INS/GPS Tightly-coupled Integration using Adaptive Unscented Particle Filter

Junchuan Zhou[1], Stefan Knedlik[2] and Otmar Loffeld[1]

[1](*University of Siegen, Center for Sensorsystems (ZESS), Germany*)

[2](*iMAR GmbH, St. Ingbert, Germany*)
(Email: Zhou@zess.uni-siegen.de)

With the rapid developments in computer technology, the particle filter (PF) is becoming more attractive in navigation applications. However, its large computational burden still limits its widespread use. One approach for reducing the computational burden without degrading the system estimation accuracy is to combine the PF with other filters, i.e., the extended Kalman filter (EKF) or the unscented Kalman filter (UKF). In this paper, the *a posteriori* estimates from an adaptive unscented Kalman filter (AUKF) are used to specify the PF importance density function for generating particles. Unlike the sequential importance sampling re-sampling (SISR) PF, the re-sampling step is not required in the algorithm, because the filter does not reuse the particles. Hence, the filter computational complexity can be reduced. Besides, the latest measurements are used to improve the proposal distribution for generating particles more intelligently. Simulations are conducted on the basis of a field-collected 3D UAV trajectory. GPS and IMU data are simulated under the assumption that a NovAtel DL-4plus GPS receiver and a Landmark[TM] 20 MEMS-based IMU are used. Navigation under benign and highly reflective signal environments are considered. Monte Carlo experiments are made. Numerical results show that the AUPF with 100 particles can present improved system estimation accuracy with an affordable computational burden when compared with the AEKF and AUKF algorithms.

KEY WORDS

1. INS/GPS integration.   2. Adaptive Kalman filtering.   3. Unscented particle filter.

1. INTRODUCTION.   The integration of an inertial navigation system (INS) and the global positioning system (GPS) can provide overall improved navigation performance due to their complementary characteristics. With the emergence of the micro-electromechanical system (MEMS) based low-cost strapdown inertial measurement unit (IMU), cost-effective INS/GPS integrated navigation systems become a reality (EI-Sheimy and Niu, 2007). The INS/GPS integration is often referred to the data fusion of measurements obtained from a GPS receiver antenna and an INS and formulated as a state space estimation task where the Kalman filter (KF) or a modified version of it is applied to obtain the (sub)-optimal solutions. Nevertheless, the multi-antenna GPS system using double difference carrier

phase measurements is highly recommended for the integration due to its redundant attitude information derived from the GPS side (e.g., Knedlik et al. (2009)). The INS/GPS integration can be briefly classified as loosely-coupled, tightly-coupled and deeply-coupled. For the loosely-coupled integration, the implementation is simpler with a lower real-time computational burden. However, in that case, one local KF is used for GPS and one KF for integration, so cascaded filtering problems may occur which must be carefully accounted for (e.g., Groves, 2008). Besides, when less than four satellites are tracked, the INS predicted estimates cannot be updated from the available GPS measurements. These problems can be easily solved in the tightly-coupled integration because in such a more centralized Kalman filtering scheme, the GPS pseudorange and delta pseudorange (or Doppler) measurements are used as observations to update the navigation filter. The INS estimated solution can be continuously updated by the measurements from the remaining satellites, even if the number of the tracked satellites drops below four. However, the system observation model turns out to be non-linear. As the core of the integrated navigation system, this non-linearity should be carefully treated in the design of the integration KF. Deeply-coupled integration requires access to the GPS receiver tracking loops, which is not given for the majority of commercial GPS receivers. Thus, it is not in the scope of this paper.

