



# Article INS/GNSS Tightly-Coupled Integration Using Quaternion-Based AUPF for USV

# Guoqing Xia and Guoqing Wang \*

College of Automation, Harbin Engineering University; Harbin 150001, China; xiaguoqing@hrbeu.edu.cn \* Correspondence: wangguoqing@hrbeu.edu.cn; Tel.: +86-158-465-98526

Academic Editor: Gert F. Trommer Received: 16 May 2016; Accepted: 28 July 2016; Published: 2 August 2016

Abstract: This paper addresses the problem of integration of Inertial Navigation System (INS) and Global Navigation Satellite System (GNSS) for the purpose of developing a low-cost, robust and highly accurate navigation system for unmanned surface vehicles (USVs). A tightly-coupled integration approach is one of the most promising architectures to fuse the GNSS data with INS measurements. However, the resulting system and measurement models turn out to be nonlinear, and the sensor stochastic measurement errors are non-Gaussian and distributed in a practical system. Particle filter (PF), one of the most theoretical attractive non-linear/non-Gaussian estimation methods, is becoming more and more attractive in navigation applications. However, the large computation burden limits its practical usage. For the purpose of reducing the computational burden without degrading the system estimation accuracy, a quaternion-based adaptive unscented particle filter (AUPF), which combines the adaptive unscented Kalman filter (AUKF) with PF, has been proposed in this paper. The unscented Kalman filter (UKF) is used in the algorithm to improve the proposal distribution and generate a posterior estimates, which specify the PF importance density function for generating particles more intelligently. In addition, the computational complexity of the filter is reduced with the avoidance of the re-sampling step. Furthermore, a residual-based covariance matching technique is used to adapt the measurement error covariance. A trajectory simulator based on a dynamic model of USV is used to test the proposed algorithm. Results show that quaternion-based AUPF can significantly improve the overall navigation accuracy and reliability.

Keywords: unscented particle filter; INS/GPS integration; USV; tightly-coupled integration; quaternion

# 1. Introduction

The development of unmanned surface vehicles (USVs) for scientific, military and commercial purpose in applications such as oil and gas exploration, oceanographic data collection, hydrographic, oceanographic and environmental survey, mine counter measure, surveillance and reconnaissance, anti-submarine warfare and fast inshore attack craft for combat training require the corresponding development of navigation systems [1]. The need of the robustness, accuracy and reliability is a guarantee of long duration, unmanned operations for USVs. The sensor units on board, which serve as environment perceptions, are of critical importance for automation operation. A reliable and accurate navigation system is very important for a USV.

The integration of Global Navigation Satellite System (GNSS) and Inertial Navigation System (INS) is widely used for positioning and attitude determination for vehicles. The development of micro-electromechanical system (MEMS) technology has brought low-cost INS/GNSS integration approaches into practice [2,3]. A growing number of research groups are developing integrated navigation systems utilizing INS and GNSS due to the complementary nature of INS and GNSS principles. The INS/GNSS integration can be classified into loosely-coupled, tightly-coupled and deeply-coupled [4]. For the loosely-coupled manner, independent redundant solutions are available

from the GNSS receiver and INS process. However, the disadvantage is that typically four satellites have to be in view to obtain position and velocity solutions from the GNSS receiver. In addition, cascaded filtering problems may occur when one local Kalman filter (KF) is used in GPS data processing and the other is used in integration. These problems can be easily solved in tightly-coupled integration. In tightly-coupled integration, a centralized KF is employed. The pseudorange and delta pseudorange (or Doppler) measurements are used as observations to update the navigation filter. Furthermore, the system does not need a full GNSS solution to assist the INS. This means that the error correlation of inertial measurement unit (IMU) measurements can maintain the update of the estimated solution by the INS even if the number of the tracked satellites is less than four. However, the resulting system observation model turns out to be nonlinear, which should be carefully treated in the design of the integration KF.

The EKF (extended Kalman filter) is known as state-of-the-art for fusion INS and GNSS data in tightly-coupled integration. The linearization should be implemented in both the nonlinear system model and the observation model first in order to apply EKF. This approximation will result in large errors when EKF calculates a posterior mean and covariance, which may lead to suboptimal performance or even divergence of the filter. UKF (unscented Kalman filter) is a recursive MMSE (Minimum Mean Square Error) estimator based on optimal Gaussian approximation Kalman filter framework. Unlike the EKF, the UKF uses the true nonlinear model and approximates the distribution of the state random variables [5]. The state distribution in the UKF is specified using a minimal set of deterministically chosen sample points to capture the posterior mean and covariance accurately to 2nd order for any nonlinearity. It is based on the assumption that it is much easier to approximate a Gaussian distribution than to simulate an arbitrary nonlinear function [6]. Recently, with the development of the computer technology, particle filter (PF) turns out to be more attractive for nonlinear and non-Gaussian applications, and has been successfully used in [7] to recursively update the posterior distribution by sequential importance sampling and resampling. However, the large computational burden impeded the practical use of PF. In addition, the sample impoverishments accompanying the degeneracy of the system performance are primary disadvantages of the basic PF [8]. To overcome these problems, in [3], the strategy of combining the UKF with PF was proposed.

To achieve better performance from the KF framework, the stochastic information provided to the filter must be as accurate as possible. It is therefore necessary to adapt the stochastic model to accommodate for changes in vehicle dynamics and environment conditions. Insufficient or incorrect knowledge about statistics may lead to degradation of the system performance or even divergence of the filter. In [9], a filter innovation sequence based on adaptive Kalman filtering (AKF) was introduced, which showed a major improvement in adjusting process noise error covariance and sensor measurement error covariance adaptively. In this paper, an adaptive unscented particles filter (AUPF) algorithm based on quaternion is proposed using a residual-based covariance matching technique.

In the remainder of this paper, the content is organized as follows. In Section 2, a trajectory simulator with corresponding sensors measurements allocated in a USV is developed. In Section 3, quaternion-based propagation and an observation model are introduced. Furthermore, a quaternion-based GPS/INS integration algorithm using AUPF is proposed. Simulation results are analyzed and compared to illustrate the performance of the proposed AUPF algorithm.

#### 2. Sensor Measurement Simulation

In this part, a USV trajectory simulator is designed. We briefly summarize the different component parts of the USV platform. These include the mathematical dynamic state-space models used for the simulator, the sensors subsystem and the controller design. The core components of such a simulator are schematically depicted in Figure 1, which include a control system, a guidance system and a navigation system (GNC). The GNS system takes the noisy GPS and MEMS-IMU measurements as inputs, and then fuses them from the dynamics model of USV to estimate the optimal vehicle navigation state solutions. There state estimations together with desired trajectories from guidance systems are then adopted by the controller, which generates an optimal control law to drive the thruster of the USV.



