scientific reports

OPEN

Design of robust cubature fission particle filter algorithm in multi-source cooperative navigation

Wei Sun & Jingzhou Liu✉

As a part of the multi-source cooperative navigation scheme, data fusion has a significant impact on the quality of state estimation. Particle filtering has gradually become the focus of many fusion methods due to its unique theoretical advantages in nonlinear non-Gaussian systems. However, the particle degradation and the resulting sample impoverishment restrict its application in complex engineering scenarios. In this paper, a robust cubature fission particle filter (RCFPF) is proposed to deal with these problems. First, in the framework of cubature rule, Huber function is used to combine the L2 norm and L1 norm to improve the importance density function(IDF), suppress the observation noise. Meanwhile, the proposed distribution(PD) is further optimized by combining the Gaussian distribution with Laplace distribution to alleviate particle degradation. Second, the particle swarm is fissioned before resampling, and the particle weight is reconstructed by fission of high weight particles and covering low weight particles to inhibit sample impoverishment. The vehicle experiments of multi-source cooperative navigation show that the proposed algorithm achieves better test results in accuracy and robustness than extended Kalman filter (EKF), strong tracking particle filter (STPF), and cubature particle filter (CPF).

The relative position provided by cooperative navigation is the basis of vehicle intelligent transportation and location-based service[1,2], which can meet a variety of road traffic requirements, including smart transportation, collision avoidance, and congestion reduction. Therefore, it has attracted extensive attention in recent years. In order to improve the efficiency of position perception and obtain higher precision cooperative navigation solutions, scholars have made a lot of research work and gradually improve the quality of navigation parameters[3–7]. However, most of these studies integrate new navigation modes from the perspective of measurement and update the cooperative navigation system to obtain better solutions. There are few improvements on the data fusion methods that play an important role in the solution. The improvement of relevant literature remains within the framework of Kalman filter (KF) requiring linear Gaussian system[8]. Because of the inherent theoretical deficiency, its accuracy in dealing with multi-source complex systems is always limited to some extent. Unlike Kalman filter, particle filter (PF) has almost no requirements for the system and noise characteristics[9], so it becomes a possibility of multi-source cooperative navigation data fusion.

As one of the realization methods of Markov-Bayesian recursion, PF uses a set of weighted particles to iteratively update to approximate a posteriori probability density distribution of the system model. In theory, it can carry out probability reasoning and recursive estimation for any system[10]. Gordon et al. performed resampling after Monte Carlo importance sampling, so that PF began to have a basis for practical applications[11]. Since then, PF has gradually become a focus in the field of data fusion and improved with various theories[12–18]. Although particle filtering has made great progress in recent years, particle degradation and the resulting sample impoverishment problem have always restricted the further development and application of the above-mentioned various improved PFs[12–18]. Particle degradation means that after multiple iterations, only a small number of particles in the particle swarm have large weights, while most of the particles have negligible weights, so that a lot of calculation time and resources are wasted on particles with small weights, which affects the performance of the algorithm[19–21]. Kong has proved that particle degradation is an inherent problem of PF[22]. Doucet and Wang further pointed out that the variance of particle weight increases with time[23,24]. However, the proposed distribution (PD) generated by the importance density function (IDF) in PF is the direct approximation of the particle to the posterior probability distribution, and its simulation accuracy has an important impact on the

School of Geomatics, Liaoning Technical University, Fuxin 123000, Liaoning, China. ✉email: ljz19970304@163.com

estimation performance. Selecting an appropriate IDF can suppress the weight variance and alleviate the degradation problem[25]. Therefore, researchers have carried out a lot of researches around IDF.

The state equation is the earliest and most extensive choice of IDF because of its relative simplicity and easy accessibility[26]. This choice is easy to implement and will not increase additional computational load. However, using the state transition probability density represented by the state equation as the PD does not consider the current measurement information. Since the particle weight is proportional to the likelihood function, choosing the equation of state as the IDF will make the variance of the particle weight larger, especially when the likelihood function is relatively "steep" or located at the tail of the state transition probability distribution, it will accelerate the degradation and even cause the filter divergence[27]. To solve this problem, related scholars introduced the annealing algorithm to optimize the probability density[28,29]. Although the number of particles in the likelihood region is increased, the loop process introduced by the annealing optimization also greatly increases the computational load and limits the practical application of PF. Another idea of IDF selection is to introduce measurement, i.e., using existing filtering methods to make the a priori distribution and the likelihood function have a larger overlap area through the measurement so as to better match the posterior distribution. According to different Bayesian integral approximation strategies, commonly used improved PF includes extended particle filter (EPF)[30] based on EKF, unscented particle filter (UPF) based on Unscented Kalman filter (UKF)[31], and cubature particle filter (CPF) based on cubature Kalman filter (CKF)[32]. Since the use of existing filtering as IDF can greatly improve the calculation performance of PF under the condition of appropriately increasing the amount of calculation, this method is more and more favored by researchers and has achieved considerable development. Liu and Hu have obtained a strong tracking particle filter (STPF) based on the improved EKF, which was used to compensate the model error and apply it to fault detection[33,34]. Zhang uses the truncated adaptive theory to generate the proposed distribution under the CPF framework to optimize the accuracy of the algorithm in strongly nonlinear systems[35]. Havangi uses a particle swarm optimization algorithm to improve UKF, adjust the position and velocity of the particle swarm to improve the accuracy of the posterior probability of particle simulation[36].

However, the various studies mentioned above all use different methods to improve the KF to produce PD to approximate the posterior probability density within the PF framework. Although its accuracy is continuously improved through the combination of various theories, the theoretical basis of these methods are based on the central limit theorem and limited to the Gaussian distribution of L2 norm estimation. The fusion of the L1 norm only revolves around its sparsity and fails to make full use of other properties of the L1 norm[16–18]. In fact, for more complex multi-source systems, the posterior distribution is often affected by multiple factors, which is difficult to approximate it with a single distribution as "irregular". The definition of PD in Gaussian distribution limits the superiority of PF in dealing with general nonlinear non-Gaussian systems. At the same time, various studies focus on the approximation of the real model of the system and few literatures pay a attention to suppressing the interference of various noises of the sensor itself to the PD.

In this paper, for the above-mentioned deficiencies and limitations the Huber M estimation based on the third-order cubature law is applied to generate PD in PF. After the cubature rule is applied to transfer the system states, the algorithm combines the L2 norm and L1 norm using the Huber function for a better approximation of the posterior probability distribution. The posterior probability distribution of the system is redefined through their corresponding Gaussian distribution and Laplace distribution. The proposed idea also can make full use of the robustness of the L1 norm to suppress the interference of observation noise to PD. In order to further deal with the sample impoverishment problem, the algorithm uses the fission method to derive the high weight particles before resampling, and uses the offspring particles to cover the low weight particles in the parent to reconstruct the particle swarm weight. Finally, the robust cubature fission particle filter is formed. The vehicle experiments of multi-source cooperative navigation show that RCFPF performs better in accuracy and robustness than EKF, STPF, and CPF. It provides a new idea for optimizing PD, alleviating particle degradation and sample impoverishment, and also provides a new algorithm for data fusion of multi-source cooperative navigation.

The rest of the paper is organized as follows: the second part introduces the cooperative navigation system, mainly the measurement of each subsystem that needs data fusion. The third part introduces the specific methods of the algorithm, including the process of classical PF, the construction of IDF and the specific process of particle swarm weight reconstruction. In the fourth part, the vehicle experiment of cooperative navigation is carried out, then the results comparison and analysis are given. The fifth part summarizes the paper and gives the conclusion.

