

Article

Adaptive Square-Root Unscented Particle Filtering Algorithm for Dynamic Navigation

Wenhui Wei ^{1,*}, Shesheng Gao ², Yongmin Zhong ³ , Chengfan Gu ⁴ and Gaoge Hu ²¹ School of Geological Engineering and Surveying and Mapping, Chang'An University, Xi'an 710064, China² School of Automatics, Northwestern Polytechnical University, Xi'an 710072, China; gshshnpu@nwpu.edu.cn (S.G.); hugaoge1111@126.com (G.H.)³ School of Engineering, RMIT University, Bundoora, Melbourne 3083, Australia; yongmin.zhong@rmit.edu.au⁴ Department of Mechanical Engineering, University of Melbourne, Parkville, Melbourne 3010, Australia; chengfan.gu@gmail.com

* Correspondence: weiwenhui91@chd.edu.cn; Tel.: +86-136-4926-6206

Received: 26 May 2018; Accepted: 17 July 2018; Published: 18 July 2018



Abstract: This paper presents a new adaptive square-root unscented particle filtering algorithm by combining the adaptive filtering and square-root filtering into the unscented particle filter to inhibit the disturbance of kinematic model noise and the instability of filtering data in the process of nonlinear filtering. To prevent particles from degeneracy, the proposed algorithm adaptively adjusts the adaptive factor, which is constructed from predicted residuals, to refrain from the disturbance of abnormal observation and the kinematic model noise. Cholesky factorization is also applied to suppress the negative definiteness of the covariance matrices of the predicted state vector and observation vector. Experiments and comparison analysis were conducted to comprehensively evaluate the performance of the proposed algorithm. The results demonstrate that the proposed algorithm exhibits a strong overall performance for integrated navigation systems.

Keywords: performance analysis; particle filter; adaptive filtering; Cholesky factorization; integrated navigation

1. Introduction

Nonlinear filtering is ubiquitous in many areas such as integrated navigation system, geodetic positioning, automatic control, information fusion and signal processing. It aims to estimate the state of a nonlinear dynamic system from observations. The extended Kalman filtering (EKF) is a widely used filtering method for nonlinear systems [1,2]. It linearizes nonlinear system equations by a truncated Taylor series expansion and then applies the linear Kalman filter to the linearized system equations. However, it still requires the linearized state obey the Gaussian distribution, which is usually not consistent with practical applications [3]. Further, when the probability function of state distribution involves multiple peaks, the filtering solution will be biased or even divergent [4]. EKF also involves a complicated calculation process of solving Jacobian matrix. The unscented Kalman filter (UKF) avoids the linearization error of EKF by approximating the probability density of state distribution using unscented transformation (UT) [5,6]. It does not need to calculate Jacobian matrix. However, this method inherits the linear update structure of the Kalman filtering and also requires the system state obey the Gaussian distribution, which is unsuitable for nonlinear systems with non-Gaussian system state model.

The particle filtering (PF) is an optimal recursive Bayesian filtering method based on Monte Carlo simulation [7,8]. Since it is not limited by system linearity and the system state is not subject to the Gaussian

distribution, this method can deal with nonlinear system models with non-Gaussian system state [8–10]. However, PF suffers from the particle degeneracy phenomenon and the accuracy largely depends on the choice of importance sampling density function and resampling scheme [11–13]. Research efforts have been focused on design of a good importance sampling density function and improvement of the resampling scheme to improve the PF performance [14–17]. The unscented particle filtering (UPF) is a method to obtain a better importance sampling density function using UT to approximate the posterior probability density function of the state [17–20]. However, this method still suffers from the particle degeneracy phenomenon if the dynamic system is affected by the disturbances of abnormal observation and kinematic model noise [10,17,20]. In fact, the disturbances caused by abnormal observation or kinematic model noise are unavoidable in practical engineering applications [21,22]. In addition, due to the use of a large number of particles, PF also causes an expensive computational load. The parallel implementation within a shared-memory architecture [23], reduced-order system modelling to reduce the filtering dimensionality [24] and improvement of the algorithm structure can be used to improve the computational performance of PF [18,20].

The robust adaptive filtering is a method to handle the problem of degradation or divergence due to abnormal observation and kinematic model noise. It robustly estimates the covariance matrix of observation noise and adaptively adjusts the covariance matrix of state noise by augmenting the adaptive factor into the covariance matrix of state prediction to improve the filtering robustness [21,22,25]. Yang et al. reported a robust adaptive filter by combining the robust maximum-likelihood estimation with the adaptive filtering process to adaptively adjust the weight matrix of predicted parameters according to the difference between system observation and model information [26]. This filter can be adaptively converted into the classical Kalman filter, adaptive Kalman filter and Sage filter by modifying the weight matrix and adaptive factor. Ding et al. reported a process noise scaling method by improving the robustness of adaptive filtering, where the status of the filter operation is monitored using covariance matching [27]. Gao et al. [28,29] combined the random weighting concept with adaptive filtering for a dynamic navigation system. This method establishes unbiased random weighting estimations of observation and state noises and feeds them back to the kinematic and observation models of a dynamic navigation system to improve the filtering robustness. Azam et al. [30,31] studied the online input estimation techniques to handle cases in which the input of the robust adaptive filtering is unknown.

There are few studies focusing on the use of robust adaptive filtering to improve the UPF performance. Xue et al. [32] reported a new robust adaptive unscented particle filtering algorithm. In order to prevent particles from degeneracy, this algorithm adaptively determines the equivalent weight function according to robust estimation and adaptively adjusts the adaptive factor constructed from predicted residuals to inhibit the disturbances of abnormal observation and kinematic model noise. However, due to the adaptive adjustment to the covariance matrices of predicted state vector and observation vector, this algorithm cannot guarantee the covariance matrices in the filtering process are positive definite, leading to the illness of the filtering process [33]. The square-root filtering provides a solution to overcome this problem. It can improve the update accuracy of covariance matrices by Cholesky factorization and effectively avoid the negative definiteness of covariance matrices.

This paper presents a new adaptive square-root unscented particle filtering (ASUPF) algorithm by combining adaptive filtering and square-root filtering into UPF. This algorithm uses adaptive factors to reasonably control the statistics of observation and kinematic models to inhibit the disturbances of systematic noises, thus preventing particles from degeneracy. Further, Cholesky factorization is used to suppress the negative definiteness of the covariance matrices of predicted state vector and observation vector. Simulation and experimental analyses as well as comparison analysis with the existing nonlinear filtering algorithms were conducted to comprehensively evaluate the performance of the proposed nonlinear filtering algorithm for dynamic navigation.

2. Construction of Adaptive Factor

The role of the adaptive factor in the filtering process is to correct the predicted values using the observation values, as well as to estimate and correct the unknown or inaccurate system model parameters and noise statistics.