Figure 1. Schematic diagram of the simulation system.

# 2.1. Trajectory Simulator of the USV

As in [10], the marine craft equations of motion in six degrees of freedom can be written in vectorial setting form:

$$\dot{\boldsymbol{\eta}} = \boldsymbol{J}_{\boldsymbol{\Theta}}\left(\boldsymbol{\eta}\right)\boldsymbol{\nu} \tag{1}$$

$$M\dot{\nu} + C(\nu)\nu + D(\nu)\nu + g(\eta) = \tau_{wave} + \tau_{wind} + \tau_{currents} + \tau$$
(2)

where  $\eta \in \mathbb{R}^6$  denotes the position and the orientation vector.  $\nu \in \mathbb{R}^6$  denotes the linear and angular velocity vectors that are decomposed in the body-fixed reference frame.  $\tau \in \mathbb{R}^6$  describes the forces and moments acting on the craft in the body-fixed fame. The generalized position, velocity and force vectors have the form that represented as Equation (3):

$$\boldsymbol{\eta} = \begin{bmatrix} \boldsymbol{p}_{b/n}^n \\ \boldsymbol{\Theta}_{nb} \end{bmatrix}, \quad \boldsymbol{v} = \begin{bmatrix} \boldsymbol{v}_{b/n}^b \\ \boldsymbol{\omega}_{b/n}^b \end{bmatrix}, \quad \boldsymbol{\tau} = \begin{bmatrix} f_b \\ m_b \end{bmatrix}$$
(3)

The same notation as in [10] is used,  $p_{b/n}^n \in \mathbb{R}^3$  is the position expressed in the north east down (NED) frame, {N}.  $\Theta_{nb} \in S^3$  represents the Euler angles.  $v_{b/n}^b$  and  $\omega_{b/n}^b$  represent the linear and angular velocity of body frame expressed in{B} frame with respect to {N} frame.  $f_b \in \mathbb{R}^3$  and  $m_b \in \mathbb{R}^3$  are the forces and moments acting on the vehicle, respectively.  $\mathbb{R}^3$  and  $S^3$  denote the Euclidean space of dimension three and the sphere, respectively.  $J_{\Theta}(\eta)$  is the transformation matrix. M, C(v), and D(v) represent the inertial, Coriolis-Centripetal and damping matrices, respectively.  $g(\eta)$  represents the restoring forces and moments.

A sliding mode trajectory tracking controller is designed to track the reference trajectory for USV as in [11]. All relevant position, orientation, linear and angular velocities, acceleration and forces describing the USV's trajectory are calculated.

# 2.2. GPS Date Simulation

There are three kinds of measurements from GPS receiver, i.e., pesudorange, Doppler and carrier phase. In this subsection, we will introduce the measurement mode of pesudorange and Doppler. The first step is to develop the GPS constellation model, which can be used to generate the position of the satellites in the simulation. GPS archive data are available from the website of the International GNSS service (IGS). The next step is to model the signal transmission from the satellites. The main GPS measurement errors include ionospheric errors, tropospheric errors, etc. Further details can be

found in [12]. Considering all of these aspects, the biased pseudorange measurement from one satellite vehicle at one instance is formulated as:

$$\widetilde{\rho} = \rho + c \left( t_u - t_s \right) + T_{iono} + T_{tropo} + \varepsilon_{\rho} \tag{4}$$

where  $\tilde{\rho}$  denotes a measured range of user to satellite,  $\rho$  is a true range of user to satellite,  $t_u$  and  $t_s$  denote receiver clock bias and satellite clock bias, respectively,  $T_{iono}$  and  $T_{tropo}$  denote ionospheric delay and tropospheric delay, respectively,  $\varepsilon_{\rho}$  denotes other un-modeled errors, i.e., multipath delay.

The relative motion between the satellite and the receiver results in change of the observed frequency of satellite signal. The Doppler can be used to estimate the user velocities from the satellite velocities. As in [3], the Doppler shift can be written as a projection of the relative velocity vector on the satellite line-of-sight vector:

$$\widetilde{\dot{\rho}} = \left(I_{u}^{s}\right)^{T} \left(\boldsymbol{v}_{s} - \boldsymbol{v}_{u}\right) + c\dot{\boldsymbol{t}}_{u} + \varepsilon_{\dot{\rho}}$$

$$\tag{5}$$

where  $I_u^s$  is the use-to-satellite line-of-sight unit vector,  $v_s$  and  $v_u$  represent the satellite and receiver velocity respectively.  $I_u^s$  can be expressed as:

$$I_{u}^{s} = \frac{1}{||p_{s} - p_{u}||} [x_{s} - x_{u}, y_{s} - y_{u}, z_{s} - z_{u}]^{T}$$
(6)

where  $p_s = (x_s, y_s, z_s)$  and  $p_u = (x_u, y_u, z_u)$  denote the position vector of satellite and receiver expressed in Earth-Central Earth Fixed (ECEF) coordinate.

# 2.3. IMU Date Simulation

For the purpose of simulating IMU data, raw measurements of accelerometer and gyroscope are needed. The trajectory simulator data are used as the basis in simulating the sensor data. Moreover, the velocity and angular rate data for the USVs are used.

The IMU measurements are provided by extracting the acceleration and angular velocity from the simulator model of USVs, which can be modeled as:

$$f_{IMU} = R_n^b(\Theta) \left( \dot{v}_{nb}^n - g_n \right) + f_b^{bias} + \varepsilon_{acc}^b$$
(7)

$$\boldsymbol{\omega}_{IMU} = \boldsymbol{\omega}_{nb}^{b} + \boldsymbol{\omega}_{b}^{bias} + \boldsymbol{\varepsilon}_{gyro}^{b} \tag{8}$$

where  $g_n$  represents gravity expressed in a navigation frame.  $\varepsilon_{acc}^b$  and  $\varepsilon_{gyro}^b$  denote the zero mean Gaussian distribute noise.  $f_b^{bias}$  and  $\omega_b^{bias}$  are bias errors of the specific forces and angular rate measurements, respectively. For the low cost MEMS-IMU, the sensor errors have the non-Gaussian characteristics. The bias errors in the gyroscope and accelerometer measurements in body frame will be transformed to be the position and velocity drifts in the navigation frame.

#### 3. Quaternion-Based INS/GNSS Integration

# 3.1. Quaternion-Based Propagration and Observation Models

Define the vector form of quaternions as  $q = \begin{bmatrix} q_0 & \overline{q}^T \end{bmatrix}^T$  with one real part  $q_0$  and three imaginary parts given by the vector  $\overline{q} = \begin{bmatrix} q_1 & q_2 & q_3 \end{bmatrix}^T$ . Quaternions are represented as a complex number with four bases and are used to compute the rotation from navigation frame to body frame. Based on Euler's theorem, every change in the relative orientation of two rigid bodies or reference frame can be produced by means of a simple rotation from one frame to another along fixed axes.