Cooperative navigation

Vehicle cooperative navigation is generally realized on the platform of vehicle ad hoc networks (VANET). In order to simplify the system and facilitate interpretation, this paper takes the working logic between vehicles a and b (receiver a and receiver b) in VANET as an example, and its algorithm and conclusion can be extended to ad hoc networks with more vehicles[7]. In this article, the cooperative positioning system uses the pseudo range and Doppler frequency shift acquired by the receiver, the relative distance measured by Ultra Wide Band(UWB) to realize it. The workflow of relative position sensing of cooperative navigation is shown as follows: vehicle a obtains additional information through UWB communication, i.e., the pseudorange and Doppler frequency shift of vehicle b, makes double difference in combination with local pseudorange and Doppler frequency shift, measures the distance between two vehicles with UWB, and performs data fusion of the navigation system with the help of RCFPF to obtain the final cooperative navigation positioning solution. In order to fully demonstrate the algorithm process, we first introduces the fusion measurements required by each navigation system. Second, the system equations of cooperative navigation are given, and the data fusion is carried out in its framework.

GPS observation. As one of the most important observations of the receiver, the pseudorange can be used to determine position information through single-point positioning, i.e., three-dimensional coordinates and

time can be calculated through the pseudorange of at least four satellites. In tightly integrated cooperative navigation, the system only obtains pseudoranges for data fusion and avoids directly obtaining position information. Define the pseudorange between the receiver and satellite at time t as shown in Eq. (1):

$$\rho_a^e(t) = R_a^e(t) + \delta_a(t) + \Omega^e + \zeta_a^e(t) \tag{1}$$

where ρ_a^e is the observed pseudorange from receiver a to satellite e; R_a^e is the true distance from receiver a to satellite e; δ_a is the clock error of receiver a; Ω^e is the common error related to satellite e, such as satellite clock error, atmospheric delay error, etc.; ζ_a^e is the non-common error related to satellite e, such as the thermal noise of receiver a, multipath interference and other non-line-of-sight errors.

When satellite e and satellite j are observed by receiver a and receiver b at the same time, the receiver clock error and satellite-related common error in Eq. (1) can be eliminated by the double differenced pseudoranges of the two satellites, as shown in Eq. (2):

$$
\begin{aligned}
\rho_{ab}^{ej}(t) &= \rho_a^e(t) - \rho_b^e(t) - \rho_a^j(t) + \rho_b^j(t) \\
&= R_{ab}^{ej}(t) + \zeta_{ab}^{ej}(t)
\end{aligned}
\tag{2}
$$

where ρ_{ab}^{ej} represents the double differenced pseudorange observed by receiver a and receiver b on satellite e and satellite j; R_{ab}^{ej} is the double differenced true distance from receiver a and receiver b to satellite e and satellite j; ζ_{ab}^{ej} is the non-common error residual of receiver a and receiver b to satellite e and satellite j that cannot be eliminated by double difference. Define the unit line-of-sight vector $\overrightarrow{\kappa_{sat}}$ from receiver rec to satellite sat:

$$\overrightarrow{\kappa_{sat}} = \frac{r_{sat}(t) - r_{rec}(t)}{\|r_{sat}(t) - r_{rec}(t)\|} \tag{3}$$

where, r_{sat} and r_{rec} are satellite and receiver positions, respectively.

The single difference definition of the real distance between the receiver and the satellite is given by $\overrightarrow{\kappa_{sat}}$:

$$
\begin{cases}
R_a^e - R_b^e = \overrightarrow{\kappa_e}^{\mathrm{T}} \overrightarrow{r_{ab}} \\
R_a^j - R_b^j = \overrightarrow{\kappa_j}^{\mathrm{T}} \overrightarrow{r_{ab}}
\end{cases}
\tag{4}
$$

where $\overrightarrow{\kappa_e}$ and $\overrightarrow{\kappa_j}$ are the unit line-of-sight vectors from receiver a (or receiver b) to satellite e and satellite j, respectively. $\overrightarrow{r_{ab}}$ is the relative distance vector between receiver a and receiver b.

Then the double differenced true range R_{ab}^{ej} can be obtained according to Eq. (4), as shown in Eq. (5):

$$R_{ab}^{ej}(t) = \left[\overrightarrow{\kappa_e}(t) - \overrightarrow{\kappa_j}(t)\right]^{\mathrm{T}} \overrightarrow{r_{ab}}(t) \tag{5}$$

Substituting Eq. (5) into Eq. (2), we can get:

$$\rho_{ab}^{ej}(t) = \left[\overrightarrow{\kappa_e}(t) - \overrightarrow{\kappa_j}(t)\right]^{\mathrm{T}} \overrightarrow{r_{ab}}(t) + \zeta_{ab}^{ej}(t) \tag{6}$$

It should be noted that since the positioning error of Global Positioning System (GPS) is generally within tens of meters, which is almost negligible compared with the distance from the navigation satellite to the earth's surface, the unit line-of-sight vector between the receiver and satellite can be obtained by using the approximate receiver position and the known satellite ephemeris[37].

Doppler frequency shift is the signal frequency change caused by the relative movement between a satellite and a receiver, and can be used to obtain the moving speed of the receiver. In this paper, the definition of Doppler frequency shift $\vartheta_a^e(t)$ of satellite e by receiver a at time t is shown in Eq. (7):

$$\vartheta_a^e(t) = \frac{qu}{c}\frac{dR_a^e(t)}{dt} + qu\frac{d\delta_a(t)}{dt} + qu\frac{d\Omega^e(t)}{dt} + \gamma_a^e(t) \tag{7}$$

where c is the speed of light; qu is the carrier frequency of the satellite signal; γ_a^e is the Doppler observation error. The double differenced Doppler frequency shift $\vartheta_{ab}^{ej}(t)$ is obtained as:

$$\vartheta_{ab}^{ej}(t) = \frac{qu}{c}\frac{dR_{ab}^{ej}(t)}{dt} + \gamma_{ab}^{ej}(t) \tag{8}$$

in Eq. (8), γ_{ab}^{ej} is the residual error of the Doppler observation error between satellite e and satellite j by receiver a and receiver b. Substituting the double differenced true distance into it gets:

$$\vartheta_{ab}^{ej}(t) = \frac{qu}{c}\left[\overrightarrow{\kappa_e}(t) - \overrightarrow{\kappa_j}(t)\right]^{\mathrm{T}} \overrightarrow{v_{ab}}(t) + \gamma_{ab}^{ej}(t) \tag{9}$$

where $\overrightarrow{v_{ab}}$ is the relative velocity between receiver a and receiver b.

UWB observation. UWB takes the half of the product of signal propagation time and propagation speed as the relative distance, and uses at least three reference stations with known coordinates to obtain the three-dimensional coordinates of the target point through resection. Under normal circumstances, the vehicles run on the same road and the driving track is regular, this paper assumes that there are no obstacles between UWB transceivers. In Shen's work[7], the quadratic polynomial error model is derived using the real pulse UWB data

of multispectral Solutions Inc. (MSSI) provided by Joseph et al[38]. The experimental results by Shenshowed that the probability density of the UWB ranging error can be assumed to obey the zero-mean Gaussian distribution with variance of 0.3 m^2, and the UWB measurement data were simulated through these conclusions. In order to control variables and more closely compare and analyze the performance of the proposed algorithm, the UWB data used in this paper remains the same as that inShen's work. The definition of UWB ranging here meets the following requirements:

$$\hat{R}_{ab} = \sqrt{(x_a - x_b)^2 + (y_a - y_b)^2 + (s_a - s_b)^2} + \varsigma_{ab} \tag{10}$$

where x, y, s are the three-dimensional positions of vehicles a and b, respectively; ς_{ab} is the observation noise at UWB distance.

