----------|------------------------------------------|---------------|
| Gyro     | Random walk drive noise                  | $3^\circ/h$   |
|          | White noise error                         | $3^\circ/h$   |
|          | Output frequency                          | $50$ Hz       |
| Accelerometer | Random walk drive noise                  | $10^{-4}$ g   |
|          | Output frequency                          | $50$ Hz       |
| Barometer | Gaussian noise standard dev.              | $0.5$ m       |
|          | Output frequency                          | $5$ Hz        |
| GNSS     | Position noise standard dev.              | $5$ m         |
|          | Pseudo range noise standard dev.          | $5$ m         |
|          | Initial equivalent clock error            | $60$ m        |
|          | Equivalent clock drift noise              | $10$ m        |
|          | Output frequency                          | $5$ Hz        |
| UWB      | Gaussian noise standard dev.              | $0.15$ m      |
|          | Output frequency                          | $5$ Hz        |

In the simulation, the positioning estimation error is calculated as per the following equation:

$$ E = \sqrt{Err_E^2 + Err_N^2 + Err_U^2} $$

(37)

where $Err_E$, $Err_N$ and $Err_U$ are the estimation errors on the east, north and upward directions respectively.

4.2. Simulation Results for Cooperative Navigation

To compare the proposed SPBP with other methods, this section simulates several cooperative positioning algorithms based on the same dataset, including H-SPAWN [21], Cooperative Positioning based Extended Kalman Filter (CP-EKF) [37], Multidimensional Scaling (MDS) [38] and Non-linear Regression (NLR) [39] methods. The positioning results are illustrated in Figures 6 and 7. The detail of each algorithm is presented as follows:

Figure 6. Positioning error CDF comparisons for all vehicles: H-SPAWN (particle number = 2500), H-SPAWN (particle number = 1500), NLR, CP-EKF, MDS, SPBP (positioning errors from 200 simulation trials).

Remote Sens. 2022, 14, x FOR PEER REVIEW 15 of 24
Figure 6 presents the simulation results of several cases, where the cumulative distribution functions (CDF) of position errors for all UAVs are used as the performance metrics. Figure 7 provides the example performance of a single UAV without satellite position measurement. NLR is a cooperative positioning method merely depending on the geometric relationship of UAV swarm and its precision is obviously lower than other cooperative methods as shown in the figure. CP-EKF integrates the navigation information of IMU, barometer, GNSS, and adjacent UAVs. However, without iterative update, the indirect cooperative information of non-adjacent UAV is not fully used. Hence, the positioning precision of CP-EKF is higher than that of NLR, but lower than MDS method and other BP-based cooperative methods. The relative positioning process of the MDS filter only utilizes the geometric constraints of the UAV swarm, thus its performance is lower than BP-based methods. Compared with H-SPAWN, the SPBP proposed in this paper

1. H-SPAWN (particle number = 2500): It is short for hybrid sum-product algorithm for wireless network, a typical BP-based positioning algorithm proposed in [21]. Its message passing process is identical with Equations (20)–(24). In case 1, 2500 particles are used.

2. H-SPAWN (particle number = 1000): This algorithm is the same as that stated in 1, but the number of particles used is 1000.

3. CP-EKF: It is a CN method, in which EKF is used for integrating local measurements and other UAV cooperative data. The error of navigation state is designed as the state of filter. The cooperative measurement obtained by inter-UAV ranging is also incorporated into the measurement model. It should be noted that the covariance of noise in the measurement model considers both the UWB ranging error and the collaborator’s position error.

4. NLR: In this method, a nonlinear regression equation set is established on the basis of all distance and position measurements, and the UAV’s position is solved in iterative least square method.

5. MDS: This method is based on multidimensional scaling and the maximum likelihood estimation. The approach is composed of three sub-steps: (a) construction of the distance matrix, (b) relative position estimation, (c) coordinates registration. If the node scale is large, the network can be divided into several local maps for relative positioning respectively.