Given the invariant axis (rotation axis) and rotation angle (with magnitude  $||\varphi||$ ), the quaternion vector can be represented as:

$$q = \begin{bmatrix} \cos\left(\mathbf{0.5} \mid \varphi \mid \right) \\ \sin\left(\mathbf{0.5} \mid \varphi \mid \right) u \end{bmatrix}, \text{ with } u = \begin{bmatrix} u_x, u_y, u_z \end{bmatrix}^T$$
(9)

where *u* denotes the unit vector along the invariant axis. Apparently, *q* has the normality property, that is to say ||q|| = 1. It can be concluded that the quaternion vector has only three degrees of freedom, although *q* has four elements. Given a rotation vector  $\boldsymbol{\varphi} = [\varphi_1, \varphi_2, \varphi_3]^T$ , the quaternion vector can be computed as:

$$||\varphi|| = \sqrt{\varphi_1^2 + \varphi_2^2 + \varphi_3^2}, u = \frac{\varphi}{||\varphi||}, q = \begin{bmatrix} \cos\left(0.5 ||\varphi|\right) \\ \sin\left(0.5 ||\varphi|\right) \\ \frac{\varphi}{||\varphi||} \end{bmatrix}$$
(10)

Conventionally, with a sufficiently small time interval, the quaternion vector is updated using vectors added together in discrete time domain. However, it should be noticed that the unit sphere defined by quaternion is not a Euclidean vector space. That is to say, the common definition of addition and scaling cannot be applied directly. In the AUPF algorithm, we will apply the quaternion product rule to update the quaternion vector. Thus, the system propagation model in discrete time domain can be expressed as:

$$p_{n,k+1} = p_{n,k} + v_{n,k} \cdot T$$

$$v_{n,k+1} = v_{n,k} + \left[ R_b^n \left( q_k \right) f_{ib,k}^b + g_n \right] \cdot T$$

$$q_{k+1} = q_k \otimes (\Delta q_k)$$

$$f_{b,k+1}^{bias} = f_{b,k}^{bias} + w_f$$

$$\omega_{b,k+1}^{bias} = \omega_{b,k}^{bias} + w_{\omega}$$

$$c\Delta t_{k+1} = c\Delta t_k + c\Delta t_k \cdot T + w_{cbk}$$

$$c\Delta t_{k+1} = c\Delta t_k + w_{cdk}$$
(11)

where  $\otimes$  denotes the quaternion product.  $p_{n,k}$  and  $v_{n,k}$  denote the position and velocity in the navigation frame at epoch k.  $g_n$  represents gravity expressed in navigation fame, which is assumed to be constant for local navigation.  $c\Delta t_k$  denotes the receiver clock drift error, which is modeled as a random walk process.  $c\Delta t_k$  represents the range equivalent of the receiver clock bias, which is the integration of clock drift error. *T* is the system propagation time interval.  $w_f$ ,  $w_\omega$ ,  $w_{cbk}$  and  $w_{cdk}$  are Gaussian nose terms.  $R_b^n(q)$  denotes the rotational transformation matrix from the body frame to the navigation frame, which can be formulated using the parameters of quaternion as:

$$R_b^n = \begin{bmatrix} 1-2(q_2^2+q_3^2) & 2(q_1q_2-q_0q_3) & 2(q_1q_3+q_0q_2) \\ 2(q_1q_2+q_0q_3) & 1-2(q_1^2+q_3^2) & 2(q_2q_3-q_0q_1) \\ 2(q_1q_3-q_0q_2) & 2(q_2q_3+q_0q_1) & 1-2(q_1^2+q_2^2) \end{bmatrix}$$
(12)

 $\Delta q_k$  represents the quaternion rotation of body frame during the time interval:

$$\Delta q_{k} = \begin{bmatrix} \cos\left(0.5 \mid\mid \theta_{k} \mid\mid\right) \\ \sin\left(0.5 \mid\mid \theta_{k} \mid\mid\right) \frac{\theta_{k}}{\mid\mid \theta_{k} \mid\mid} \end{bmatrix}$$
(13)

where  $\theta_k$  denotes the integral of the body frame angular rate measurements.

Equation (11) forms the propagation model of the system. In the tightly-coupled integration, the system observation model is formulated as Equation (14). Where j denotes the number of satellites

in view,  $I_j$  represents the estimated line-of-sight unit vector pointing from the initial estimate of the user position to the *j*-th satellite:

$$\begin{bmatrix} \hat{\rho}_{1,k} - \tilde{\rho}_{1,k} \\ \vdots \\ \hat{\rho}_{j,k} - \tilde{\rho}_{j,k} \\ \hat{\rho}_{1,k} - \tilde{\rho}_{1,k} \\ \vdots \\ \hat{\rho}_{j,k} - \tilde{\rho}_{j,k} \end{bmatrix} = \begin{bmatrix} \begin{pmatrix} \left( -I_{1,k}^T \right)_{1 \times 3} & \mathbf{0}_{1 \times 3} & \mathbf{0}_{1 \times 9} & 1 & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ \left( -I_{j,k}^T \right)_{1 \times 3} & \mathbf{0}_{1 \times 3} & \mathbf{0}_{1 \times 9} & 1 & 0 \\ \mathbf{0}_{1 \times 3} & \left( -I_{1,k}^T \right)_{1 \times 3} & \mathbf{0}_{1 \times 9} & 0 & 1 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ \mathbf{0}_{1 \times 3} & \left( -I_{j,k}^T \right)_{1 \times 3} & \mathbf{0}_{1 \times 9} & 0 & 1 \\ \end{bmatrix}_{2j \times 17} \cdot \begin{bmatrix} \delta p_k^n \\ \delta v_k^n \\ \delta \psi_k^n \\ \delta \psi_{perror}^n \\ \delta \phi_{ib,k}^h \\ c \delta t_k \\ c \delta t_k \end{bmatrix} + \varepsilon_k \quad (14)$$

where  $\hat{\rho}_{j,k}$  is the predicted pseudorange measurement from the *j*-th satellite, and:

$$\hat{\rho}_{j,k} = \sqrt{\left(x_{j,k}^n - \hat{x}_{u,k}^n\right)^2 + \left(y_{j,k}^n - \hat{y}_{u,k}^n\right)^2 + \left(z_{j,k}^n - \hat{z}_{u,k}^n\right)^2} + c\Delta \hat{t}_k \tag{15}$$

where  $\hat{x}_{u,k'}^n \hat{y}_{u,k'}^n \hat{z}_{u,k}^n$  are the USV position estimates expressed in NED frame.  $x_{j,k'}^n y_{j,k'}^n z_{j,k}^n$  are the *j*-th satellite position coordinates expressed in NED frame.