Consider the following nonlinear system

$$\begin{aligned} \mathbf{x}_k &= f(\mathbf{x}_{k-1}, \mathbf{v}_{k-1}) \\ \mathbf{y}_k &= h(\mathbf{x}_k, \mathbf{n}_k) \end{aligned} \quad (1)$$

where $\mathbf{x}_k \in R^n$ is the state vector at epoch k , $\mathbf{y}_k \in R^n$ is the system observation, $\mathbf{v}_k \in R^n$ is the process noise with the variance \mathbf{R}_k , $\mathbf{n}_k \in R^n$ is the observation noise with the variance \mathbf{Q}_k , both $f(\cdot)$ and $h(\cdot)$ are a nonlinear function and $k = 0, 1, \dots, N$ is the sampling epoch.

According to the theory of robust estimation, the predicted residual vector reflects the disturbance of the dynamic system, since it contains the state information that has not been corrected by observation. Therefore, the predicted residual vector can be used as the variable to construct the error discriminant statistic and adaptive factor of the kinematic model. The predicted residual vector at time k can be expressed as

$$\bar{\mathbf{V}}_k = \mathbf{y}_k - \bar{\mathbf{y}}_k \quad (2)$$

where $\bar{\mathbf{y}}_k$ is the predicted observation vector.

Accordingly, the error discriminant statistic can be constructed by using $\bar{\mathbf{V}}_k$

$$\Delta \bar{\mathbf{V}}_k = \left(\frac{(\bar{\mathbf{V}}_k)^T \bar{\mathbf{V}}_k}{\text{tr}(\mathbf{P}_{y_k y_k})} \right)^{\frac{1}{2}} \quad (3)$$

where $\Delta \bar{\mathbf{V}}_k$ is the error of the predicted residual vector, $\mathbf{P}_{y_k y_k}$ is the covariance matrix of the predicted observation vector and $\text{tr}(\cdot)$ represents the trace of a matrix. According to (3), three kinds of adaptive factor can be constructed, namely the two-segment function adaptive factor, three-segment function adaptive factor and exponential function adaptive factor [26].

The two-segment function adaptive factor can be constructed as

$$\alpha_k = \begin{cases} 1 & |\Delta \bar{\mathbf{V}}_k| \leq c \\ c/|\Delta \bar{\mathbf{V}}_k| & |\Delta \bar{\mathbf{V}}_k| > c \end{cases} \quad (4)$$

where α_k represents the adaptive factor, satisfying $0 \leq \alpha_k \leq 1$ and $c = 1.0 \sim 2.5$ is a constant.

The three-segment function adaptive factor can be constructed as

$$\alpha_k = \begin{cases} 1 & |\Delta \bar{\mathbf{V}}_k| \leq c_0 \\ \frac{c_0}{|\Delta \bar{\mathbf{V}}_k|} \left(\frac{c_1 - |\Delta \bar{\mathbf{V}}_k|}{c_1 - c_0} \right)^2 & c_0 < |\Delta \bar{\mathbf{V}}_k| \leq c_1 \\ 0 & |\Delta \bar{\mathbf{V}}_k| > c_1 \end{cases} \quad (5)$$

where α_k satisfies $0 \leq \alpha_k \leq 1$, $c_0 = 1.0 \sim 1.5$ and $c_1 = 3.0 \sim 8.5$ are constants.

The exponential function adaptive factor can be constructed as

$$\alpha_k = \begin{cases} 1 & |\Delta \bar{\mathbf{V}}_k| \leq c \\ e^{-((\Delta \bar{\mathbf{V}}_k - c)^2)} & |\Delta \bar{\mathbf{V}}_k| > c \end{cases} \quad (6)$$

where α_k satisfies $0 \leq \alpha_k \leq 1$, c is a constant and its value is usually 1.5.

3. Adaptive Square-Root Unscented Particle Filtering Algorithm

Abnormal interference can be caused by various system factors such as the additional thrust change of carrier's manoeuvre, mechanical disturbance, sensors anomaly and systematic noises and various environmental factors such as air resistance, weather conditions and radiation. It will lead to a sudden increase in observation error (i.e., the observation abnormality), or the inconformity of the navigation kinematic model with the actual model (i.e., the model abnormality), leading to a decrease in the accuracy of dynamic navigation. Combined with the advantages of adaptive filtering and square-root filtering, an ASUPF algorithm for nonlinear systems is proposed in this section. This algorithm selects appropriate adaptive factors to control the information of the kinematic and observation models and suppresses the influence of abnormal interference, to improve the filtering accuracy. Simultaneously, in order to suppress the negative definiteness of the covariance matrices, Cholesky factorization is applied to the filtering process.

Consider the nonlinear system described as (1), the ASUPF algorithm includes the following steps.

Step 1: Initialization

Draw N sampling points according to the initial mean and variance. For $k = 0$, $x_0^i \sim p(x_0)$, $i = 1, 2, \dots, N$. Assume

$$\begin{aligned} \bar{x}_0^i &= E[x_0^i] \\ S_0^i &= chol\{E[(x_0^i - \bar{x}_0^i)(x_0^i - \bar{x}_0^i)^T]\} \\ w_0^i &= p(y_0|x_0^i) \end{aligned} \quad (7)$$

where x_0^i and \bar{x}_0^i represent the i th initial particle and its estimated value, S_0^i represents the i th Cholesky factorization factor at the initial time, w_0^i denotes the initial weight of the i th particle and $chol\{\cdot\}$ is the Cholesky factorization operator.

Step 2: For $k = 1, 2, \dots, N$, conduct importance sampling.

(i) Calculate the Sigma points and weights

$$\begin{cases} x_{0,k-1}^i = \bar{x}_{k-1}^i \\ x_{j,k-1}^i = \bar{x}_{k-1}^i + \sqrt{(N+\lambda)}S_{k-1}^i & j = 1, \dots, N \\ x_{j,k-1}^i = \bar{x}_{k-1}^i - \sqrt{(N+\lambda)}S_{k-1}^i & j = n+1, \dots, 2N \end{cases} \quad (8)$$

$$\begin{cases} W_0^m = \lambda/(N+\lambda) \\ W_0^c = \lambda/(N+\lambda) + (1-\alpha^2 + \beta) \\ W_j^m = W_j^c = 0.5/(N+\lambda) & j = 1, \dots, 2N \end{cases} \quad (9)$$

where $x_{j,k-1}^i$ represents the j th Sigma point, W_j represents the weight of the j th Sigma point and $\sum W_j = 1$, $j = 0, 1, \dots, 2N$. $\lambda = \alpha^2(N + \kappa)$ is the size factor, κ is the second-order size factor, N is the number of particles, α is the factor determining the extent of sample distribution with respect to the predicted state mean and $10^{-3} < \alpha \leq 1$. β is usually determined according to the prior knowledge of the distribution of x and $\beta = 2$ is optimal for the Gaussian distribution.