6. SPBP: The proposed method in this paper.

Figure 6 presents the simulation results of several cases, where the cumulative distribution functions (CDF) of position errors for all UAVs are used as the performance metrics. Figure 7 provides the example performance of a single UAV without satellite position measurement. NLR is a cooperative positioning method merely depending on the geometric relationship of UAV swarm and its precision is obviously lower than other cooperative methods as shown in the figure. CP-EKF integrates the navigation information of IMU, barometer, GNSS, and adjacent UAVs. However, without iterative update, the indirect cooperative information of non-adjacent UAV is not fully used. Hence, the positioning precision of CP-EKF is higher than that of NLR, but lower than MDS method and other BP-based cooperative methods. The relative positioning process of the MDS filter only utilizes the geometric constraints of the UAV swarm, thus its performance is lower than BP-based methods. Compared with H-SPAWN, the SPBP proposed in this paper
is a method for approximating the distribution of state based on finite sampling points (i.e., sigma points). The difference between H-SPAWN and SPBP is similar to the difference between particle filter and UKF. Therefore, when there is no strong nonlinearity in the measurement, such as NLOS, multipath, and other errors, the CN model is basically in line with the linear mixed problem model, and the SPBP can realize similar performance to H-SPAWN as well as lower calculation complexity. As for the SPBP method proposed in this paper, its positioning precision is close to that of H-SPAWN (particle number = 2500) and superior to that of H-SPAWN (particle number = 1500). For instance, the percentiles with positioning error not greater than 2 m are lower than 94.9% for SPBP, close to 92.3% for H-SPAWN (particle number = 2500), and 82.1%, 11.7%, 45.7%, and 33.4% respectively for H-SPAWN (particle number = 1500), NLR, MDS and CP-EKF methods.

To further analyze the complexity of the proposed method, the ‘tic’ and ‘toc’ functions of MATLAB are used to record the running time of the code, which can roughly represent the computation load. As can be seen from Figure 8, which presents the efficiency comparison among all the above methods, NLR and CP-EKF do not need iterative passing of message, hence the minimum computation load. The computational efficiency of the MDS filter is similar to SPBP but with large fluctuation, mainly because the iterative process of MDS filter is largely dependent on prior information, which is affected by measurement biases. For H-SPAWN (particle number = 2500), the processing time at each epoch is more than 30 times that of SPBP. This means that with similar performance, SPBP has much lower calculation complexity than H-SPAWN.

![Figure 8](image-url)  
**Figure 8.** Example of processing time at each computation epoch: H-SPAWN (particle number = 2500), H-SPAWN (particle number = 1500), NLR, CP-EKF, MDS, SPBP.

5. Flight Tests

5.1. System Configuration

Figures 9 and 10 show the UAV platform for the test and the configuration and signal flow of the navigation equipment, respectively. Each UAV is equipped with Ublox M8 GPS module, Forsense IMU6132 module, BMP280 atmospheric sensor and LinkTrack UWB module. The sensor data are collected, packed and sent to Raspberry Pi4 by STM32H7 processor. Raspberry Pi4 is for storing and processing data and sending control commands. While ranging between UAVs, UWB as ranging and communication module may also be used by STM32H7 to send its position and other information to be passed to other UAVs. Each UWB module can range and communicate with the other four based on time-division mechanism.
5.2. Test Results

The effectiveness of the proposed method was verified by an outdoor automatic flight test on a UAV swarm on the drill ground. This swarm is composed of five quadrotor UAVs (Figure 11). To evaluate the navigation performance, each UAV was equipped with on-board RTK equipment to provide the true value of the reference position. Figure 12 displays the flight scene, and the 3D and 2D flight trajectories of UAV swarm. As shown, after taking off, the UAV swarm flew in an L-shaped loop for 6 cycles before landing.
The flight system of UAV swarm. (a) RTK Base Station. (b) Wireless Access Point. (c) UAV Swarm. (d) Ground Station.

Remote Sens. 2022, 14, x FOR PEER REVIEW 18 of 24

Figure 11. The flight system of UAV swarm.

(a)

(b)

(c)

Figure 12. The flight scene and trajectory of UAV swarm. (a) Flight scene of UAV swarm. (b) Three-dimensional flight trajectory. (c) Two-dimensional flight trajectory.