System modeling.   The state equation of the cooperative navigation system is defined as:

$$\chi(t + \tau) = \Phi\chi(t) + \Gamma\varsigma(t) \tag{11}$$

where τ is the measurement interval; χ is the state vector; Φ is the state transition matrix; Γ is the process noise model; ς is the three-axis relative acceleration noise, and satisfies a zero-mean Gaussian distribution with a standard deviation of $\sigma = 1$. The covariance matrix of the process noise is set to $\mathbf{Q} = \sigma^2\Gamma\Gamma^T$. The specific representations of each matrix are as follows:

$$\chi = \begin{bmatrix} \overrightarrow{\mathbf{r}_{ab}} & \overrightarrow{\mathbf{v}_{ab}} \end{bmatrix}^T, \Phi = \begin{bmatrix} \mathbf{I}_3 & \tau\mathbf{I}_3 \\ \mathbf{0}_3 & \mathbf{I}_3 \end{bmatrix}, \Gamma = \begin{bmatrix} 0.5\tau^2\mathbf{I}_3 & \tau\mathbf{I}_3 \end{bmatrix}^T$$

The measurement equation of the system is showed as the Eq. (12):

$$\mathbf{Z}(t) = h(\chi(t)) + \zeta(t) \tag{12}$$

where $\mathbf{Z}$ is the vector of measurements, including the double differenced pseudoranges, double differenced Doppler frequency shifts and relative distances between vehicles; h is a nonlinear function composed of Eqs. (6), (9) and (10); ζ is the noise of the observation vector; the covariance matrix of the system's observations is defined as $\mathbf{\Pi}$. If the number of simultaneously observed GPS satellites is L, the system observation vector and its observation error can be defined as:

$$\mathbf{Z} = \begin{bmatrix} \rho_{ab}^{12} & \cdots & \rho_{ab}^{1L} & \vartheta_{ab}^{12} & \cdots & \vartheta_{ab}^{1L} & \hat{R}_{ab} \end{bmatrix}^T$$

$$\zeta = \begin{bmatrix} \zeta_{ab}^{12} & \cdots & \zeta_{ab}^{1L} & \gamma_{ab}^{12} & \cdots & \gamma_{ab}^{1L} & \varsigma_{ab} \end{bmatrix}^T$$

In order to obtain the covariance matrix of observation noise, we first need to define the observation error variance of pseudo range σ_ρ^2, Doppler frequency shift σ_ϑ^2, UWB ranging error σ_r^2 and the matrix A as the follow:

$$\mathbf{A} = \begin{bmatrix} \mathbf{1}_{(L-1)\times 1} & -\mathbf{I}_{(L-1)} & -\mathbf{1}_{(L-1)\times 1} & \mathbf{I}_{(L-1)} \end{bmatrix} \tag{13}$$

where **1** represents a matrix whose elements are all **1**. Then, the observation noise matrix can be obtained, as shown in Eq. (14):

$$\mathbf{\Pi} = \begin{bmatrix} \mathbf{\Pi}_\rho & \mathbf{0}_{L-1} & \mathbf{0} \\ \mathbf{0}_{L-1} & \mathbf{\Pi}_\vartheta & \mathbf{0} \\ \mathbf{0}_{1\times(L-1)} & \mathbf{0}_{1\times(L-1)} & \mathbf{\Pi}_r \end{bmatrix} \tag{14}$$

where $\mathbf{\Pi}_\rho = \sigma_\rho^2\mathbf{A}\mathbf{A}^T, \mathbf{\Pi}_\vartheta = \sigma_\vartheta^2\mathbf{A}\mathbf{A}^T, \mathbf{\Pi}_r = \sigma_r^2$.

Robust cubature fission particle filter
Classical particle filter.   Filtering can fuse a variety of measurement data, and use these data to finally obtain the estimated target parameters. Its purpose is to estimate, but the process can be regarded as the fusion of multiple measurement data. The particle filter uses weighted particles to perform Bayesian estimation based on Monte Carlo idea. Under the assumption of the first-order Markov process, the algorithm logic is divided into two stages, namely state prediction and time update[39]. In the state prediction stage, the algorithm predicts the particle state through the kinematic model of the system. Liu et al. have proved that $q(\chi_k|\chi_{0:k-1}, \mathbf{Z}_{1:k})$ is the optimal approximation to a posteriori distribution $p(\chi_k|\mathbf{Z}_{1:k})$ [40]. However, in the actual algorithm implementation, because it is difficult to realize, a suboptimal approximation $q(\chi_k|\mathbf{Z}_k)$ (IDF) is often used to randomly extract the particle χ_k^i to generate new particle states, and each state is evaluated by the particle weight obtained by time update, as shown in the following equation:

$$p(\chi_k|\mathbf{Z}_{1:k}) = q(\chi_k|\mathbf{Z}_k)$$
$$= \sum_{i=1}^N \tilde{w}_k^i \delta(\chi_k - \chi_k^i) \tag{15}$$

where N is the number of particles and $\delta()$ is the Dirac function. The particle weight is calculated as follows:

$$\tilde{w}_k^i = \tilde{w}_{k-1}^i \frac{p(\mathbf{Z}_k|\boldsymbol{\chi}_k^i)p(\boldsymbol{\chi}_k^i|\boldsymbol{\chi}_{k-1}^i)}{q(\boldsymbol{\chi}_k^i|\boldsymbol{\chi}_{k-1}^i,\mathbf{Z}_k)} \tag{16}$$

where $p(\mathbf{Z}_k|\boldsymbol{\chi}_k^i)$ is the likelihood function, $p(\boldsymbol{\chi}_k^i|\boldsymbol{\chi}_{k-1}^i)$ is the state transition probability, and $q(\boldsymbol{\chi}_k^i|\boldsymbol{\chi}_{k-1}^i,\mathbf{Z}_k)$ is the importance density. In order to update the particle state, the weight needs to be normalized:

$$\tilde{w}_k^i = \tilde{w}_k^i / \sum_{i=1}^N \tilde{w}_k^i \tag{17}$$

Considering the problem of particle degradation, resampling technology is used in classical particle filters. First, the effective particle number N_{eff} needs to be judged. When the number of effective particles is less than the specified threshold N_{th}, the algorithm performs resampling to avoid resource waste and improve computing efficiency by eliminating small-weight particles. The calculation of effective particle number is as follows:

$$N_{eff} = \frac{1}{\sum_{i=1}^N (\tilde{w}_k^i)^2} \tag{18}$$

The corresponding threshold N_{th} should be set according to engineering experience, generally $0.3 \sim 0.8\, N_{eff}$. Resampling is performed if N_{eff} is less than the threshold. Otherwise, the posterior estimation of the state is obtained directly to complete the particle filter.

Importance density function design. As mentioned earlier, IDF has a crucial impact on the performance of a particle filter. Li et al. has proved that the measurement with significant noise will produce a less precise PD when introduced into the IDF, causing interference to the simulation of a posteriori distribution[41]. For further explanation, we take CKF as IDF as an example. CKF is an L2 norm estimation based on Gaussian distribution. Although the cubature law compensates some errors from the perspective of system nonlinear error in the generation stage of PD. The L2 idea of minimum variance will square the error in the calculation process and amplify the accuracy loss caused by outliers. Therefore, the measurement outliers far away from the sample population have an obvious impact on the residuals squares sum of the sample population, resulting in the decline or even divergence of the overall estimation accuracy of CPF. Different from L2 estimation, the error probability distribution of L1 estimation belongs to Laplace distribution, which has more significant "thick tail" characteristics[42–44], when the variance of Laplace distribution is equal to that of Gaussian distribution, Laplace distribution has a higher probability density near and far from the mean than Gaussian distribution. So it is more suitable for the likelihood description of significant noise. At the same time, the error growth rate of L1 norm estimation is "linear" and significantly slower than the quadratic growth of L2 norm error, which makes L1 estimation have better suppression effect and stronger robustness against colored noise and outliers.