(ii) Predict and update the particles using UKF

According to the kinematic model, the predicted state vector is expressed as

$$\chi_{j,k/k-1}^i = f(\chi_{j,k-1}^i) \quad (10)$$

The estimate of the predicted state vector is calculated by

$$\bar{x}_{k/k-1}^i = \sum_{j=0}^{2N} W_j^m \chi_{j,k/k-1}^i \quad (11)$$

Applying Cholesky factorization to the covariance matrix of the predicted state vector yields

$$\mathbf{S}_{k/k-1}^i = qr \left\{ \left[\sqrt{\mathbf{W}_1^c} (\mathbf{x}_{1:2n,k/k-1}^i - \bar{\mathbf{x}}_{k/k-1}^i) \sqrt{\mathbf{P}_{W_k}} \right] \right\} \quad (12)$$

$$\mathbf{S}_{k/k-1}^i = cholupdate \left\{ \mathbf{S}_{k/k-1}^i, \mathbf{x}_{0,k/k-1}^i - \bar{\mathbf{x}}_{k/k-1}^i, \mathbf{W}_0^c \right\} \quad (13)$$

where $cholupdate\{\cdot\}$ represents the update operator of Cholesky factorization factor.

By using the adaptive factor α_k^i , $\mathbf{S}_{k/k-1}^i$ can be modified

$$\bar{\mathbf{S}}_{k/k-1}^i = \mathbf{S}_{k/k-1}^i / \sqrt{\alpha_k^i} \quad (14)$$

where α_k^i is constructed as (4) and the variance $\mathbf{P}_{y_k y_k}^i$ of observation information can be calculated by $\mathbf{S}_{y_k}^i$ and $\mathbf{S}_{\hat{y}_k}^i$.

According to the observation model, the observation vector can be written as

$$\mathbf{Y}_{k|k-1}^i = h(\mathbf{x}_{k|k-1}^i) \quad (15)$$

The estimate of the observation vector is calculated as

$$\bar{\mathbf{y}}_{k/k-1}^i = \sum_{j=0}^{2N} \mathbf{W}_j^c \mathbf{Y}_{j,k|k-1}^i \quad (16)$$

Applying Cholesky factorization to the covariance matrix of the observation vector yields

$$\mathbf{S}_{y_k}^i = qr \left\{ \left[\sqrt{\mathbf{W}_1^c} (\mathbf{Y}_{1:2n,k/k-1}^i - \bar{\mathbf{y}}_{k/k-1}^i) \sqrt{\mathbf{P}_{k-1}^i} \right] \right\} \quad (17)$$

$$\mathbf{S}_{\hat{y}_k}^i = cholupdate \left\{ \mathbf{S}_{y_k}^i, \mathbf{Y}_{0,k/k-1}^i - \bar{\mathbf{y}}_{k/k-1}^i, \mathbf{W}_0^c \right\} \quad (18)$$

where $qr\{\cdot\}$ represents the QR factorization of matrices.

The covariance matrix of $\mathbf{x}_{j,k|k-1}^i$ and $\mathbf{Y}_{j,k|k-1}^i$ can be obtained as

$$\mathbf{P}_{x_k y_k} = \sum_{j=0}^{2N} \mathbf{W}_j^c [(\mathbf{x}_{j,k|k-1}^i - \bar{\mathbf{x}}_{k|k-1}^i) \cdot (\mathbf{Y}_{j,k|k-1}^i - \bar{\mathbf{y}}_{k|k-1}^i)]^T \quad (19)$$

Update the state vector

$$\bar{\mathbf{x}}_k^i = \bar{\mathbf{x}}_{k|k-1}^i + \mathbf{K}_k^i (\mathbf{y}_k - \bar{\mathbf{y}}_{k|k-1}^i). \quad (20)$$

Update the covariance matrix of the estimated state vector

$$\mathbf{U}^i = \mathbf{K}_k^i \mathbf{S}_{\hat{y}_k}^i \quad (21)$$

$$\mathbf{S}_k^i = cholupdate \left\{ \bar{\mathbf{S}}_{k/k-1}^i, \mathbf{U}^i, -1 \right\} \quad (22)$$

$$\mathbf{P}_k^i = \mathbf{S}_k^i (\mathbf{S}_k^i)^T \quad (23)$$

where the gain matrix is expressed as

$$\mathbf{K}_k^i = (\mathbf{P}_{x_k y_k}^i / (\mathbf{S}_{y_k}^i)^T) / \mathbf{S}_{y_k}^i \quad (24)$$

It can be seen from above that the covariance matrix of the state vector is directly transmitted and updated in the form of Cholesky factorization factor, thus ensuring the positive definiteness of the covariance matrix and enhancing the numerical stability of the update process of the covariance matrix.

When the kinematic model is disturbed, the predicted residual increases and the adaptive factor α_k^i decreases, leading to the reduced utilization of the predicted state. Accordingly, the interference of model abnormality will be suppressed.

Let $N(\bar{x}_k^i, P_k^i)$ calculated by (20) and (23) be the important density function and conduct importance resampling to obtain the new particle $x_k^i \sim N(\bar{x}_k^i, P_k^i)$.

Step 3: Calculate the weights

$$w_k^i = w_{k-1}^i \frac{p(y_k | x_k^i) p(x_k^i | x_{k-1}^i)}{q(x_k^i | x_{k-1}^i, y_k)} \quad (25)$$

and normalize them as $\tilde{w}_k^i = w_k^i / \sum_{i=1}^n w_k^i$.

Step 4: Calculate the estimate threshold

$$\hat{N}_{eff} = 1 / \sum_{i=1}^N (\tilde{w}_k^i)^2 \quad (26)$$

The severity of particle degeneracy can be determined by comparing the result obtained from (26) with the established threshold. The smaller \hat{N}_{eff} is, the worse the particles degeneracy is. In this case, in order to inhibit particles degeneracy, M new particles can be resampled from the posterior density function obtained above. Then, a common weight $1/M$ is assigned to each new particle.

Step 5: Calculate the estimate of the nonlinear state vector

$$\hat{x}_k = \sum_{i=1}^N \tilde{w}_k^i x_k^i \quad (27)$$

Step 6: Go to Step 2 for the state estimation at the next epoch.

In the above recursive process of filtering, the proposed filter constantly checks whether there is a change in the kinematic model. The original kinematic model will be modified according to the change (if any) such that it can adapt to the dynamic change. In other words, the filter itself constantly uses the noise statistical characteristic or gain matrix to reduce the estimated state error, improve the filtering accuracy and provide a better sampling function for the importance sampling process. Simultaneously, the Cholesky factorization of covariance matrices guarantees the stability of the filtering process.

4. Performance Evaluation and Discussion

Experimental analysis was conducted to evaluate the performance of the proposed ASUPF. The comparison analysis of ASUPF with EKF, UKF, PF and UPF was also conducted for the performance evaluation.