Figure 12. The flight scene and trajectory of UAV swarm. (a) Flight scene of UAV swarm. (b) Three-dimensional flight trajectory. (c) Two-dimensional flight trajectory.
The entire flight process is divided into three stages: take-off, flight in loop (cooperation stage), and landing. It is set as follows: in the first and third stages, the navigations of all UAVs are realized by the combination of INS/GPS; after entering the second stage, UAV 1 and UAV 2 can obtain the satellite position measurements continuously, while other UAVs reject the input of satellite measurements and position cooperatively. To prove the performance of the proposed method, H-SPAWN was selected to compare with the method proposed in this paper based on actual flight. In the second stage, the horizontal positioning error CDFs of UAV 3, 4 and 5 are presented in Figure 13, and the horizontal positioning error of each UAV is illustrated in Figure 14.

![Figure 13. Positioning error CDF comparisons: H-SPAWN (Particle number = 3000), H-SPAWN (Particle number = 1000), SPBP.](image)

In order to compare the positioning performances of the mentioned algorithms more clearly, Table 2 lists the positioning results of the several algorithms for comparison. As shown, the total positioning precision of SPBP is similar to that of H-SPAWN (particle number = 3000), while the positioning precision of SPBP used for each UAV is superior to that of H-SPAWN (particle number = 1000). In terms of the average processing time at each epoch, the computation load of the proposed SPBP algorithm is 4 times and 20 times lower than those of H-SPAWN (particle number = 1000) and H-SPAWN (particle number = 3000) respectively. The flight test results demonstrate that the SPBP algorithm can significantly improve the calculation efficiency while guaranteeing the performance.

| Method Type                    | RMSE in the Positioning Error (Units: m) | Average Processing Time (Units: s) |
|-------------------------------|-----------------------------------------|-----------------------------------|
|                               | UAV3  | UAV4  | UAV5  | Total  |                                   |
| H-SPAWN (Particles = 3000)    | 0.581 | 0.696 | 0.324 | 0.556  | 0.3745                            |
| H-SPAWN (Particles = 1000)    | 0.764 | 0.875 | 0.478 | 0.725  | 0.0885                            |
| SPBP                          | 0.612 | 0.487 | 0.459 | 0.524  | 0.0175                            |
Figure 14. Positioning error comparisons: H-SPAWN (particle number = 3000), H-SPAWN (particle number = 1000), SPBP. (a) Positioning error of UAV 3. (b) Positioning error of UAV 4. (c) Positioning error of UAV 5.
6. Discussion

This paper proposes a new CN method based on SPBP. In this method, a limited number of sigma points are selected to approximate the posteriori distribution of the navigation state quantity. The calculation load of this method is greatly less than that of traditional nonparametric BP method. Based on onboard multisource navigation sensor, the SPBP method proposed in [24] is expanded to a 3D environment in this paper, the state vectors of the time and measurement updates of sigma point filter are separated, and the state vectors are augmented only when updating the measurement of the cooperative information. Thereby, the messages passed by different UAVs can be received in every time of iteration and the SPBP-based new CN method proposed in this paper is more suitable for the UAV flight scene.

Particularly, when the setting of convergence threshold is small, the covariance matrix of the sigma point filter may be overconfident over repeated iterations, resulting in over-concentrated particles in the next sampling. Although the SPBP method can be used to approximate state distribution on the basis of a few sigma points, it is still needed to satisfy the Gaussian assumption of the system. The estimation performance may be poor when the navigation system is not corrected for a long time or the observation is strongly nonlinear. For example, when NLOS exists in UWB measurement and is not removed, the navigation result may be biased; meanwhile, the state covariance cannot be adjusted accordingly so that the next sampling scope cannot cover the correct position.

In the future, the SPBP algorithm can combine with some performance evaluation methods, such as posteriori Cramer-Rao bound [40,41], to solve the problem of overconfidence in the covariance matrix.

In addition, for the cooperative navigation system, an excellent communication network is indispensable. For the case where the UAVs are closely distributed and the number is small, a wireless communication network can be established in a point-to-point manner, which can be realized through XBEE or UWB, and the cost and complexity are low. If the scale or the distribution range of UAV swarm is large, the situation becomes complex. It is best to realize cooperative information interaction through broadcasting, so that the UAVs that receive cooperative information and are within the UWB working distance can complete cooperative navigation.

7. Conclusions

Against the CN background of low-cost UAV swarm, this paper proposes a distributed CN method that integrates common onboard navigation sensors and cooperative measurement information. In this method, the filtering of sigma point filter is divided into two steps. First, local measurements are integrated. Then, the mean and covariance of state are used to approximate the belief of the navigation state, and the message passing in traditional BP method is implemented by augmentation, resampling and cooperative measurement update of the state, realizing a low-complexity approximation to the traditional message passing scheme.