In this paper, we first fuse L2 and L1 estimation based on Huber function in the more advanced CKF algorithm for nonlinear systems, and combine Gaussian distribution and Laplace distribution to better approximate the posterior probability of the system under the cubature law; secondly, the measurement update of CKF is transformed into linear regression, and solved by Huber M estimation. The M estimation uses Huber weight function as its weight function, which can give full play to the robustness of L1. So the algorithm will reduce the weight of the disturbed measurement, truncate and average the filtering innovation to achieve robustness.

Cubature point calculation. CKF approximates the probability density function by weighting the cubature points. In order to realize function transfer, the algorithm first needs to generate a cubature point set with equal weight according to the cubature criterion:

$$\begin{cases} \xi_i = \sqrt{n}[eye]_i, i = 1,...,2n \\ \omega_i = \frac{1}{2n} \end{cases} \tag{19}$$

where ξ_i is the cubature point; n is the state dimension; $[eye]_i$ represents the projection coordinates of the unit vector in the n-dimensional space on each axis; ω_i is the weight of a cubature point.

State update. First, the cubature point $\boldsymbol{\xi}_i$ is used to obtain the updated state cubature point $\boldsymbol{\chi}_{i,k-1|k-1}$ according to the posterior state estimation $\widehat{\boldsymbol{\chi}}_{k-1|k-1}$ and its covariance matrix $\mathbf{P}_{k-1|k-1}$ at time $k-1$:

$$\boldsymbol{\chi}_{i,k-1|k-1} = \sqrt{\mathbf{P}_{k-1|k-1}}\boldsymbol{\xi}_i + \widehat{\boldsymbol{\chi}}_{k-1|k-1} \tag{20}$$

Second, based on the system dynamics model, the function is transferred through the state cubature points, and the approximation $\boldsymbol{\chi}_{i,k|k-1}^*$ of the state transition probability density of each state cubature point is obtained:

$$\boldsymbol{\chi}_{i,k|k-1}^* = f\left(\boldsymbol{\chi}_{i,k-1|k-1}\right) \tag{21}$$

Finally, the system priori state is obtained by equal-weighted fusion of the state cubature point information, and the priori error covariance matrix is estimated:

$$\widehat{\boldsymbol{\chi}}_{k|k-1} = \frac{1}{2n} \sum_{i=1}^{2n} \boldsymbol{\chi}_{i,k|k-1}^{*} \tag{22}$$

$$\mathbf{P}_{k|k-1} = \frac{1}{2n} \sum_{i=1}^{2n} \boldsymbol{\chi}_{i,k|k-1}^{*} \boldsymbol{\chi}_{i,k|k-1}^{*\mathrm{T}} - \widehat{\boldsymbol{\chi}}_{k|k-1} \widehat{\boldsymbol{\chi}}_{k|k-1}^{\mathrm{T}} + \mathbf{Q}_{k-1} \tag{23}$$

where $\widehat{\boldsymbol{\chi}}_{k|k-1}$ and $\mathbf{P}_{k|k-1}$ are the priori state and its covariance matrix after the weighted average of each cubature point at time k.

Measurement update.   Measurement update needs to use $\widehat{\boldsymbol{\chi}}_{k|k-1}$ and $\mathbf{P}_{k|k-1}$ obtained from state update. First, calculate the new cubature point $\boldsymbol{\chi}_{i,k|k-1}$ according to the a priori error covariance matrix:

$$\boldsymbol{\chi}_{i,k|k-1} = \sqrt{\mathbf{P}_{k|k-1}} \boldsymbol{\xi}_i + \widehat{\boldsymbol{\chi}}_{k|k-1} \tag{24}$$

In order to obtain the measurement likelihood description of each cubature point, it is necessary to use the new cubature point $\boldsymbol{\chi}_{i,k|k-1}$ to generate the observation cubature point $z_{i,k|k-1}$ through the observation equation h:

$$z_{i,k|k-1} = h\big(\boldsymbol{\chi}_{i,k|k-1}\big) \tag{25}$$

Second, perform equal-weighted fusion estimation on the measured cubature points to obtain the measured estimated value $\hat{z}_{k|k-1}$:

$$\hat{z}_{k|k-1} = \frac{1}{2n} \sum_{i=1}^{2n} z_{i,k|k-1} \tag{26}$$

Finally, the measurement variance matrix $\mathbf{P}_{zz,k|k-1}$ and the covariance matrix between state and measurement $\mathbf{P}_{\chi z,k|k-1}$ between the state prediction and the measurement prediction are obtained. The specific equations are as follows:

$$\mathbf{P}_{zz,k|k-1} = \frac{1}{2n} \sum_{i=1}^{2n} z_{i,k|k-1} z_{i,k|k-1}^{\mathrm{T}} - \hat{z}_{k|k-1} \hat{z}_{k|k-1}^{\mathrm{T}} + \mathbf{\Pi} \tag{27}$$

$$\mathbf{P}_{\chi z,k|k-1} = \sum_{i=1}^{2n} \omega_i \boldsymbol{\chi}_{i,k|k-1} z_{i,k|k-1}^{\mathrm{T}} - \widehat{\boldsymbol{\chi}}_{k|k-1} \hat{z}_{k|k-1}^{\mathrm{T}} \tag{28}$$

M estimation to solve linear regression.   After the measurement update is completed, a linear regression equation needs to be constructed, and the measurement update is further transformed into solving the linear regression problem.

Using state prediction and measurements combined with observation equation to construct linear regression equation. The relationship between true state and valuation can be expressed as:

$$\delta\boldsymbol{\chi}_k = \boldsymbol{\chi}_k - \widehat{\boldsymbol{\chi}}_{k|k-1} \tag{29}$$

where $\boldsymbol{\chi}_k$ is the true value of the state at time k, and $\delta\boldsymbol{\chi}_k$ is the state prediction error.

The measurement Eq. (12) can be approximated as:

$$\mathbf{Z}_k = h\big(\widehat{\boldsymbol{\chi}}_{k|k-1}\big) + \mathbf{H}_k(\boldsymbol{\chi}_k - \widehat{\boldsymbol{\chi}}_{k|k-1}) + \boldsymbol{\zeta}_k \tag{30}$$

where $\mathbf{H}_k = \left[\big(\mathbf{P}_{k|k-1}\big)^{-1} \mathbf{P}_{\chi z,k|k-1}\right]^{\mathrm{T}}$. Combining Eq. (29) and Eq. (30) to obtain a linear regression equation:

$$\begin{bmatrix} \mathbf{Z}_k - h\big(\widehat{\boldsymbol{\chi}}_{k|k-1}\big) + \mathbf{H}_k\widehat{\boldsymbol{\chi}}_{k|k-1} \\ \widehat{\boldsymbol{\chi}}_{k|k-1} \end{bmatrix} = \begin{bmatrix} \mathbf{H}_k \\ \mathbf{I} \end{bmatrix} \boldsymbol{\chi}_k + \begin{bmatrix} \boldsymbol{\zeta}_k \\ -\delta\boldsymbol{\chi}_k \end{bmatrix} \tag{31}$$

where $\mathbf{I}$ is the unit matrix. Simplify Eq. (31) as follows:

$$\mathbf{\Xi}_k = \mathbf{\Theta}_k \boldsymbol{\chi}_k + \mathbf{\Lambda} \tag{32}$$

Each matrix variable is specifically defined as:

$$\mathbf{\Xi}_k = \mathbf{T}_k^{-1/2} \begin{bmatrix} \mathbf{Z}_k - h\big(\widehat{\boldsymbol{\chi}}_{k|k-1}\big) + \mathbf{H}_k\widehat{\boldsymbol{\chi}}_{k|k-1} \\ \widehat{\boldsymbol{\chi}}_{k|k-1} \end{bmatrix} \tag{33}$$

$$\mathbf{\Theta}_k = \mathbf{T}_k^{-1/2} \begin{bmatrix} \mathbf{H}_k \\ \mathbf{I} \end{bmatrix} \tag{34}$$

$$\Lambda_k = \mathbf{T}_k^{-1/2} \begin{bmatrix} \boldsymbol{\zeta}_k \\ -\delta \boldsymbol{\chi}_k \end{bmatrix} \tag{35}$$

where $\mathbf{T}_k = diag([\boldsymbol{\Pi}_k \mathbf{P}_{k|k-1}])$.