The predicted pseudorange-rate measurements are calculated as:

$$\hat{\hat{\rho}}_{j,k} = I_{j,k}^{n,n} \left( x_{j,k}^n - \hat{x}_{u,k}^n \right) + I_{j,k}^{n,e} \left( y_{j,k}^n - \hat{y}_{u,k}^n \right) + I_{j,k}^{n,d} \left( z_{j,k}^n - \hat{z}_{u,k}^n \right) + c \Delta \hat{t}_k$$
(16)

where

$$\begin{split} I_{j,k}^{n,n} &= \left( x_{j,k}^n - \hat{x}_{u,k}^n \right) / \hat{d}_{j,k}, I_{j,k}^{n,e} &= \left( y_{j,k}^n - \hat{y}_{u,k}^n \right) / \hat{d}_{j,k}, I_{j,k}^{n,d} &= \left( z_{j,k}^n - \hat{z}_{u,k}^n \right) / \hat{d}_{j,k} \\ \hat{d}_{j,k} &= \sqrt{\left( x_{j,k}^n - \hat{x}_{u,k}^n \right)^2 + \left( y_{j,k}^n - \hat{y}_{u,k}^n \right)^2 + \left( z_{j,k}^n - \hat{z}_{u,k}^n \right)^2} \end{split}$$

where  $\hat{\rho}_{j,k}$  is the predicted delta range measurement from the *j*-th satellite.  $\hat{x}_{u,k}^n$ ,  $\hat{y}_{u,k}^n$ ,  $\hat{z}_{u,k}^n$  are the USV velocity estimates expressed in NED frame.  $\dot{x}_{j,k}^n$ ,  $\dot{y}_{j,k}^n$ ,  $\dot{z}_{j,k}^n$  are the *j*-th satellite velocity coordinates expressed in NED frame.

Due to introducing the quaternion, there exists dimension mismatch among the state vector and the state error covariance matrix. The reason is that the degree of freedom of a quaternion vector is three rather than four. When the state vector includes quaternion vector elements, the dimension of the state vector is  $18 \times 1$ ; however, the dimension of the state error covariance matrix is  $17 \times 17$ . In order to solve the problem of dimension mismatch, we introduce the rotation vector in the rotation space, which is transformed from the corresponding quaternion vector error.

## 3.2. Quaternion-Based Integration Using UPF

The idea of UKF comes from the fact that it is much easier to approximate a Gaussian distribution, rather than to simulate a nonlinear function. The particles for UPF are generated using the UKF a posterior estimates. The state vector and its associated errors are defined as:

$$\hat{x}_{k}^{+} = \begin{bmatrix}
\hat{p}_{n,k}^{+} \\
\hat{v}_{n,k}^{+} \\
\hat{q}_{k}^{+} \\
\hat{q}_{k}^{+} \\
\hat{f}_{b,k}^{bias+} \\
\hat{c}\Delta \hat{t}_{k}^{+} \\
c\Delta \hat{t}_{k}^{+} \\
c\Delta \hat{t}_{k}^{+}
\end{bmatrix}_{18 \times 1}, \delta \hat{x}_{k}^{+} = \begin{bmatrix}
\hat{p}_{n,k}^{+} - p_{n,k} \\
\hat{v}_{n,k}^{+} - v_{n,k} \\
\hat{p}_{k}^{+} \\
\hat$$

where  $\hat{\varphi}_k^+$  is the rotation vector corresponding to  $q_k^+ \otimes (q_k)^{-1}$ . The state error covariance  $P_k^+$  can be formulated as:

$$\boldsymbol{P}_{k}^{+} = \begin{bmatrix} \sigma_{\delta\hat{p}}^{-} & 0_{3\times3} \\ 0_{3\times3} & \sigma_{\delta\hat{p}}^{2} & 0_{3\times3} & 0_{3\times3} & 0_{3\times3} & 0_{3\times3} \\ 0_{3\times3} & 0_{3\times3} & \sigma_{\delta\hat{p}}^{2} & 0_{3\times3} & 0_{3\times3} & 0_{3\times3} \\ 0_{3\times3} & 0_{3\times3} & 0_{3\times3} & \sigma_{\delta\hat{f}}^{2} & 0_{3\times3} & 0_{3\times3} & 0_{3\times3} \\ 0_{3\times3} & 0_{3\times3} & 0_{3\times3} & 0_{3\times3} & \sigma_{\delta\hat{c}\hat{f}}^{2} & 0_{3\times3} & 0_{3\times3} \\ 0_{1\times3} & 0_{1\times3} & 0_{1\times3} & 0_{1\times3} & 0_{1\times3} & \sigma_{c\delta\hat{f}}^{2} & 0 \\ 0_{1\times3} & 0_{1\times3} & 0_{1\times3} & 0_{1\times3} & 0_{1\times3} & 0 & \sigma_{c\delta\hat{f}}^{2} \end{bmatrix}_{17\times17}$$
(18)

The sigma-points are generated according to Equation (19)

$$\hat{x}_{k-1,i}^{+} = \hat{x}_{k-1}^{+} + \left(\sqrt{nP_{k-1}^{+}}\right)_{i}^{T}, i = 1, \dots, n$$

$$\hat{x}_{k-1,i+n}^{+} = \hat{x}_{k-1}^{+} - \left(\sqrt{nP_{k-1}^{+}}\right)_{i}^{T}, i = 1, \dots, n$$
(19)

where 2n (n = 17) equally weighted sigma-points are generated. The calculated sigma-points has the form of:

$$\hat{x}_{k-1,i}^{+} = \begin{bmatrix}
\hat{p}_{n,k-1}^{+} + \Delta \hat{p}_{n,k-1,i}^{+} \\
\hat{v}_{n,k-1}^{+} + \Delta \hat{v}_{n,k-1,i}^{+} \\
\hat{v}_{n,k-1}^{+} + \Delta \hat{v}_{n,k-1,i}^{+} \\
\delta q \left(\hat{\varphi}_{k-1,i}^{+}\right) \otimes \hat{q}_{k-1}^{+} \\
\hat{f}_{b,k-1}^{bias+} + \Delta \hat{f}_{b,k-1,i}^{bias+} \\
\hat{\omega}_{b,k-1}^{bias+} + \Delta \hat{\omega}_{b,k-1,i}^{bias+} \\
\hat{c}\Delta \hat{t}_{k-1}^{+} + \Delta c\Delta \hat{t}_{k-1,i}^{+} \\
\hat{c}\Delta t_{k-1}^{+} + \Delta c\Delta \hat{t}_{k-1,i}^{+}
\end{bmatrix}, \hat{x}_{k-1,i+n}^{+} = \begin{bmatrix}
\hat{p}_{n,k-1}^{+} - \Delta \hat{p}_{n,k-1,i}^{+} \\
\hat{v}_{n,k-1}^{+} - \Delta \hat{v}_{n,k-1,i}^{+} \\
\delta q \left(-\hat{\varphi}_{k-1,i}^{+}\right) \otimes \hat{q}_{k-1}^{+} \\
\hat{c}\Delta \hat{t}_{k-1}^{+} + \Delta c\Delta \hat{t}_{k-1,i}^{+} \\
\hat{c}\Delta \hat{t}_{k-1}^{+} - \Delta c\Delta \hat{t}_{k-1,i}^{+} \end{bmatrix}$$
(20)