The simulation results show that in the simulation scene, the positioning performance of the SPBP method proposed in this paper is significantly superior to that of traditional single-step cooperative positioning methods such as CP-EKF and non-Bayesian CN method such as NLR and MDS. With similar performance, the computation load of the proposed SPBP method is much lower than that of the H-SPAWN method. The flight test results also verify the effectiveness of the proposed SPBP method. The future research will continue to study the SPBP and performance evaluation combined CN method, as well as the method for monitoring the integrity of CN information.
Author Contributions: Conceptualization, M.C. and Z.X.; methodology, M.C.; software, M.C. and F.S.; validation, M.C.; formal analysis, M.C.; investigation, M.C.; data curation, M.C. and F.S.; writing—original draft preparation, M.C.; writing—review and editing, M.C., R.W., J.X. and Z.X.; visualization, R.W. and J.X.; supervision, Z.X.; project administration, Z.X.; funding acquisition, Z.X. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by the National Natural Science Foundation of China (Grant No. 61873125, 62073163), National Defense Basic Research Program(JCKY2020606SC009), the Aeronautic Science Foundation of China (Grant No. ASFC-2020Z071052001, Grant No. 202055052003), and the Natural Science Research Start-up Foundation of Recruiting Talents of Nanjing University of Posts and Telecommunications (Grant No. NY221137).

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Not applicable.

Acknowledgments: Throughout the writing of this dissertation I have received a great deal of support and assistance. I would particularly like to acknowledge my teammate, Chenfa Shi, Jie Wang, Mingjiang Ma, for their wonderful collaboration and patient support during the flight test. In addition, I would like to thank my parents for their wise counsel and sympathetic ear. You are always there for me. Finally, I could not have completed this dissertation without the support of my friend, Yiming Ding, who provided stimulating discussions as well as happy distractions to rest my mind outside of my research.

Conflicts of Interest: The authors declare no conflict of interest. The funders had no role in the design of the study; in the collection, analyses, or interpretation of data; in the writing of the manuscript, or in the decision to publish the results.

References

1. Karantanellis, E.; Marinos, V.; Vassilakis, E.; Christaras, B. Object-Based Analysis Using Unmanned Aerial Vehicles (UAVs) for Site-Specific Landslide Assessment. Remote Sens. 2020, 12, 1711. [CrossRef]
2. Zhou, W.; Li, J.; Liu, Z.; Shen, L. Improving multi-target cooperative tracking guidance for UAV swarms using multi-agent reinforcement learning. Chin. J. Aeronaut. 2021. [CrossRef]
3. Zhen, Z.; Chen, Y.; Wen, L.; Han, B. An intelligent cooperative mission planning scheme of UAV swarm in uncertain dynamic environment. Aerosp. Sci. Technol. 2020, 100, 105826. [CrossRef]
4. Wu, T.; Wang, J.; Tian, B. Periodic event-triggered formation control for multi-UAV systems with collision avoidance. Chin. J. Aeronaut. 2021. [CrossRef]
5. Kang, Y.; Park, B.-J.; Cho, A.; Yoo, C.-S.; Kim, Y.; Choi, S.; Koo, S.-O.; Oh, S. A Precision Landing Test on Motion Platform and Shipboard of a Tilt-Rotor UAV Based on RTK-GNSS. Int. J. Aeronaut. Space Sci. 2018, 19, 994–1005. [CrossRef]
6. Xiong, J.; Cheong, J.W.; Xiong, Z.; Dempster, A.G.; Tian, S.; Wang, R. Hybrid Cooperative Positioning for Vehicular Networks. IEEE Trans. Veh. Technol. 2020, 69, 714–727. [CrossRef]
7. Causa, F.; Fasano, G. Improving Navigation in GNSS-Challenging Environments: Multi-UAS Cooperation and Generalized Dilution of Precision. IEEE Trans. Aerosp. Electron. Syst. 2021, 57, 1462–1479. [CrossRef]
8. Shen, J.; Wang, S.; Zhai, Y.; Zhan, X. Cooperative relative navigation for multi-UAV systems by exploiting GNSS and peer-to-peer ranging measurements. IET Radar SonarNavig. 2021, 15, 21–36. [CrossRef]
9. Guo, K.; Li, X.; Xie, L. Ultra-Wideband and Odometry-Based Cooperative Relative Localization with Application to Multi-UAV Formation Control. IEEE Trans. Cybern. 2020, 50, 2590–2603. [CrossRef]
10. Efmataneshnik, M.; Alam, N.; Kealy, A.; Dempster, A.G. A Fast Multidimensional Scaling Filter for Vehicular Cooperative Positioning. J. Navig. 2012, 65, 223–243. [CrossRef]
11. Lee, J.K.; Greiner-Brzezinska, D.A.; Toth, C. Network-based Collaborative Navigation in GPS-Denied Environment. J. Navig. 2012, 65, 445–457. [CrossRef]
12. Srirangarajan, S.; Tewfik, A.H.; Luo, Z.Q. Distributed sensor network localization using SOCP relaxation. IEEE Trans. Wirel. Commun. 2008, 7, 4886–4895. [CrossRef]
13. Lin, L.; So, H.C.; Chan, F.K.W.; Chan, Y.T.; Ho, K.C. A new constrained weighted least squares algorithm for TDOA-based localization. Signal Process. 2013, 93, 2872–2878. [CrossRef]
14. Lui, K.W.K.; Ma, W.K.; So, H.C.; Chan, F.K. Semi-Definite Programming Algorithms for Sensor Network Node Localization With Uncertainties in Anchor Positions and/or Propagation Speed. IEEE Trans. Signal Process. 2009, 57, 752–763. [CrossRef]
15. López-Estrada, F.-R.; Rotondo, D.; Valencia-Palomino, G. A Review of Convex Approaches for Control, Observation and Safety of Linear Parameter Varying and Takagi-Sugeno Systems. Processes 2019, 7, 814. [CrossRef]
16. Vaghefi, R.M.; Buehrer, R.M. Cooperative Localization in NLOS Environments Using Semidefinite Programming. *IEEE Commun. Lett.* 2015, 19, 1382–1385. [CrossRef]