Huber M estimation is used to solve the above linear regression. Define loss function:

$$J(\boldsymbol{\chi}_k) = \sum_{i=1}^{M} \rho(\eta_i) \tag{36}$$

where M is the total dimension of the state and measurements, and η_i is the i-th value of residual vector $\boldsymbol{\eta} = \boldsymbol{\Theta}_k \boldsymbol{\chi}_k - \boldsymbol{\Xi}_k$.

The Huber function as Function $\rho(\eta_i)$, shown as follows:

$$\rho(\eta_i) = \begin{cases} \frac{1}{2}\eta_i^2, & |\eta_i| \geq \gamma \\ \gamma|\eta_i| - \frac{1}{2}\gamma^2, & |\eta_i| > \gamma \end{cases} \tag{37}$$

where γ is the constraint factor, in this paper, it is set to 0.25. When γ is close to 0, the ρ function is close to the Laplace distribution of L1 norm estimation. When γ is close to infinity, the ρ function is close to the Gaussian distribution of L2 norm estimation. M estimation requires the minimum loss function, i.e., it is equivalent to

$$\sum_{i=1}^{M} \varphi(\eta_i) \frac{\partial \eta_i}{\partial \chi} = 0 \tag{38}$$

where

$$\varphi(\eta_i) = \rho'(\eta_i) = \begin{cases} \eta_i, & |\eta_i| \geq \gamma \\ \gamma \, \mathrm{sgn}(\eta_i), & |\eta_i| > \gamma \end{cases} \tag{39}$$

Defining $\psi(\eta_i) = \varphi(\eta_i)/\eta_i$, and the matrix $\boldsymbol{\Psi}$ can be obtained as follows:

$$\boldsymbol{\Psi} = diag[\psi(\eta_i)] \tag{40}$$

Using the above-mentioned matrices to simplify Eq. (38), we obtain Eq. (41):

$$\boldsymbol{\Theta}_k^{\mathrm{T}} \boldsymbol{\Psi} \left(\boldsymbol{\Theta}_k \boldsymbol{\chi}_k - \boldsymbol{\Xi}_k \right) = 0 \tag{41}$$

Defining μ as the number of iterations, using the iterative method to solve Eq. (41), a posteriori state $\hat{\chi}_k$ at time k can be obtained:

$$\hat{\chi}_k^{(\mu+1)} = \left(\boldsymbol{\Theta}_k^{\mathrm{T}} \boldsymbol{\Psi}^\mu \boldsymbol{\Theta}_k \right)^{-1} \boldsymbol{\Theta}_k^{\mathrm{T}} \boldsymbol{\Psi}^\mu \boldsymbol{\Xi}_k \tag{42}$$

The corresponding a posteriori error covariance matrix $\mathbf{P}_{k|k}$ is calculated as follows:

$$\mathbf{P}_{k|k} = \left(\boldsymbol{\Theta}_k^{\mathrm{T}} \boldsymbol{\Psi} \boldsymbol{\Theta}_k \right)^{-1} \tag{43}$$

Weight reconstruction. Resampling is a method to eliminate the low weight particles to avoid the waste of computing resources caused by particle degradation. At the same time, by selecting and copying the high weight particles, resampling can maintain the confidence level of the particle swarm while keeping the total number of particles unchanged. However, this will destroy the diversity of particles. After several iterations, the algorithm will approximate the posterior distribution by only a few high weight particles in the initial particle swarm. The most extreme case is that the offspring particles are the offspring of a "most reliable" particle in the primary generation, i.e., the sample impoverishment. The simplest way to avoid sample impoverishment is to increase the number of particles, but this method is not efficient and will greatly increase the computational load.

In order to further alleviate the sample impoverishment and minimize the increased computational load, fission technology is adopted to derive particles before resampling[45]. First, when the number of effective particles is less than the threshold, the particle swarm is arranged in the order of weight from small to large. Second, fission: take the current particle as the parent, select the first N_{th} high weight particles, take their state as the mean of Gaussian distribution, randomly select the state particles from this distribution to obtain the offspring, and use the offspring particles to replace the low weight particles in the parent in turn, the number of fission particles is proportional to the weight of the parent particles . Finally, normalize the weight of the new particle swarm, take each parent and its offspring particles as a group, evenly distribute the weight of the parent particle, and complete the weight reconstruction of the particle swarm. Compared with other methods of weight reconstruction, the fission method is simple to calculate and easy to implement, which can further alleviate the problem of sample impoverishment.

Figure 1. Test environment and experimental settings.

Equipment	Horizontal accuracy (m)	Vertical accuracy (m)
Leica GS10	0.01	0.02
Novatel INS-LCI	0.02	0.05

Table 1. RTK position accuracy parameters.

Vehicle test results and analysis

In order to verify the performance of the RCFPF algorithm, compare and analyze its impact on the relative position perception of cooperative navigation, the data used in the experiment were consistent with that in the experiment by Shen[7]. In order to obtain the original data of the algorithm, the relative distance between vehicles was obtained by UWB based on MSSI. Meanwhile, in order to obtain the reference truth value, the Leica GS10 receiver and the Novatel INS-LCI system integrating GNSS-INS were equipped on vehicles *a* and *b*, respectively, and the real time kinematic (RTK) positioning of them was implemented to obtained high-precision position estimations, which was used as the truth value to evaluate the algorithm performance. The pseudorange, Doppler frequency shift obtained by satellite receiver and UWB range were used to directly execute the proposed algorithm. Figure 1 shows the test environment and experimental settings. Table 1 shows the accuracy of the equipment under RTK.

The data time in the experiment was about 9 min. During this time, the two vehicles transmitted measurement information through UWB communication. The data sampling rate of GPS was 1 Hz. At the same time, the Differential Global Position System (DGPS) correction was executed through the GPS observations by a reference station of Nottingham Institute of geospatial research. In order to ensure a wide field of vision in the sky during data acquisition and ensure that the receiver can observe as many satellites as possible, the acquisition location is Clifton Boulevard between Derby road and Loughborough Road Road, Nottingham, UK. In the experiment, the initial relative position of the two vehicles is determined by GPS single point positioning, and the initial relative speed is set to zero. The initial state covariance matrix is set to $100 \times \mathbf{I}_6$, meanwhile $\sigma_\rho^2 = \sigma_\vartheta^2 = 1$ which can be used to get the measurement noise matrix $\boldsymbol{\Pi}$ according to the Eq. (14). The experiment is based on the EKF, CPF, STPF and RCFPF to perform data fusion and analysis to draw our conclusions.