4.1. Experimental Setup

An experiment was designed for ground vehicle navigation using a SINS/GPS integrated navigation system. The experimental setup is shown in Figure 1. The test vehicle is a white urban off-road vehicle, where an SINS/GPS integrated navigation system is mounted on the vehicle via the fixed plate dynamic navigation. The vehicle also carries auxiliary facilities including a DC power supply which is mounted on the vehicle via the fixed plate, an industrial personal computer

(IPC), a data memory and an ampere-voltage meter. Table 1 provides the specifications of these auxiliary facilities.

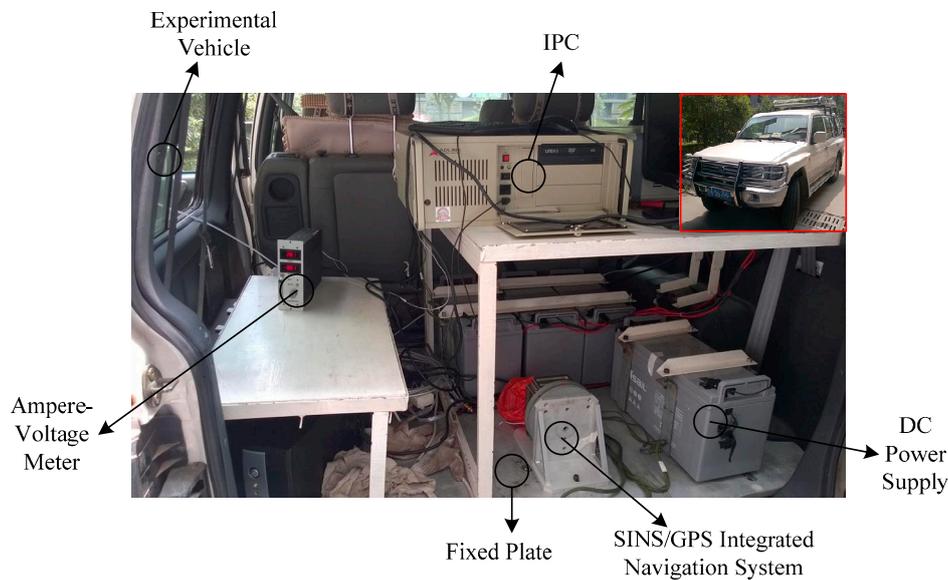


Figure 1. Experimental setup.

Table 1. Specifications of the auxiliary facilities.

Item	Model	Specifications
IPC	ADLINK RK-610A	It is a 2.73 GHz Intel Core Duo CPU and 2.00 GB RAM PC, installed with GPS Status Toolbox PRO v5.1. This PC is equipped with a navigation system interface board and a 17-inch LCD monitor.
Data memory	LCW-S02	It has a RS-232/485 interface, the storage rate is 10 KB/s and the storage capacity is 32 G that can be expanded. The optional baud rate is 4800~115,200 bps. The file system is FAT32 and the storage file format is *.txt. The operating temperature is -35°C ~ 85°C .
DC power supply	Sail 6-GFM-100	It consists of four groups of sustainable and stable discharge batteries, where each battery rated voltage is 12 V and the rated capacity is 30.0 AH (10 h and the termination voltage of 10.8 V).
Ampere-voltage meter	Transmit G-2505	The voltage range is 0~50 V, the current range is 0~5 A and the measurement accuracy is 0.5% FS.
Fixed plate	—	It is a 10 mm thick steel plate with screw holes and bracket.

The framework of the experimental system is shown in Figure 2. The SINS/GPS integrated navigation system provides SINS measurement, GPS positioning and integrated navigation results (position, velocity and attitude), respectively. These navigation data are stored to the data memory through the RS-232 interface and further transferred to IPC for post-processing and filtering. In addition, the GPS Status Toolbox in IPC is adopted to dynamically monitor the environment for GPS measurement, check the number and distribution of observable GPS constellations and control the GPS initialization and operation. The monitoring data are fed back to IPC through the system interface board. The ampere-voltage meter is used to dynamically measure the current and voltage of the system interface to determine whether the SINS/GPS integrated navigation system works normally or not.

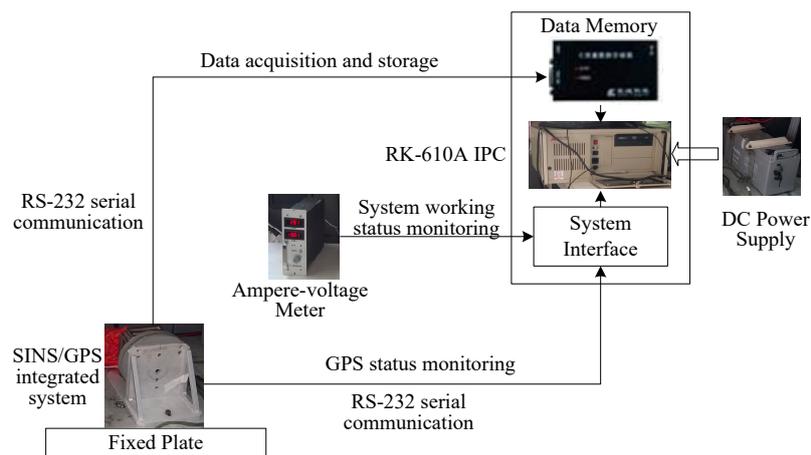


Figure 2. The framework of the experimental system.

The parameters of the SINS/GPS integrated navigation system are provided in Table 2.

Table 2. The parameters of the SINS/GPS integrated navigation system.

Parameter		Value
Update Rate		SINS 125 Hz, GPS 5 Hz
Start Time		<1 s
Operating Temperature		−30 °C~+60 °C
Angular Velocity Measurement	Measuring Range	±200 °/s
	Zero-bias Stability	10.0 °/h (1σ)
	Scale Factor	0.1% (1σ)
	Non-linear	0.01% FS (1σ)
	Random Walk Coefficient	1.0 °/hr ^{1/2} (1σ)
Acceleration Measurement	Measuring Range	±20 g
	Zero-bias Stability	2 mg (1σ)
	Scale Factor	0.1% (1σ)
	Non-linear	0.01% FS (1σ)
	Random Walk Coefficient	0.005 m/s/hr ^{1/2} (1σ)
GPS Measurement	L1/L2	Horizontal Accuracy 1.0 m, Vertical Accuracy 1.5 m (1σ)
	SBAS	Horizontal Accuracy 0.6 m, Vertical Accuracy 1.0 m (1σ)
	DGPS	Horizontal Accuracy 0.3 m, Vertical Accuracy 0.5 m (1σ)
	Velocity Accuracy	0.02 m/s (1σ)

After the one-minute initialization of the SINS/GPS integrated system, the test vehicle started to travel to the East along the Huanshan Road to the Fengyu Kou roundabout. The start position of the vehicle was (E108°46′05.89″, N34°01′41.24″). When arriving at the Fengyu Kou roundabout, the vehicle turned at the position (E108°49′04.61″, N34°03′10.28″) and then travelled back to the start position. The travelling trajectory of the test vehicle and associated position coordinates are shown in Figures 3 and 4, respectively. The travelling distance was 12.38 km, the travelling time was 19 min and the average speed of the vehicle was 39.1 km/h. During the test process, the GPS receiver received signals from at least seven navigation stars. The data obtained from the high-precision differential GPS receiver C-Nav3050 were used as reference for the comparison with the positioning results from the SINS/GPS integration system.