Define the delta rotation angle rate in the quaternion form as:

$$\delta q \left( \hat{\varphi}_{k-1,i}^{+} \right) = \begin{bmatrix} \cos \left( 0.5 || \hat{\varphi}_{k-1,i}^{+} || \right) \\ \sin \left( 0.5 || \hat{\varphi}_{k-1,i}^{+} || \right) \frac{\hat{\varphi}_{k-1,i}^{+}}{|| \hat{\varphi}_{k-1,i}^{+} ||} \end{bmatrix}$$
(21)

The a priori mean and the state error covariance matrix are computed through Equations (22) and (23):

$$\hat{\mathbf{x}}_{k}^{-} = \frac{1}{2n} \sum_{i=1}^{2n} \mathbf{f}_{k-1} \left( \hat{\mathbf{x}}_{k-1,i}^{+} \right)$$
(22)

$$\boldsymbol{P}_{k}^{-} = \frac{1}{2n} \sum_{i=1}^{2n} \left[ f_{k-1} \left( \hat{x}_{k-1,i}^{+} \right) - \hat{x}_{k}^{-} \right] \left[ f_{k-1} \left( \hat{x}_{k-1,i}^{+} \right) - \hat{x}_{k}^{-} \right]^{T} + \boldsymbol{Q}_{k-1}$$
(23)

The same as in Equation (19), we generate the sigma-points for measurements update by replacing  $P_{k-1}^+$  with  $P_k^-$ . The sigma-points  $\hat{x}_{k,i}^-$  are calculated according to Equation (20). With the sigma-points propagation, the predicted measurement and covariance matrix are calculated according to Equations (24)–(26):

$$\hat{y}_{k} = \frac{1}{2n} \sum_{i=1}^{2n} h_{k} \left( \hat{x}_{k,i}^{-} \right)$$
(24)

$$\boldsymbol{P}_{k}^{yy} = \frac{1}{2n} \sum_{i=1}^{2n} \left[ h_{k} \left( \hat{x}_{k,i}^{-} \right) - \hat{y}_{k} \right] \left[ h_{k} \left( \hat{x}_{k,i}^{-} \right) - \hat{y}_{k} \right]^{T} + \boldsymbol{R}_{k}$$
(25)

$$\boldsymbol{P}_{k}^{xy} = \frac{1}{2n} \sum_{i=1}^{2n} \left[ \hat{x}_{k,i}^{-} - \hat{x}_{k}^{-} \right] \left[ \mathbf{h}_{k} \left( \hat{x}_{k,i}^{-} \right) - \hat{y}_{k} \right]^{T}$$
(26)

At each measurement epoch of GPS, we update the state estimate and the covariance as KF frame:

$$\begin{aligned} \mathbf{K}_{k} &= \mathbf{P}_{k}^{xy} \left( \mathbf{P}_{k}^{yy} \right)^{-1} \\ \hat{\mathbf{x}}_{k}^{+} &= \hat{\mathbf{x}}_{k}^{-} + \mathbf{K}_{k} \left( \tilde{\mathbf{y}}_{k} - \hat{\mathbf{y}}_{k} \right) \\ \mathbf{P}_{k}^{+} &= \mathbf{P}_{k}^{-} - \mathbf{K}_{k} \mathbf{P}_{k}^{yy} \mathbf{K}_{k}^{T} \end{aligned}$$

$$(27)$$

For the UPF, the a posterior estimates of  $\hat{x}_k^+$  and  $P_k^+$  from Equation (27) are used to form the importance density distribution for generating particles:

$$\boldsymbol{\chi}_{k,i}^{+} = \hat{\boldsymbol{\chi}}_{k}^{+} + \Delta \hat{\boldsymbol{\chi}}_{k,i}^{+} = \begin{bmatrix} \hat{\boldsymbol{p}}_{n,k}^{+} + \Delta \hat{\boldsymbol{p}}_{n,k,i}^{+} \\ \hat{\boldsymbol{\sigma}}_{n,k}^{+} + \Delta \hat{\boldsymbol{\sigma}}_{n,k,i}^{+} \\ \hat{\boldsymbol{\sigma}}_{n,k}^{+} + \Delta \hat{\boldsymbol{\sigma}}_{n,k,i}^{+} \\ \delta \boldsymbol{q} \left( \hat{\boldsymbol{\varphi}}_{k,i}^{+} \right) \otimes \hat{\boldsymbol{q}}_{k}^{+} \\ \hat{\boldsymbol{f}}_{b,k}^{bias+} + \Delta \hat{\boldsymbol{f}}_{b,k,i}^{bias+} \\ \hat{\boldsymbol{\omega}}_{b,k}^{bias+} + \Delta \hat{\boldsymbol{\omega}}_{b,k,i}^{bias+} \\ \hat{\boldsymbol{\omega}}_{b,k}^{bias+} + \Delta \hat{\boldsymbol{\omega}}_{b,k,i}^{bias+} \\ c\Delta \hat{\boldsymbol{t}}_{k}^{+} + \Delta c\Delta \hat{\boldsymbol{t}}_{k,i}^{+} \\ \hat{\boldsymbol{\zeta}}_{k}^{+} + \Delta c\Delta \hat{\boldsymbol{t}}_{k,i}^{+} \end{bmatrix}, \Delta \hat{\boldsymbol{x}}_{k,i}^{+} \sim \mathcal{N}\left(0, \boldsymbol{P}_{k}^{+}\right)$$
(28)

where  $i = 1, \dots, N$ , N is the number of particles. The normalized importance weights are computed as:

$$w\left(\boldsymbol{\chi}_{k,i}^{+}\right) = \frac{p\left(\tilde{\boldsymbol{y}}_{k}|\boldsymbol{\chi}_{k,i}^{+}\right)\mathcal{N}\left(\boldsymbol{\chi}_{k,i}^{+}; \hat{\boldsymbol{x}}_{k}^{-}, \boldsymbol{P}_{k}^{-}\right)}{\mathcal{N}\left(\boldsymbol{\chi}_{k,i}^{+}; \hat{\boldsymbol{x}}_{k}^{+}, \boldsymbol{P}_{k}^{+}\right)}$$
(29)