Figure 2 shows the number of visible satellites at the time of data collection. It can be seen from the figure that the number of commonly visible satellites is greater than the four required for positioning in most of the time during the data acquisition. In a small amount of time less than four, we set the innovation in the Kalman filter to zero, enlarged the corresponding observation covariance matrix, and increased the distrust of the measurement to ensure the normal update of the state estimation. In order to highlight the accuracy of the approximation of a posterior distribution by RCFPF, the algorithm used a smaller number of particles and set it to 60. Because for approximate efficiency, it is meaningless to use a large number of particles for verification. And the EKF as the evaluation standard of particle approximation accuracy to judge whether these particles have accurately approximated the posterior probability from the accuracy of estimation results. The tight combination with UWB is defined as "T-CP with UWB". Figures 3, 4, 5 and 6 show the coordinate estimation errors of four "T-CP with UWB" filtering algorithms for three-axis and three-dimensional population.

As shown in Figs. 3, 4, 5 and 6, the accuracy of CPF, STPF, EKF, and RCFPF are successively improved, and the accuracy of RCFPF in each axis direction and overall accuracy is the best. This is because in the case of fewer particles, CPF and STPF used one-sided IDF, the PD that they produced cannot well approximate the posteriori distribution, which led to the fact that the accuracy of the two is inferior to that of EKF, and because STPF has adaptive function, its accuracy was slightly better than that of CPF. RCFPF combines the Gaussian distribution
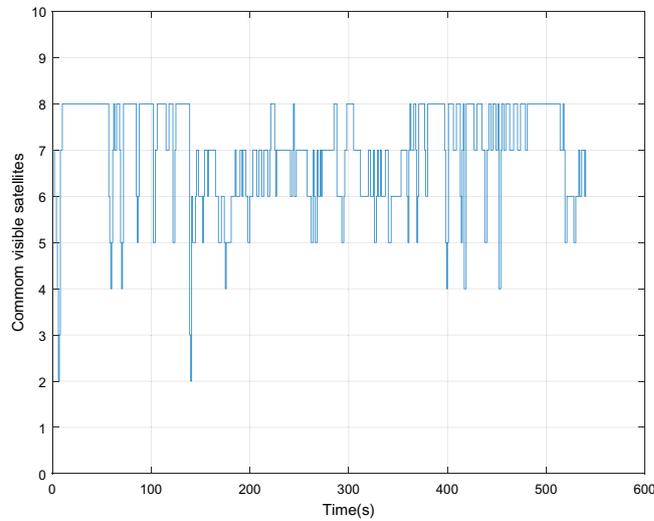
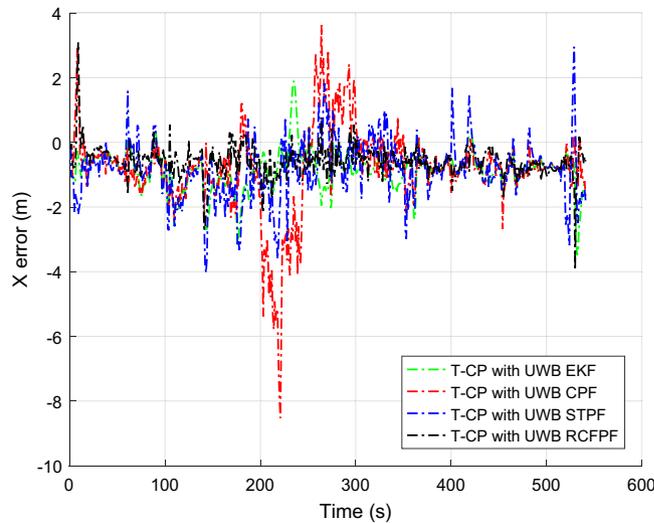**Figure 2.** Number of commonly visible satellites.

Figure 3. X-axis distance error.

and Laplace distribution to better approximate the system probability density, and combines the L1 norm to suppress observation noise, increasing the number of particles in the likelihood region, so it can obtain more accurate results under the condition of fewer particles. At the same time, it can be found that CPF has error fluctuation around 0 s and 200 s, which is because the DOP at these time is poor. Compared with other algorithms at this time, CPF is the most sensitive to DOP. The overall fluctuation of STPF is the strongest. This is because the state equation of the system is a linear equation and there is model error. STPF has a certain adaptive function. It mistakenly transfers the model error to the measurement error for adjustment, resulting in excessive adjustment at some times and large fluctuation of the results. Therefore, STPF may not be suitable for complex integrated navigation with model errors compared with other algorithms.

In order to compare the performance of four data fusion algorithms in more detail, and simultaneously prove the robustness and accuracy of RCFPF. According to the filter estimation error $\left[\vec{\mathbf{r}}'(t) - \vec{\mathbf{r}}(t)\right]$ and its covariance matrix $\boldsymbol{\alpha} = \text{cov}\left[\vec{\mathbf{r}}'(t) - \vec{\mathbf{r}}(t)\right]$, the accuracy $ACC = abs(\ell^{-1}\sum_{t=1}^{\ell}\left[\vec{\mathbf{r}}'(t) - \vec{\mathbf{r}}(t)\right])$ representing the relative position estimation deviation and the precision $PRE = sqrt(\sum_{i=1}^{3}\boldsymbol{\alpha}_{ii})$ representing the algorithm robustness are defined outside root mean squared (RMS). Where $\vec{\mathbf{r}}'$ is the relative position obtained by executing the algorithm; $\vec{\mathbf{r}}$ is the true relative position obtained by RTK; ℓ is the total number of measurement epochs; α_{ii} is the ith diagonal element of the covariance matrix $\boldsymbol{\alpha}$. The quantization standards of the four algorithms are shown in Fig. 7:

In order to further express the improvement of algorithm ϖ relative to algorithm ε, the variable β is defined as follows:
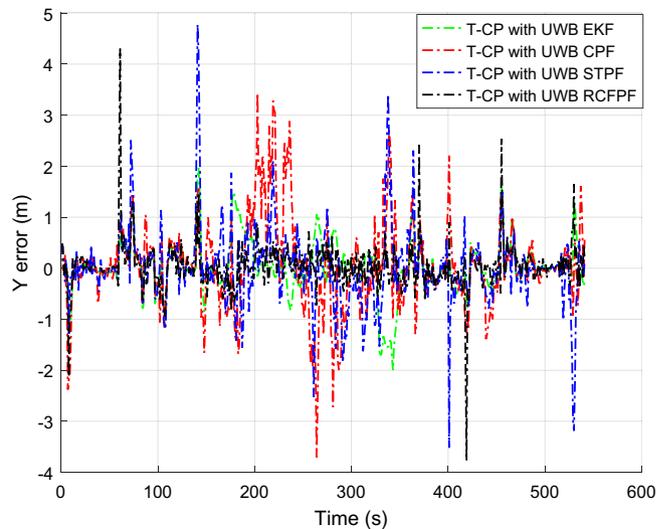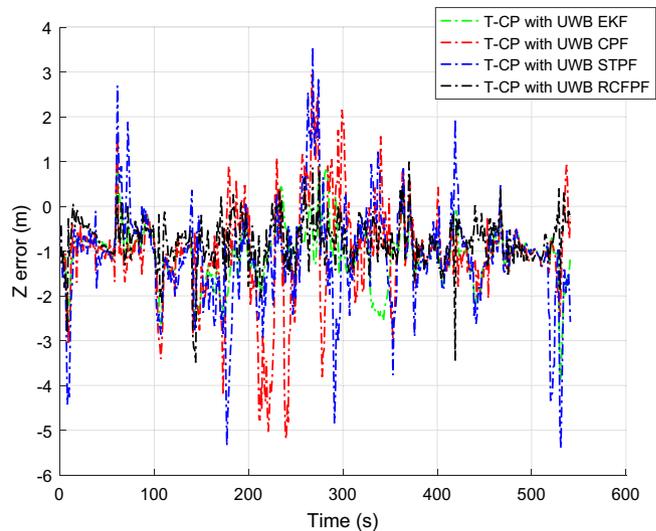
Figure 4. Y-axis distance error.