Figure 3. Vehicle traveling trajectory.

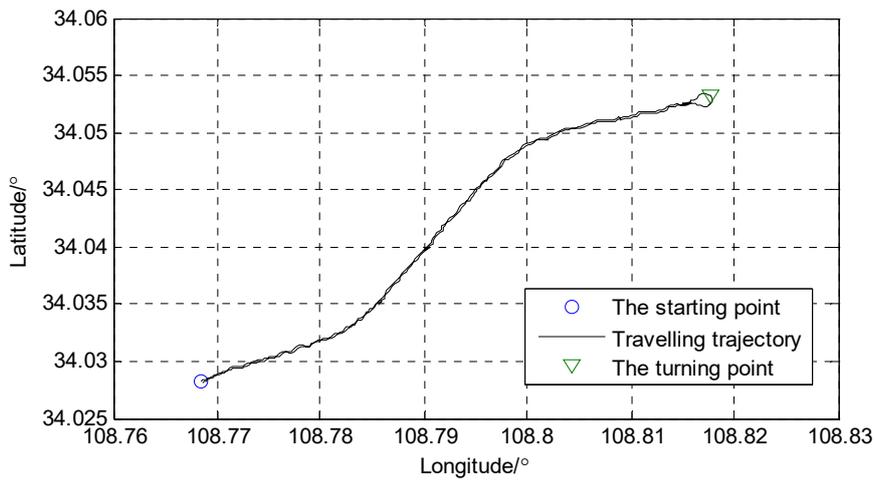


Figure 4. The position coordinates of the vehicle travelling trajectory.

4.2. System Models of SINS/GPS Integrated Navigation

The navigation frame is the E-N-U (East-North-Up) geographic coordinate system. The state vector $x(t)$ of the SINS/GPS integrated navigation system is defined as

$$x(t) = [\phi_E, \phi_N, \phi_U, \delta v_E, \delta v_N, \delta v_U, \delta L, \delta \lambda, \delta h, \varepsilon_x, \varepsilon_y, \varepsilon_z, \nabla_x, \nabla_y, \nabla_z]^T \tag{28}$$

where (ϕ_E, ϕ_N, ϕ_U) is the attitude error, $(\delta v_E, \delta v_N, \delta v_U)$ is the velocity error, $(\delta L, \delta \lambda, \delta h)$ is the position error in latitude, longitude and altitude, $(\varepsilon_x, \varepsilon_y, \varepsilon_z)$ represents the random drift of the gyroscope and $(\nabla_x, \nabla_y, \nabla_z)$ is the constant bias of the accelerometer.

The kinematic model of the SINS/GPS integrated navigation system is expressed as

$$\dot{x}(t) = f(x, t) + G(t)w(t) \tag{29}$$

where $f(x, t)$ is the nonlinear state function of the system, $w(t) = [w_{gx}, w_{gy}, w_{gz}, w_{ax}, w_{ay}, w_{az}]^T$ is the system noise consisting of gyro's Gaussian white noise (w_{gx}, w_{gy}, w_{gz}) and accelerometer's Gaussian white noise (w_{ax}, w_{ay}, w_{az}) and $G(t)$ is the coefficient matrix of the system noise.

The observation model is described as

$$z_k = \delta\rho_k = \mathbf{H}_k \mathbf{x}_k + \frac{1}{2} \begin{bmatrix} \mathbf{x}_k^T \cdot \left(\mathbf{D}_v^T \cdot \mathbf{C}_g^e \cdot \mathbf{H}^{(1)}(r_{ins}) \cdot \mathbf{C}_g^e \cdot \mathbf{D}_v \right) \cdot \mathbf{x}_k \\ \mathbf{x}_k^T \cdot \left(\mathbf{D}_v^T \cdot \mathbf{C}_g^e \cdot \mathbf{H}^{(2)}(r_{ins}) \cdot \mathbf{C}_g^e \cdot \mathbf{D}_v \right) \cdot \mathbf{x}_k \\ \mathbf{x}_k^T \cdot \left(\mathbf{D}_v^T \cdot \mathbf{C}_g^e \cdot \mathbf{H}^{(3)}(r_{ins}) \cdot \mathbf{C}_g^e \cdot \mathbf{D}_v \right) \cdot \mathbf{x}_k \\ \mathbf{x}_k^T \cdot \left(\mathbf{D}_v^T \cdot \mathbf{C}_g^e \cdot \mathbf{H}^{(4)}(r_{ins}) \cdot \mathbf{C}_g^e \cdot \mathbf{D}_v \right) \cdot \mathbf{x}_k \end{bmatrix} + v_k \quad (30)$$

where $\delta\rho_k$ is the pseudo-range difference of GPS satellites, \mathbf{H}_k is the observation matrix, v_k is the observation noise, \mathbf{C}_g^e is the transformation matrix from the geographic coordinate system to the earth coordinate system, r_{ins} is the INS position vector and \mathbf{D}_v is an auxiliary matrix, which is expressed as

$$\mathbf{D}_v = \begin{bmatrix} \mathbf{0}_{3 \times 6} & \mathbf{I}_{3 \times 3} & \mathbf{0}_{3 \times 8} \end{bmatrix} \quad (31)$$

4.3. Filtering Accuracy

For comparison analysis, trials based on the above experimental design were conducted by using EKF, UKF, PF, UPF and ASUPF, respectively. The unscented transformation parameters were $\alpha = 0.5$ and $\beta = 2$. The adaptive factor calculation parameters were $c_0 = 1$ and $c_1 = 3.5$. The sampling time was 1000 s. 50 Monte Carlo simulations were conducted for each of the five filters.

Since the position errors in the other directions have the similar trends as that in the longitude direction, only the position error in the longitude direction is discussed for conciseness. Figure 5 shows the longitude errors of EKF and UKF. It can be seen that EKF has the poor filtering accuracy, due to the error caused by the linearization of the nonlinear state model. Although UKF improves the filtering accuracy of EKF, the improved accuracy is still limited. This is because UKF approximates the posterior probability distribution of the system state using the Gaussian distribution. Its filtering accuracy is significantly degraded when the posterior probability distribution of the system state is non-Gaussian distribution, which is the case of the experimental test. Therefore, both EKF and UKF have limited accuracy for strongly nonlinear systems.