$$\overline{w}\left(\boldsymbol{\chi}_{k,i}^{+}\right) = \frac{w\left(\boldsymbol{\chi}_{k,i}^{+}\right)}{\sum\limits_{i=1}^{N} w\left(\boldsymbol{\chi}_{k,i}^{+}\right)}, \text{ with } \sum\limits_{i=1}^{N} \overline{w}\left(\boldsymbol{\chi}_{k,i}^{+}\right) = 1$$
(30)

where

$$p\left(\tilde{y}_{k}|\boldsymbol{\chi}_{k,i}^{+}\right) = \frac{1}{\sqrt{(2\pi)^{m} ||\boldsymbol{R}_{k}||}} \exp\left\{-\frac{\left[\tilde{y}_{k}-h_{k}\left(\boldsymbol{\chi}_{k,i}^{+}\right)\right]^{T} \boldsymbol{R}^{-1}\left[\tilde{y}_{k}-h_{k}\left(\boldsymbol{\chi}_{k,i}^{+}\right)\right]}{2}\right\}$$
$$\mathcal{N}\left(\boldsymbol{\chi}_{k,i}^{+}; \hat{\boldsymbol{x}}_{k}^{-}, \boldsymbol{P}_{k}^{-}\right) = \frac{1}{\sqrt{(2\pi)^{n} ||\boldsymbol{P}_{k}^{-}||}} \exp\left\{-\frac{\left[\boldsymbol{\chi}_{k,i}^{+}-\hat{\boldsymbol{x}}_{k}^{-}\right]^{T} \left(\boldsymbol{P}_{k}^{-}\right)^{-1}\left[\boldsymbol{\chi}_{k,i}^{+}-\hat{\boldsymbol{x}}_{k}^{-}\right]}{2}\right\}$$
$$\mathcal{N}\left(\boldsymbol{\chi}_{k,i}^{+}; \hat{\boldsymbol{x}}_{k}^{+}, \boldsymbol{P}_{k}^{+}\right) = \frac{1}{\sqrt{(2\pi)^{n} ||\boldsymbol{P}_{k}^{+}||}} \exp\left\{-\frac{\left[\boldsymbol{\chi}_{k,i}^{+}-\hat{\boldsymbol{x}}_{k}^{+}\right]^{T} \left(\boldsymbol{P}_{k}^{+}\right)^{-1}\left[\boldsymbol{\chi}_{k,i}^{+}-\hat{\boldsymbol{x}}_{k}^{+}\right]}{2}\right\}$$

where *m* denotes the observer dimension which varies during the time and is based on the number of satellites being tracked by the receiver.

The a posteriori mean and covariance estimates are computed as:

$$\hat{\boldsymbol{x}}_{k}^{+} = \sum_{i=1}^{p} \overline{w} \left( \boldsymbol{\chi}_{k,i}^{+} \right) \boldsymbol{\chi}_{k,i}^{+}$$

$$\boldsymbol{P}_{k}^{+} = \sum_{i=1}^{N} \overline{w} \left( \boldsymbol{\chi}_{k,i}^{+} \right) \left[ \boldsymbol{\chi}_{k,i}^{+} - \hat{\boldsymbol{x}}_{k}^{+} \right] \left[ \boldsymbol{\chi}_{k,i}^{+} - \hat{\boldsymbol{x}}_{k}^{+} \right]^{T}$$
(31)

where  $\chi_{k,i}^+ - \hat{x}_k^+$  can be calculated as:

$$\left[\left(\hat{\boldsymbol{p}}_{n,k,i}^{+}-\hat{\boldsymbol{p}}_{n,k}^{+}\right)^{T},\left(\hat{\boldsymbol{\sigma}}_{n,k,i}^{+}-\hat{\boldsymbol{\sigma}}_{n,k}^{+}\right)^{T},\left(\hat{\boldsymbol{\varphi}}_{k,i}^{+}\right)^{T},\left(\hat{\boldsymbol{f}}_{b,k,i}^{bias+}-\hat{\boldsymbol{f}}_{b,k}^{bias+}\right)^{T},\left(\hat{\boldsymbol{\omega}}_{b,k,i}^{bias+}-\hat{\boldsymbol{\omega}}_{b,k}^{bias+}\right)^{T},\left(c\Delta\hat{\boldsymbol{t}}_{k,i}^{+}-c\Delta\hat{\boldsymbol{t}}_{k}^{+}\right)^{T}\right]^{T}.$$

 $\hat{\boldsymbol{\varphi}}_{k,i}^+$  is the rotation vector corresponding to  $\hat{\boldsymbol{q}}_{k,i}^+ \otimes (\hat{\boldsymbol{q}}_k^+)^{-1}$ . Due to the fact that the unit quaternion is not mathematically closed for addition and scalar multiplication, a normalization procedure is necessary to ensure the constraint  $\boldsymbol{q}^T \boldsymbol{q} = 1$  is satisfied in the presence of measurement noise and numerical round-off errors. The normalization can be conducted by replacing  $\hat{\boldsymbol{q}}_k^+$  with  $\hat{\boldsymbol{q}}_k^+/||\hat{\boldsymbol{q}}_k^+||$ .

## 3.3. Residual-Based AUPF

For the a posterior state estimations, a reliable resolution is proposed which greatly depends on prior statistics of the system process and measurement noise. However, these statistics are hardly known exactly in practice because they are based on the types of applications and process dynamics [13]. In addition, the estimation environment regarding INS/GNSS kinematic application is not always fixed but subject to change [14]. In order to maintain the estimation accuracy, a residual-based covariance matching technique is used to adaptively adjust the algorithm parameters and to track the changes in the noise source. For the INS/GNSS integration, the IMU measurements are used in system model, and the product manual specifies the sensor turn-on bias, temperature related variations in biases and noises. In this paper, we assume that the process noise error covariance Q is approximately known, and we focus on the adaptive estimation of sensor measurement error covariance  $\hat{R}$ .