Figure 5. Z-axis distance error.

$$\beta = \left[\frac{\text{Standard } \varepsilon \text{ - Standard } \varpi}{\text{Standard } \varepsilon} \right] \times 100\% \tag{44}$$

Utilizing β to quantitatively compare the RMS, *ACC* and *PRE* of the four algorithms, the percentage of RCFPF improvement over EKF, CPF and STPF in the three standards is obtained as follows.

It can be seen from Table 2 the RMS, accuracy, and robustness of RCFPF have greatly improved compared to the other three algorithms. The experimental results have confirmed that the algorithm can further optimize PD by improving the probability approximation method and suppressing the observation noise, which improves the estimation performance of particle filter in multi-source cooperative navigation and provides a new system quality control method for vehicle cooperative navigation.

Conclusion

Based on cooperative navigation as a platform, this paper proposes a robust cubature fission particle filter for multi-source data fusion. The algorithm can fit the a posteriori probability density of the system more flexibly through the Gaussian distribution and Laplace distribution. At the same time, it can use the theoretical advantages of the fusion distribution itself to suppress the interference of the observation noise on the proposed distribution without adding additional computational load. The improved importance density function combines the advantages of L2 and L1 norm estimation. Then the resampling combines particle fission smoothing weights to
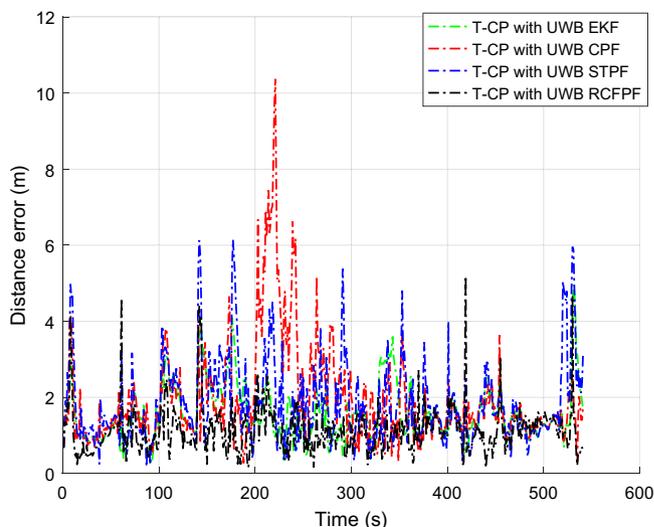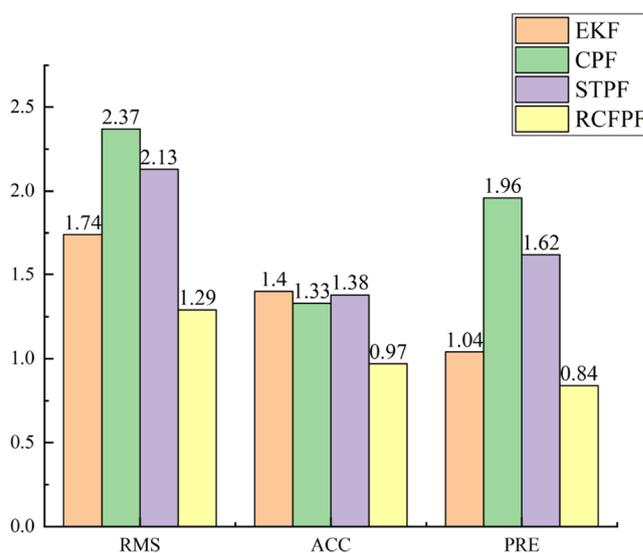
Figure 6. Three-dimensional distance error.

Figure 7. Algorithm quantization standards.

Methods	RMS (m)	Accuracy (m)	Precision (m)
RCFPF over EKF	26.12%	30.67%	18.46%
RCFPF over CPF	45.68%	27.22%	56.85%
RCFPF over STPF	39.56%	29.76%	47.85%

Table 2. Experimental results and performance improvement.

alleviate particle degradation and sample impoverishment. The vehicle experiment of multi-source cooperative navigation shows that compared with EKF, RCFPF improves RMS, Accuracy, and Precision by 26.12%, 30.67%, and 18.46%; compared with CPF, it increases by 45.68%, 27.22%, and 56.85% in RMS, Accuracy, and Precision, respectively, compared with STPF, the RMS, Accuracy, and Precision are increased by 39.56.%, 29.76%, and 47.85% respectively, which achieves the theoretical optimization effect and further improves the accuracy and robustness of the particle filter. The experimental results confirm the superiority of the proposed algorithm, provide a new system state estimation strategy for the multi-source cooperative position, and also provide a way of thinking for the further improvement of the particle filter performance.

Data availability

The datasets generated during and/or analysed during the current study are available from the corresponding author on reasonable request.