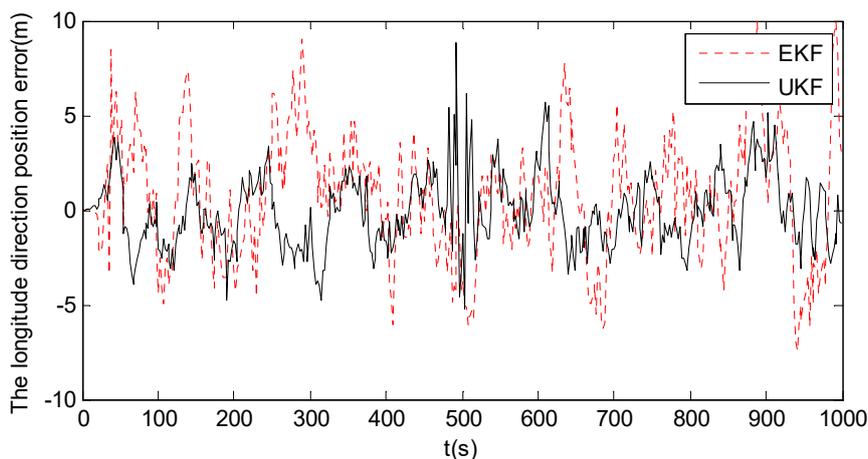


Figure 5. The longitude errors of EKF and UKF.

Figure 6 shows the longitude errors of PF, UPF and ASUPF, where the particle number is $M = 200$. Compared to Figure 5, it is obvious that all three particle filters (PF, UPF and ASUPF) have higher accuracy than both EKF and UKF. This is mainly because these three particle filters describe the priori

and posteriori information using samples instead of a function, thus overcoming the limitation of both EKF and UKF that random variables must satisfy the Gaussian distribution. However, PF suffers from the particle degradation phenomenon, leading to the limited filtering accuracy. UPF improves the filtering accuracy of PF, as it generates the importance function and conducts resampling using UT to weaken the phenomenon of particle degradation. However, due to the influence of abnormal interference on the state estimation, the filtering curve of UPF still involves large oscillations. As clearly shown in Figure 6, the abnormal interference caused by the sharp U-turn travelling at around $t = 500$ s significantly affects the performances of PF and UPF. In contrast, ASUPF improves UPF by introducing the adaptive factor to suppress the influence of abnormal interference on the kinematic and observation models. Therefore, ASUPF has much higher accuracy than both PF and UPF. Table 3 lists the root mean square errors (RMSEs) in the longitude direction for each nonlinear filter.

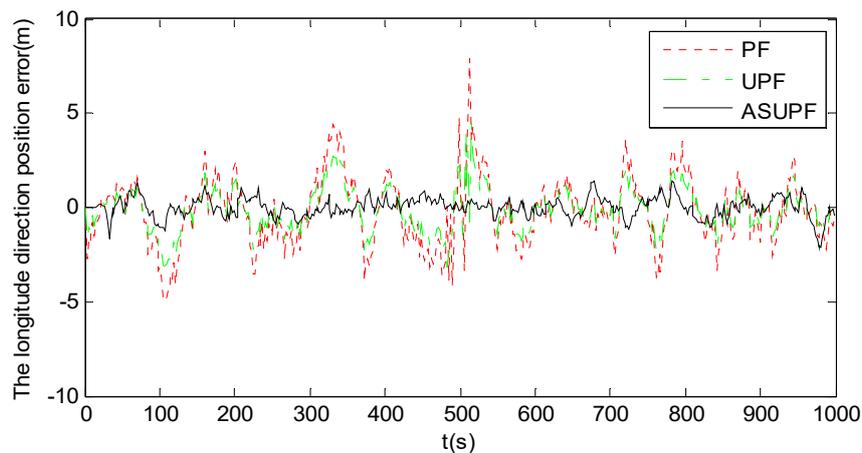


Figure 6. The longitude errors of PF, UPF and ASUPF ($M = 200$).

Table 3. The mean values of the longitude RMSEs for EKF, UKF, UPF and ASUPF.

Filter	Mean Value of Longitude RMSEs/m	Normalized Mean Value
EKF	3.62	0.7240
UKF	2.56	0.5120
PF ($M = 200$)	2.13	0.4260
UPF ($M = 200$)	1.15	0.2300
ASUPF ($M = 200$)	0.46	0.0920

Figure 7 shows the means of the longitude RMSEs for the five filters, where the means of the RMSEs of the three particle filters (PF, UPF and RAUPF) are subject to three different particle numbers $M = 50$, $M = 200$ and $M = 500$. It can be seen that both EKF and UKF involves a large error. However, all three particle filters still have higher accuracy than both EKF and UKF, even with the small number of particles ($M = 50$).

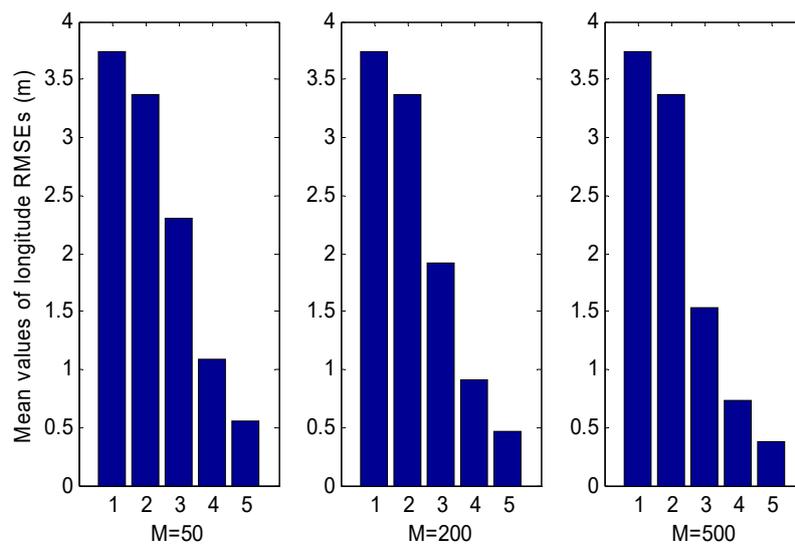


Figure 7. Mean values of the longitude RMSEs of EKF, UKF, PF, UPF and ASUPF, where the mean values for PF, UPF and ASUPF are subject to different particle numbers $M = 50$, $M = 200$ and $M = 500$ and the numbers from 1 to 5 indicate EKF, UKF, PF, UPF and ASUPF, respectively.

4.4. Computational Performance and Filtering Robustness

Trials were conducted with Matlab programs on a 2.93 GHz dual-core CPU and 2 G RAM PC to analyse the computational performances of EKF, UKF, PF, UPF and ASUPF, where the particle number was set to $M = 200$ for PF, UPF and ASUPF. Table 4 shows the computational performances of each filter.

Table 4. Computational performances of EKF, UKF, PF, UPF and ASUPF.