Different from [2], which proposed an adaptive EKF algorithm, for the UKF, modified measurement noise error covariance matrices are adaptively updated as:

$$\hat{R}_{k} = \hat{C}_{v_{k}} + P_{k}^{yy+}$$
, where  $\hat{C}_{v_{k}} = \frac{1}{N} \sum_{j=j_{0}}^{k} z_{j} z_{j}^{T}$  (32)

where  $P_k^{yy+}$  is calculated as:

$$\boldsymbol{P}_{k}^{yy+} = \frac{1}{2n} \sum_{i=1}^{2n} \left[ h_{k} \left( \hat{\boldsymbol{x}}_{k,i}^{+} \right) - \hat{\boldsymbol{y}}_{k}^{+} \right] \left[ h_{k} \left( \hat{\boldsymbol{x}}_{k,i}^{+} \right) - \hat{\boldsymbol{y}}_{k}^{+} \right]^{T}$$
(33)

where  $\hat{y}_{k}^{+} = \frac{1}{2n} \sum_{i=1}^{2n} h_{k} (\hat{x}_{k,i}^{+})$ . The  $\hat{x}_{k,i}^{+}$  denotes the sigma-points drawn from the UKF a posterior estimates. The sigma-points are generated as:

$$\hat{\mathbf{x}}_{k,i}^{+} = \hat{\mathbf{x}}_{k}^{+} + \left(\sqrt{n\mathbf{P}_{k}^{+}}\right)_{i}^{T}, i = 1, \dots, n$$

$$\hat{\mathbf{x}}_{k,i+n}^{+} = \hat{\mathbf{x}}_{k}^{+} - \left(\sqrt{n\mathbf{P}_{k}^{+}}\right)_{i}^{T}, i = 1, \dots, n$$
(34)

$$\hat{\mathbf{x}}_{k,i}^{+} = \begin{bmatrix} \hat{\mathbf{p}}_{n,k}^{+} + \Delta \hat{\mathbf{p}}_{n,k,i}^{+} \\ \hat{\sigma}_{n,k}^{+} + \Delta \hat{\sigma}_{n,k,i}^{+} \\ \hat{\sigma}_{n,k}^{+} + \Delta \hat{\sigma}_{n,k,i}^{+} \\ \delta q \left( \hat{\boldsymbol{\varphi}}_{k,i}^{+} \right) \otimes \hat{q}_{k}^{+} \\ \hat{f}_{b,k}^{bias+} + \Delta \hat{f}_{b,k,i}^{bias+} \\ \hat{\omega}_{b,k}^{bias+} + \Delta \hat{\omega}_{b,k,i}^{bias+} \\ c\Delta \hat{t}_{k}^{+} + \Delta c\Delta \hat{t}_{k,i}^{+} \\ \hat{c} \Delta \hat{t}_{k}^{+} + \Delta c\Delta \hat{t}_{k,i} \end{bmatrix}, \hat{\mathbf{x}}_{k,i+n}^{+} = \begin{bmatrix} \hat{\mathbf{p}}_{n,k}^{+} - \Delta \hat{\mathbf{p}}_{n,k,i}^{+} \\ \hat{\sigma}_{n,k}^{+} - \Delta \hat{\sigma}_{n,k,i}^{+} \\ \delta q \left( -\hat{\boldsymbol{\varphi}}_{k,i}^{+} \right) \otimes \hat{q}_{k}^{+} \\ \hat{f}_{bias}^{bias+} - \Delta \hat{f}_{b,k,i}^{bias+} \\ \hat{\omega}_{b,k}^{bias+} - \Delta \hat{\omega}_{b,k,i}^{bias+} \\ c\Delta \hat{t}_{k}^{+} - \Delta c\Delta \hat{t}_{k,i}^{+} \\ c\Delta \hat{t}_{k}^{+} - \Delta c\Delta \hat{t}_{k,i} \end{bmatrix}$$
(35)

The calculated sigma-points have the form of Equation (35). The delta rotation angle rate in the quaternion form can be formulated as:

$$\delta q\left(\hat{\boldsymbol{\varphi}}_{k,i}^{+}\right) = \begin{bmatrix} \cos\left(\mathbf{0.5} \mid \hat{\boldsymbol{\varphi}}_{k,i}^{+} \mid \right) \\ \sin\left(\mathbf{0.5} \mid \hat{\boldsymbol{\varphi}}_{k,i}^{+} \mid \right) \frac{\hat{\boldsymbol{\varphi}}_{k,i}^{+}}{\mid \mid \hat{\boldsymbol{\varphi}}_{k,i}^{+} \mid \mid} \end{bmatrix}$$
(36)

The flowchart of the proposed AUPF is illustrated in Figure 2.



Figure 2. Adaptive unscented particle filter (AUPF) algorithm flowchart.

# 4. Simulation Results

A simulator with the module of USV trajectory simulation and the raw measurements simulation for IMU and GPS was developed. The trajectory used in this simulation scenario is illustrated in Figure 3. We assume that the USV is operating at sea level. The sliding mode controller in [15] drives the USV fallowing the designed trajectory. To represent a real MEMS-IMU as closely as possible, different error types including the Gaussian model and random bias were induced in the simulated IMU. Raw GPS pseudorange and Doppler measurements were simulated at a 1 Hz rate, while the IMU raw data were simulated at a 100 Hz rate. The sensor error characteristics of MEMS-IMU are given in Table 1. The program is implemented in MATLAB R2013a using Intel Xeon E31270 CPU and 8 GB RAM.

Table 1. Error characteristics of the simulated MEMS-IMU.

Gyroscope (A	Angular Rates)	Accelerometer (Specific Forces)		
Bias in-run Stability Noise (ARW) Scale Factor Error	≤±13 [°/h] (1σ) 0.028 [°/s/ $\sqrt{Hz}$ ] (1σ) <1000 [ppm]	Bias in-run Stability Noise (VRW) Scale Factor Error	≦±1300 [μg] (1σ) 70 [μg/√Hz] (1σ) <1000 [ppm]	
2.0	1. 1.1			

Micro-electromechanical system inertial measurement unit (MEMS-IMU).



Figure 3. The trajectory of the USV.

Figure 3 shows the estimation results of trajectory of USV in horizontal plane. Although the low cost MEMS-IMU is used, the agreement with the actual trajectory is generally good. The estimated position and velocity errors can be found in Figure 4. After the initial transient, the estimation error in the north direction remains less than 3 m approximately, 0.5 m in the east direction and -4.2 m in the down direction; the errors of velocity remain within a region of approximately  $\pm 0.05$  m/s. In Figure 5, the attitudes estimated by the AUPF algorithm (using quaternion) are plotted. The quaternion are converted to their corresponding Euler angles. Except the initial transient, the roll and pitch errors remain mostly in the region  $\pm 0.1$  deg, but the yaw error tends to be a little larger. Accelerometer bias estimation results are illustrated in Figure 6. Gyroscope bias estimation results are plotted in Figure 7. Due to the fact that the yaw angle is the least observable state in practical vehicle motion, we can see form Figure 7 that the bias error on the *z*-axis takes more time to be stable.



Figure 4. (a) estimation errors of the position; (b) estimation errors of the velocity.



**Figure 5.** (a) roll, pitch and yaw of the USV; (b) estimation of the attitude errors (transformed from quaternion to corresponding Euler angles).



Figure 6. Estimation of the accelerometer bias.