References

1. Tatchikou, R., Biswas S. & Dion, F. *5th Global Telecommunications Conference (IGTC).* 2762–2766. (2005).
2. Zhang, H., Zhuang, W. & Yang, Y. Study on vehicle accurate position method in cooperative vehicle infrastructure system. *J. Highway Transp. Res. Dev.* **34**, 137–143. https://doi.org/10.3969/j.issn.1002-0268.2017.05.019 (2017).
3. Xu, B., Shen, L. & Yan, F. Vehicular node positioning based on Doppler-shifted frequency measurement on highway. *J. Electron.* **26**, 265–269. https://doi.org/10.1007/s11767-008-0110-z (2009).
4. Alam, N., TabatabaeiBalaei, A. & Dempster, A. G. A DSRC Doppler-based cooperative positioning enhancement for vehicular networks with GPS availability. *IEEE Trans. Vehicular Technol.* **60**, 4462–4470. https://doi.org/10.1109/tvt.2011.2168249 (2011).
5. Alam, N., Tabatabaei Balaei, A. & Dempster, A. G. Relative positioning enhancement in VANETs: A tight integration approach. *IEEE Trans. Intell Transp. Syst.* **14**, 47–55. https://doi.org/10.1109/tits.2012.2205381 (2013).
6. Alam, N., Kealy, A. & Dempster, A. G. An INS-aided tight integration approach for relative positioning enhancement in VANETs. *IEEE Trans. Intell. Transp. Syst.* **14**, 1992–1996. https://doi.org/10.1109/tits.2013.2265235 (2013).
7. Shen, F., Cheong, J. W. & Dempster, A. G. An ultra-wide bandwidth-based range/GPS tight integration approach for relative positioning in vehicular ad hoc networks. *Meas. Sci. Technol.* https://doi.org/10.1088/0957-0233/26/4/045003 (2015).
8. Shen, F. & Xu, G. An enhanced UWB-based range/GPS cooperative positioning approach using adaptive variational Bayesian cubature Kalman filtering. *Math. Probl. Eng.* **1–8**, 2015. https://doi.org/10.1155/2015/843719 (2015).
9. Yu, J., Tang, Y., Chen, X. & Liu, W. *8th World Congress on Intelligent Control and Automation (WCICA).* 1051–1056. (2010).
10. Pak, J. M., Ahn, C. K., Shmaliy, Y. S., Shi, P. & Lim, M. T. Accurate and reliable human localization using composite particle/FIR filtering. *IEEE Trans. Hum.-Mach. Syst.* **47**, 332–342. https://doi.org/10.1109/thms.2016.2611826 (2017).
11. Gordon, N. J., Salmond, D. J. & Smith, A. F. M. Novel-approach to nonlinear non-Gaussian Bayesian state estimation. *IEEE Proc.-F Radar Signal Process.* **140**, 107–113. https://doi.org/10.1049/ip-f-2.1993.0015 (1993).
12. Xu, M., Xu, C. & Shu, J. *5th International Conference on Computer Sciences and Automation Engineering(ICCSAE).* (2015).
13. Guo, H. *5th International Conference on Computer Sciences and Automation Engineering(ICCSA).* (2015).
14. McGinnity, S. & Irwin, G. W. Multiple model bootstrap filter for maneuvering target tracking. *IEEE Trans. Aerosp. Electron. Syst.* **36**, 1006–1012. https://doi.org/10.1109/7.869522 (2000).
15. Keshavarz-Mohammadiyan, A. & Khaloozadeh, H. *4th International Conference on Control, Instrumentation, and Automation (ICCIA).* 6–11. (2016).
16. Zhou, Z., Zhou, M. & Li, W. Object tracking algorithm based on hybrid particle filter and sparse representation. *Pattern Recognit. Artif. Intell.* **29**, 22–30. https://doi.org/10.16451/j.cnki.issn1003-6059.201601003 (2016).
17. Liu, J. & Li, X. *20th International Conference on Information Fusion (Fusion).* 17–24. (2017).
18. Wang, H., Wang, B. & Gao, Z. Sparse representation and particle filter based online object tracking algorithm. *Trans. Beijing Inst. Technol.* **36**, 635–640. https://doi.org/10.15918/j.tbit1001-0645.2016.06.016 (2016).
19. Gustafsson, F. Particle filter theory and practice with positioning applications. *IEEE Aerosp. Electron. Syst. Mag.* **25**, 53–81. https://doi.org/10.1109/maes.2010.5546308 (2010).
20. Wang, X., Li, T., Sun, S. & Corchado, J. M. A survey of recent advances in particle filters and remaining challenges for multitarget tracking. *Sensors (Basel)* https://doi.org/10.3390/s17122707 (2017).
21. Kang, S. & Yu, M.-J. Ant-Mutated Immune Particle Filter Design for Terrain Referenced Navigation with Interferometric Radar Altimeter. *Remote Sens.* https://doi.org/10.3390/rs13112189 (2021).
22. Kong, A., Liu, J. S. & Wong, W. H. Sequential imputations and bayesian missing data problems. *J. Am. Stat. Assoc.* **89**, 278–288. https://doi.org/10.2307/2291224 (1994).
23. Doucet, A., Godsill, S. & Andrieu, C. On sequential Monte Carlo sampling methods for Bayesian filtering. *Stat. Comput.* **10**, 197–208. https://doi.org/10.1023/a:1008935410038 (2000).
24. Wang, F., Lu, M. & Zhao, Q. Particle filtering algorithm. *Chin. J. Comput.* **37**, 1679–1694. https://doi.org/10.3724/SP.J.1016.2014.01679 (2014).
25. Zan, M., Zhou, H. & Han, D. Survey of particle filter target tracking algorithms. *Comput. Eng. Appl.* **55**, 8–17. https://doi.org/10.3778/j.issn.1002-8331.1809-0242 (2019).
26. Isard, M. & Blake, A. Condensation—Conditional density propagation for visual tracking. *Int. J. Comput. Vis.* **29**, 5–28. https://doi.org/10.1023/a:1008078328650 (1998).
27. Li, T., Fan, H. & Sun, S. Particle filtering: Theory, approach, and application for multitarget tracking. *Acta Autom. Sin.* **41**, 1981–2002. https://doi.org/10.16383/j.aas.2015.C150426 (2015).
28. Du, Z., Tang, F. & Li, K. The hybrid annealed particle filter. *Acta Phys. Sin.* **55**, 999–1004 (2006).
29. Chen, X., & Liu, X. Research on application of hybrid annealed particle filter algorithm in MIMO-OFDM channel estimation. *J. Air Force Eng. Univ. (Natural Science Edition)* **17**, 47–52. https://doi.org/10.3969/j.issn.1009-3516.2016.02.010 (2016).
30. Aggarwal, P., Gu, D. & Nassar, S. *20th International Technical Meeting of the Satellite Division of The Institute of Navigation(ITMSDIN).* (2007).
31. Zhou, J., Yang, Y. & Zhang, J. *24th International Technical Meeting of The Satellite Division of the Institute of Navigation (ITMSDIN).* (2011).
32. Sun, F. & Tang, L. Cubature particle filter. *Syst. Eng. Electron.* **33**, 2554–2557. https://doi.org/10.3969/j.issn.1001-506X.2011.11.39 (2011).
33. Liu, S. & He, W. *26th Chinese Control Conference(CCC).* (2007).
34. Hu, C., Zhang, Q. & Qiao, Y. A strong tracking particle filter with application to fault prediction. *Acta Autom. Sin.* **34**, 1522–1528 (2008).
35. Zhang, Y., Cheng, R. & Huang, Y. Truncated adaptive cubature particle filter. *Syst. Eng. Electron.* **38**, 382–391. https://doi.org/10.3969/j.issn.1001-506X.2016.02.22 (2016).
36. Havangi, R. Target tracking based on improved unscented particle filter with Markov chain Monte Carlo. *IETE J. Res.* **64**, 873–885. https://doi.org/10.1080/03772063.2017.1369909 (2017).
37. Teasley, S. P., Hoover, W. M. & Johnson, C. R. Differential GPS navigation. in *IEEE PLANS 80. Position Location and Navigation Symposium.* 9–16. (1980).
38. Djugash, J. & Singh, S. *11th International Symposium on Experimental Robotics (ISER).* (2009).
39. Zhou, N., Lau, L., Bai, R. & Moore, T. A genetic optimization resampling based particle filtering algorithm for indoor target tracking. *Remote Sens.* https://doi.org/10.3390/rs13010132 (2021).

40. Liu, J. S. & Chen, R. Blind deconvolution via sequential imputations. *J. Am. Stat. Assoc.* **90**, 567–576. https://doi.org/10.2307/2291068 (1995).
41. Li, T., Corchado, J. M., Bajo, J., Sun, S. & De Paz, J. F. Effectiveness of Bayesian filters: An information fusion perspective. *Inf. Sci.* **329**, 670–689. https://doi.org/10.1016/j.ins.2015.09.041 (2016).
42. Zhu, W. *et al.* Market power assessment of thermal power units considering renewable energy consumption. *Power Syst. Clean Energy* **35**, 74–82 (2019).
43. He, S., Li, G. & Xu, W. A feature screening for ultra-high dimensional discriminant analysis using rank-based energy distance. *Stat. Res.* **37**, 117–128 (2020).
44. Zeng, W. & Liu, F. An empirical study on asymmetric laplace distribution of stock index returns in China. *Stat. Inf. Forum* **27**, 27–31 (2012).
45. Chen, S. & Zhang, J. Fission bootstrap particle filtering. *Acta Electron. Sin.* **36**, 500–504 (2008).

Acknowledgements

Author contributions

All authors contributed to this article. Conceptualization, W.S. and J.L.; methodology, W.S. and J.L.; software, J.L.; validation, J.L.; formal analysis, J.L.; investigation, J.L.; writing—original draft preparation, J.L.; writing—review and editing, W.S., and J.L.; visualization, J.L.; supervision, W.S.; project administration, J.L.; funding acquisition, W.S., and J.L. All authors have read and agreed to the published version of the manuscript.

Funding

Competing interests

The authors declare no competing interests.

Additional information

Correspondence and requests for materials should be addressed to J.L.

Reprints and permissions information is available at www.nature.com/reprints.

Publisher's note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.