Filter	Equivalent Computational Complexity	Peak of CPU Utilization	Running Time/s	Normalized Running Time/s
EKF	$O(n^3)$	18%	0.202	0.0505
UKF	$O(n^4)$	23%	0.958	0.2395
PF	$O(Mn^3)$	42%	2.411	0.6028
UPF	$O(Mn^3 + n^4)$	48%	3.078	0.7695
ASUPF	$O(Mn^3 + n^4)$	49%	3.089	0.7722

It can be seen that the computational times of PF, UPF and ASUPF are obviously larger than those of EKF and UKF. This is because the computational processes of these three particle filters are more complex, involving sampling a large number of particles, allocating weights and resampling. Thus, they require more CPU utilizations.

In order to analyse the robust performances of EKF, UKF, PF, UPF and ASUPF, the above experimental data were divided into two groups. One was within the sharp U-turn time period (484.2 s, 512.8 s) and the other was within the rest time periods. Based on each group of experimental data, the longitude RMSEs of EKF, UKF, PF, UPF and ASUPF were calculated, where the particle number was set to $M = 200$ for PF, UPF and ASUPF. The RMSE differences between the two groups of experimental data indicate the robust performances of each filter. Table 5 shows the results on the robustness of each filter.

Table 5. Robust performances of EKF, UKF, PF, UPF and ASUPF.

Filter	Longitude Direction Position RMSE/m			Normalized Difference
	The Sharp U-turn Time Period	The Other Time Periods	Difference	
EKF	5.3584	3.5753	1.7831	0.8915
UKF	4.1364	2.8243	1.3121	0.6561
PF	3.1658	2.0370	1.1288	0.5644
UPF	1.5469	0.8663	0.6806	0.3403
ASUPF	0.5517	0.4191	0.1326	0.0663

It can be seen from Table 5 that abnormal disturbances affect EKF, UKF and PF more significantly than UPF and ASUPF. This is also in agreement with the oscillations in the error curves of EKF, UKF and PF as shown in Figures 5 and 6. However, the influence of abnormal disturbances on ASUPF is even more than twice smaller than that on UPF. This is because ASUPF can control the noise statistics of the kinematic and observation models by adjusting the adaptive factor to suppress the influence of abnormal interferences.

4.5. Overall Performance

Define the overall performance index of a filtering algorithm as

$$S = f(p, R_T, R) = \frac{1}{W \cdot (p, R_T, R)^T} \quad (32)$$

where S represents the overall performance index of the filtering algorithm and the larger the value is, the better the performance of the algorithm is p , R_T and R are the three performances of the filtering algorithm, that is, the accuracy, computational performance and robustness, respectively. $W = (\beta_1, \beta_2, \beta_3)$, where β_i ($i = 1, 2, 3$) are the weights of the three performances, respectively, and

$$\sum_{i=1}^3 \beta_i = 1 \quad 0 \leq \beta_i \leq 1 \quad (33)$$

Under different performance requirements of a navigation system, the value of $W = (\beta_1, \beta_2, \beta_3)$ is different. For example, $W = (0.6, 0.2, 0.2)$ indicates that the priority of the navigation system is the positioning accuracy, while the computational performance and robustness are subservient to the positioning accuracy.

Table 6 shows the overall performance indices of EKF, UKF, UPF and ASUPF under three different priorities of accuracy, computational performance and robustness (represented by the three values of W), where the values of p , R_T and R correspond to the normalized values of the three performances as shown in Tables 3–5, respectively.

Table 6. Overall performance indexes of the nonlinear filtering algorithms.

Filtering Algorithms	Accuracy Priority $W = (0.6, 0.2, 0.2)$	Timing Priority $W = (0.2, 0.6, 0.2)$	Robustness Priority $W = (0.2, 0.2, 0.6)$
EKF	1.6056	2.8296	1.4496
UKF	2.0563	2.6503	1.8385
PF	2.0449	1.7866	1.8369
UPF	2.7781	1.7368	2.4748
ASUPF	4.4861	2.0202	4.7030

The overall performance indices of EKF and UKF under the three different priorities show that both EKF and UKF have a strong advantage in the computational performance. Although both accuracy and robustness are weak, the accuracy performance is better than the robustness performance for both EKF and UKF. This is also in agreement with the experimental results of EKF and UKF (see Figures 5 and 7 and Table 5).

The overall performance indices of ASUPF under the three different priorities show that ASUPF has strong accuracy and robustness performances and its robustness is highest in Table 6. This proves that the improvement of ASUPF in adaptability and stability is effective. Although the computational performance of ASUPF is lower than those of EKF and UKF, ASUPF has a much better overall performance than the other filters for the integrated navigation system.

In general, the performance requirements of a navigation system determine the selection of an appropriate filter. For a navigation system desiring high accuracy and strong robustness, ASUPF should be considered. For a navigation system desiring a high computational performance, either EKF or UKF should be considered.

5. Conclusions

This paper presents a new ASUPF for nonlinear systems by combining adaptive filtering and square-root filtering into UPF. This algorithm improves UPF by using the adaptive factor to refrain from the disturbances of the noise statistics of observation and kinematic models, thus overcoming the particle degeneracy problem involved in UPF. It also applies Cholesky factorization to suppress the negative definiteness of the covariance matrices of predicted state vector and observation vector. Experiments and comparison analysis demonstrate that the proposed ASUPF can effectively prevent particles from degeneracy and improve the filtering accuracy of dynamic navigation. Future work will focus on the sensitivity analysis of the proposed ASUPF in comparison with the existing nonlinear filtering algorithms such as EKF, UKF, PF and UPF.

Author Contributions: W.W. completed the theoretical research and design of the manuscript, including the principle and design of the filtering algorithm and completed the writing of the manuscript. S.G. guided and supervised the research content and technical principle of the algorithms in the manuscript. Y.Z. conducted the experimental analysis in the manuscript, including data collection and charts drawing. C.G. provided technical inputs, reviewed the experimental scheme and contributed to the writing of the manuscript. Research associate G.H. completed the literature retrieval and guided the experimental process.

Funding: The work of this paper was supported by the National Natural Science Foundation of China (Project No. 41704016), the China Postdoctoral Science Foundation (Project No. 2017M613029) and the Postdoctoral Research Project Foundation of Shan'xi Province (Project No. 2017BSHEDZZ84).