17. Wyse, H.; Lien, J.; Win, M.Z. Cooperative Localization in Wireless Networks. *Proc. IEEE* 2009, 97, 427–450. [CrossRef]

18. Nasiri, H.; Koivunen, V. A Bayesian Algorithm for Distributed Network Localization Using Distance and Direction Data. *IEEE Trans. Signal Inf. Process. Over Netw.* 2019, 5, 290–304. [CrossRef]

19. Li, B.; Wu, N.; Wang, H.; Tseng, P.-H.; Kuang, J. Gaussian message passing-based cooperative localization on factor graph in wireless networks. *Signal Process.* 2015, 111, 1–12. [CrossRef]

20. Yuan, W.; Wu, N.; Wang, H.; Li, B.; Kuang, J. Joint synchronization and localization based on Gaussian belief propagation in sensor networks. In Proceedings of the 2015 IEEE International Conference on Communications (ICC), London, UK, 8–12 June 2015; pp. 6646–6651. [CrossRef]

21. Caceres, M.A.; Penna, F.; Wyse, H.; Garello, R. Hybrid Cooperative Positioning Based on Distributed Belief Propagation. *IEEE J. Sel. Areas Commun.* 2011, 29, 1948–1958. [CrossRef]

22. Li, W.; Yang, Z.; Hu, H. Sequential Particle-Based Sum-Product Algorithm for Distributed Inference in Wireless Sensor Networks. *IEEE Trans. Autom. Control* 2001, 45, 477–482. [CrossRef]

23. Rhudy, M.; Gu, Y.; Gross, J.; Napolitano, M.R. Evaluation of Matrix Square Root Operations for UKF within a UAV GPS/INS Sensor Fusion Application. *Int. J. Navig. Obs.* 2011, 416828. [CrossRef]

24. Julier, S.J.; Uhlmann, J.; Hlawatsch, F. Sigma Point Belief Propagation. *IEEE Signal Process. Lett.* 2013, 20, 1352–1365. [CrossRef]

25. Malteswaran, M.; Vaidhe, V.; Irwin, S.; Robin, B. IMM-UKF-TFS Model-based Approach for Intelligent Navigation. *J. Navig.* 2013, 66, 859–877. [CrossRef]