**Figure 7.** Estimation of the gyroscope bias.

In order to illustrate the performance of the proposed AUPF algorithm, several groups of tests adopting the AEKF, AUKF, PF and AUPF are conducted, respectively, in this part. For comparing the navigation accuracy among the algorithms mentioned above, we choose the norm of the position ( $||\Delta p||$ ) and velocity ( $||\Delta v||$ ) estimation errors for analysis, which are calculated as  $||\Delta x|| = \sqrt{\Delta x_N^2 + \Delta x_E^2 + \Delta x_D^2}$ . As shown in Figure 8, AUPF presents best estimation accuracy compared with others.



Figure 8. (a) Norm of position estimation errors; (b) Norm of velocity estimation errors.

Furthermore, with the same group of GPS and IMU data, the estimation accuracy and computation time of the algorithms are illustrated in Table 2.  $M(||\Delta p||)$  and  $M(||\Delta v||)$  denote the mean of the norm of the position and velocity estimation errors.  $V(||\Delta p||)$  and  $V(||\Delta v||)$  denote the variance of the norm of the position and velocity estimation errors. The time represents the algorithm processing time, which depends on the computation power of the computer and only for reference. We conduct one simulation run for AEKF and AUKF because the same prior statistical parameters and the same group of sensor data result in unchanged performance. For the PF based algorithm, the particles are generated randomly, so the estimation results have slight differences after each simulation. Table 2 shows the average values of 100 runs of the algorithms. We can see that AUPF (100 particles) can significantly improve the accuracy compared with PF (300 particles), and with computational efficiency at the same time.

Although the validation of the proposed AUPF algorithm is conducted from simulation in this paper, it has a theoretical foundation with a stringent mathematical representation. However, a real data experiment is an intuitively clear way to validate the performance of the presented AUPF algorithm, and it is of great practical interest to validate and compare it with other filters in experiments.

Algorithm	Runs	M(  ∆p  ) [m]	V(  ∆p  ) [m]	M(  ∆v  ) [m/s]	V(  ∆v  ) [m/s]	Time [s]
AEKF	1	5.2451	0.0426	0.0327	0.0042	3.3774
AUKF	1	5.1962	0.0397	0.0286	0.0037	17.3859
PF(300)	100	4.5173	0.0315	0.0207	0.0021	283.4482
AUPF(100)	100	4.2358	0.0273	0.0183	0.0014	62.1538

**Table 2.** Performance comparison between state-of-the-art algorithms.

Adaptive extended Kalman filter (AEKF); Adaptive unscented Kalman filter (AUKF); Particle filter (PF); Adaptive unscented particle filter (AUPF).

## 5. Conclusions

In this paper, we have studied the problem of navigation system design for low-cost USVs, in which we mainly focus on the filtering algorithm. A quaternion-based tightly-coupled integration approach is chosen to fuse the low-cost MEMS-INS measurements and GPS data. Due to the nonlinear and non-Gaussian properties of the INS/GNSS tightly-coupled integration problem, an APUF algorithm is proposed, which combines UKF with PF. A residual-base covariance matching technique is used to adaptively adjust the parameters and to track the changes in observation data in an adaptive manner. The resulting AUPF algorithm can reduce the computational burden without degrading the system estimation accuracy. A USV trajectory simulator is designed to generate the raw senor measurements of IMU and GNSS. The simulation results verify the robustness and accuracy of the AUPF algorithm in comparison to the AEKF and AUKF. In addition, the comparison with PF underlines the computational advantage of AUPF.

**Author Contributions:** Guoqing Xia and Guoqing Wang conceived and designed the experiments; Guoqing Wang performed the experiments; Guoqing Xia and Guoqing Wang analyzed the data; Guoqing Wang wrote the paper.

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

# References

- Manley, J.E. Unmanned surface vehicles, 15 years of development. In Proceedings of the OCEANS 2008, Quebec City, QC, Canada, 15–18 September 2008; pp. 1–4.
- 2. Pourtakdoust, S.H.; Ghanbarpour, A.H. An adaptive unscented Kalman filter for quaternion-based orientation estimation in low-cost AHRS. *Aircr. Eng. Aerosp. Technol.* **2007**, *79*, 485–493. [CrossRef]
- 3. Zhou, J. Low-Cost MEMS-INS/GPS Integration Using Nonlinear Filtering Approaches. Ph.D. Thesis, Universitätsbibliothek der Universität Siegen, Siegen, Germany, April 2013.
- 4. Groves, P.D. *Principles of GNSS, Inertial, and Multisensor Integrated Navigation Systems*; Artech house: Boston, MA, USA, 2013.
- 5. Julier, S.J.; Uhlmann, J.K.; Durrant-Whyte, H.F. A new approach for filtering nonlinear systems. In Proceedings of the 1995 American Control Conference, Seattle, WA, USA, 21–23 June 1995; pp. 1628–1632.
- Simon, D. Optimal State Estimation: Kalman, H Infinity, and Nonlinear Approaches; John Wiley & Sons: Hoboken, NJ, USA, 2006.
- 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]
- 8. Haug, A.J. A Tutorial on Bayesian Estimation and Tracking Techniques Applicable to Nonlinear and Non-Gaussian *Processes*; MITRE Corporation: McLean, VA, USA, 2005.
- 9. Mohamed, A.H.; Schwarz, K.P. Adaptive Kalman filtering for INS/GPS. J. Geod. 1999, 73, 193–203. [CrossRef]
- 10. Fossen, T.I. Nonlinear Modelling and Control of Underwater Vehicles. Ph.D. Thesis, Faculty of Information Technology, Norwegian University of Science and Technology, Trondheim, Norway, 1991.
- Xia, G.; Wang, G.; Chen, X.; Zhao, A.; Pang, C. DRFNN adaptive observer based sliding mode tracking control of an underactuated surface vehicle. Mechatronics and Automation. In Proceedings of the 2015 IEEE International Conference on Mechatronics and Automation (ICMA), Beijing, China, 2–5 August 2015; pp. 2255–2260.

- 12. Misra, P.; Enge, P. *Global Positioning System: Signals, Measurements and Performance,* 2nd ed.; Ganga-Jamuna Press: Lincoln, MA, USA, 2006.
- 13. Bhatti, U.I. Improved Integrity Algorithms for Integrated GPS/INS Systems in the Presence of Slowly Growing Errors. Ph.D. Thesis, Department of Civil and Environmental Engineering, Imperial College London, London, UK, March 2007.
- Hide, C.; Moore, T.; Smith, M. Adaptive Kalman filtering algorithms for integrating GPS and low cost INS. In Proceedings of the Position Location and Navigation Symposium (PLANS 2004), Monterey, CA, USA, 26–29 April 2004; pp. 227–233.
- 15. Hide, C.; Moore, T.; Smith, M. Adaptive Kalman filtering for low-cost INS/GPS. J. Navig. 2003, 56, 143–152. [CrossRef]



© 2016 by the authors; licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC-BY) license (http://creativecommons.org/licenses/by/4.0/).