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

References

1. Li, W.; Jia, Y.; Du, J. Distributed extended Kalman filter with nonlinear consensus estimate. *J. Frankl. Inst.* **2017**, *354*, 7983–7995. [[CrossRef](#)]
2. Li, J.; Wei, X.; Zhang, G. An Extended Kalman Filter-Based Attitude Tracking Algorithm for Star Sensors. *Sensors* **2017**, *17*, 1921.
3. Simon, D. Optimal state estimation. In *Infinity and Nonlinear Approaches*; Kalman, H., Ed.; John Wiley & Sons: Hoboken, NJ, USA, 2006.
4. Hu, G.; Gao, S.; Zhong, Y. A derivative UKF for tightly coupled INS/GPS integrated navigation. *ISA Trans.* **2015**, *56*, 135–144. [[CrossRef](#)] [[PubMed](#)]
5. Liu, X.; Qu, H.; Zhao, J.; Yue, P.; Wang, M. Maximum Correntropy Unscented Kalman Filter for Spacecraft Relative State Estimation. *Sensors* **2016**, *16*, 1530. [[CrossRef](#)] [[PubMed](#)]
6. Gao, S.; Hu, G.; Zhong, Y. Windowing and random weighting-based adaptive unscented Kalman filter. *Int. J. Adapt. Control Signal Process.* **2015**, *29*, 201–223. [[CrossRef](#)]
7. Doucet, A.; Godsill, S.; Andrieu, C. On sequential Monte Carlo methods for Bayesian filtering. *Stat. Comput.* **2000**, *10*, 197–208. [[CrossRef](#)]
8. Rawlings, J.B.; Bakshi, B.R. Particle filtering and moving horizon estimation. *Comput. Chem. Eng.* **2006**, *30*, 1529–1541. [[CrossRef](#)]
9. Zhang, B.; Chen, M.; Zhou, D.; Li, Z. Particle-filter-based estimation and prediction of chaotic states. *Chaos Solitons Fractals* **2007**, *32*, 1491–1498. [[CrossRef](#)]

10. Oppenheim, G.; Philippe, A.; de Rigal, J. The particle filters and their applications. *Chemom. Intell. Lab. Syst.* **2008**, *91*, 87–93. [[CrossRef](#)]
11. Ming, Q.; Jo, K.H. A novel particle filter implementation for a multiple-vehicle detection and tracking system using tail light segmentation. *Int. J. Control Autom. Syst.* **2013**, *11*, 577–585.
12. Arulampalam, M.S.; Maskell, S.; Gordon, N.; Clapp, T. A tutorial on particle filters for online nonlinear/non-Gaussian Bayesian tracking. *IEEE Trans. Signal Process.* **2002**, *50*, 174–188. [[CrossRef](#)]
13. Li, Y.; Sun, S.; Hao, G. A Weighted Measurement Fusion Particle Filter for Nonlinear Multisensory Systems Based on Gauss-Hermite Approximation. *Sensors* **2017**, *17*, 2222. [[CrossRef](#)] [[PubMed](#)]
14. Wang, X.; Ni, W. An improved particle filter and its application to an INS/GPS integrated navigation system in a serious noisy scenario. *Meas. Sci. Technol.* **2016**, *27*, 095005. [[CrossRef](#)]
15. Rabbou, M.A.; El-Rabbany, A. Integration of GPS precise point positioning and MEMS-based INS using unscented particle filter. *Sensors* **2015**, *15*, 7228–7245. [[CrossRef](#)] [[PubMed](#)]
16. Budhiraja, A.; Chen, L.; Lee, C. A survey of numerical methods for nonlinear filtering problems. *Phys. D Nonlinear Phenom.* **2007**, *230*, 27–36. [[CrossRef](#)]
17. Tao, L.; Yuan, G.; Wang, L. Particle Filter with Novel Nonlinear Error Model for Miniature Gyroscope-Based Measurement While Drilling Navigation. *Sensors* **2016**, *16*, 371.
18. Van der Merwe, R.; Doucet, A.; Freitas, N.; Wan, E. *The Unscented Particle Filter*; Tech. Rep. CUED/F-INFENG/TR 380; Engineering Department, Cambridge University: London, UK, 2000.
19. Liang, Z.; Ma, X.; Dai, X. Robust tracking of moving sound source using scaled unscented particle filter. *Appl. Acoust.* **2008**, *69*, 673–680. [[CrossRef](#)]
20. Ali, J.; Fang, J. Realization of an autonomous integrated suite of strapdown astro-inertial navigation systems using unscented particle filtering. *Comput. Math. Appl.* **2009**, *57*, 169–183. [[CrossRef](#)]
21. Yang, Y.; Cui, X. Adaptively robust filter with multi adaptive factors. *Surv. Rev.* **2008**, *40*, 260–270. [[CrossRef](#)]
22. Gao, S.; Zhong, Y.; Li, W. Robust adaptive filtering method for SINS/SAR integrated navigation system. *Aerosp. Sci. Technol.* **2011**, *15*, 425–430. [[CrossRef](#)]
23. Azam, S.E.; Ghisi, A.; Mariani, S. Parallelized sigma-point Kalman filtering for structural dynamics. *Comput. Struct.* **2012**, *92*, 193–205. [[CrossRef](#)]
24. Azam, S.E.; Mariani, S. Online damage detection in structural systems via dynamic inverse analysis: A recursive Bayesian approach. *Eng. Struct.* **2018**, *159*, 28–45. [[CrossRef](#)]
25. Sayed, H.; Rupp, M. Robust issues in adaptive filtering. In *Digital Signal Processing Fundamentals*; Madiseti, V.K., Ed.; Taylor & Francis: Boca Raton, FL, USA, 2010.
26. Yang, Y.; He, H.; Xu, G. A new adaptive robust filtering for kinematic geodetic positioning. *J. Geodesy* **2001**, *75*, 109–116. [[CrossRef](#)]
27. Ding, W.; Wang, J.; Rizos, C. Improving adaptive Kalman estimation in GPS/INS integration. *J. Navig.* **2007**, *60*, 517–529. [[CrossRef](#)]
28. Gao, S.; Gao, Y.; Zhong, Y.; Wei, W. Random weighting estimation method for dynamic navigation positioning. *Chin. J. Aeronaut.* **2011**, *24*, 318–323. [[CrossRef](#)]
29. Gao, S.; Wei, W.; Zhong, Y.; Subic, A. Sage Windowing and Random Weighting Adaptive Filtering Method for Kinematic Model Error. *IEEE Trans. Aerosp. Electron. Syst.* **2015**, *51*, 1488–1500. [[CrossRef](#)]
30. Azam, S.E.; Chatzi, E.; Papadimitriou, C. A dual Kalman filter approach for state estimation via output-only acceleration measurements. *Mech. Syst. Signal Process.* **2015**, *60*, 866–886. [[CrossRef](#)]
31. Azam, S.E.; Chatzi, E.; Papadimitriou, C.; Smyth, A. Experimental validation of the Kalman-type filters for online and real-time state and input estimation. *J. Vib. Control* **2017**, *23*, 2494–2519. [[CrossRef](#)]
32. Xue, L.; Gao, S.; Zhong, Y. Robust adaptive unscented particle filter. *Int. J. Intell. Mechatron. Robot.* **2013**, *3*, 55–56. [[CrossRef](#)]
33. Zhang, N.; Yang, X. Gaussian Mixture Unscented Particle Filter with Adaptive Residual Resample for Nonlinear Model. In Proceedings of the International Conference on Intelligent Computing and Cognitive Informatics, Singapore, 8–9 September 2015.