26. Meyer, F.; Hlinka, O.; Hlawatsch, F. Sigma Point Belief Propagation. *Aerosp. Sci. Technol.* 2013, 68, 1352–1365. [CrossRef]

27. Xiong, J.; Cheong, J.W.; Xu, J.; Yu, Y.; Dempster, A.G. Cooperative positioning for low-cost close formation flight based on relative estimation and belief propagation. *Aerosp. Sci. Technol.* 2020, 106, 106068. [CrossRef]

28. Meyer, F.; Hlinka, O.; Hlawatsch, F. Sigma Point Belief Propagation. *IEEE Signal Process. Lett.* 2013, 21, 145–149. [CrossRef]

29. Luo, F.; Wang, S.; Gong, Y.; Jing, X.; Zhang, L. Geographical Information Enhanced Cooperative Localization in Vehicular Ad-Hoc Networks. *IEEE Signal Process. Lett.* 2018, 25, 556–560. [CrossRef]

30. Wang, R.; Xiong, Z.; Liu, J.; Xu, J.; Shi, L. Chi-square and SPRT combined fault detection for multisensor navigation. *IEEE Trans. Aerosp. Electron. Syst.* 2016, 52, 1352–1365. [CrossRef]

31. Luo, F.; Wang, S.; Gong, Y.; Jing, X.; Zhang, L. Geographical Information Enhanced Cooperative Localization in Vehicular Ad-Hoc Networks. *IEEE Signal Process. Lett.* 2018, 25, 556–560. [CrossRef]

32. Xiong, J.; Cheong, J.W.; Xiong, Z.; Dempster, A.G.; Tian, S.; Wang, R. Integrity for Multi-Sensor Cooperative Positioning. *IEEE Trans. Intell. Transp. Syst.* 2021, 22, 792–807. [CrossRef]

33. Yu, K.; Wen, K.; Li, Y.; Zhang, S.; Zhang, K. A Novel NLOS Mitigation Algorithm for UWB Localization in Harsh Indoor Environments. *IEEE Trans. Veh. Technol.* 2019, 68, 686–699. [CrossRef]

34. Wang, J.; Gao, Y.; Li, Z.; Meng, X.; Hancock, C.M. A Tightly-Coupled GPS/INS/UWB Cooperative Positioning Sensors System Supported by V2I Communication. *Sensors* 2016, 16, 944. [CrossRef] [PubMed]

35. Press, W.H.; Teukolsky, S.A.; Vetterling, W.T.; Flannery, B.P. *Numerical Recipes in C: The Art of Scientific Computing*; Cambridge University Press: New York, NY, USA, 1992.

36. Xiong, J.; Cheong, J.W.; Xiong, Z.; Dempster, A.G.; List, M.; Woske, F.; Rievers, B. Carrier-Phase-Based Multi-Vehicle Cooperative Positioning Using V2V Sensors. *IEEE Trans. Veh. Technol.* 2020, 69, 9528–9541. [CrossRef]

37. Alam, N.; Balaei, A.T.; Dempster, A. A DSRC Doppler-Based Cooperative Positioning Enhancement for Vehicular Networks with GPS Availability. *IEEE Trans. Veh. Technol.* 2011, 60, 4462–4470. [CrossRef]

38. Fan, Y.; Qi, X.; Li, B.; Liu, L. Fast clustering-based multidimensional scaling for mobile networks localization. *IET Commun.* 2020, 14, 135–143. [CrossRef]

39. Nguyen, T.M.; Zaini, A.H.; Guo, K.; Lihua, X. An Ultra-Wideband-based Multi-UAV Localization System in GPS-denied Environments. In Proceedings of the International Micro Air Vehicle Conference and Competition 2016 (IMAV 2016), Beijing, China, 18 October 2016.

40. Li, S.; Lv, J.; Tian, S. Posterior Cramer-Rao lower bound for wireless sensor localisation networks. *Electron. Lett.* 2018, 54, 1296–1298. [CrossRef]

41. Yin, F.; Fritsche, C.; Gustafsson, E.; Zoubir, A.M. TOA-Based Robust Wireless Geolocation and Cramér-Rao Lower Bound Analysis in Harsh LOS/NLOS Environments. *IEEE Trans. Signal Process.* 2013, 61, 2243–2255. [CrossRef]