Regarding the INS/GPS tightly-coupled integration, the EKF is considered as state-of-the-art for fusing the INS and GPS data. The principle of the EKF is to linearize the non-linear system and observation models around the updated dynamic trajectory using a Taylor series expansion. It relies on the first order linearization of the non-linear models with the assumption of Gaussian noise. In certain circumstances, the EKF may not be able to present reliable system estimation solutions. The UKF was first proposed by Julier et al. (1995) as another non-linear filter, which propagates a fixed number of sigma points through the non-linear system and observation models to capture the system *a posteriori* mean and covariance estimates. Its estimation accuracy can reach to the third order of the Taylor series expansion. Nevertheless, the algorithm still treats the noise as Gaussian distributed. The PF neither makes assumptions on the linearity of the system models, nor on the characteristics of the noise. Therefore, it is considered as a non-parametric estimation method for non-linear/non-Gaussian applications. With the recent advances in computer technology, it turns out to be more attractive for navigation applications (Gordon et al., 1993). However, its large computational burden is still its main obstacle in practical use. Moreover, the system performance degeneracy and sample impoverishments (Arulampalam et al., 2002) are other main implementation issues of the basic SISR PF algorithm. To overcome these problems, some engineers propose to combine the PF with the EKF or UKF algorithm to form the extended particle filter (EPF) or the unscented particle filter (UPF) (Haug, 2005; Simon, 2006). The method mentioned by Simon is to use a bank of EKFs or UKFs (one for each particle) with the likelihood function to derive the system *a posteriori* estimates. In the proposal from Haug, only one EKF or UKF is required. Its *a posteriori* mean and covariance estimates are used to specify the PF importance density function to generate particles more intelligently. After calculating the normalized importance weights of the particles, the system *a posteriori* estimates are refined. This method does not require the re-sampling step, because the filter does not reuse the particles. Such an implementation is a variant of the so-called Gaussian

particle filter, which is asymptotically optimal in the number of particles when the additive Gaussian noise assumption holds true. However, this Gaussian assumption can be relaxed and in general be non-Gaussian in the case that the importance density distribution can be approximated as Gaussian and the analytical expression for the likelihood function can be specified (Kotecha and Djuric, 2003). Both methods present highly accurate estimation solutions with a reduced number of particles as compared to the SISR PF algorithm. However, the method proposed by Haug is more attractive, because only one EKF or UKF is used in the algorithm. And hence, it is considered in this paper.

In order to obtain the accurate *a posteriori* estimates from the EKF or UKF algorithm to build the importance density function for generating particles more intelligently, the prior statistical knowledge on the system process and measurement noises must be known (Hide et al., 2003; Mehra, 1971; Mohamed and Schwarz, 1999). Insufficient or incorrect knowledge about these statistics may lead to the degradation of the system performance, or even provoke the filter to diverge. In this paper, the residual-based covariance matching technique is used. Accordingly, the adaptive unscented particle filter (AUPF) algorithm is proposed.

In the remainder of this paper, the content is organized as follows. In Section 2, the UKF and the UPF algorithms are given. In Section 3, the innovation-based and its near relative, the residual-based adaptive Kalman filtering techniques are presented. The flowchart of the proposed AUPF algorithm is shown in Section 4. In Section 5, the mathematical models developed for the low-cost INS/GPS tightly-coupled integration are specified. The experimental setup is introduced in Section 6, and in Section 7, simulation results are compared and analyzed to illustrate the concept of the proposed AUPF algorithm. Last but not least, Section 8 presents the summary for this paper.

2. UNSCENTED PARTICLE FILTER.

2.1. *Unscented Kalman Filter*. The idea behind the UKF comes from the assumption that it is much easier to approximate a Gaussian distribution, rather than to simulate any arbitrary non-linear functions (Julier et al., 1995, 2000). Consider the case that the system process and measurement noises are additive and Gaussian distributed, the UKF algorithm from (e.g., Simon, 2006) is summarized in Table 1.

In Table 1, n denotes the dimension of the system state vector. $\left(\sqrt{n\mathbf{P}_{k-1}^{+}}\right)$ is the matrix square root of $n\mathbf{P}_{k-1}^{+}$, such as $\left(\sqrt{n\mathbf{P}_{k-1}^{+}}\right)^{T}\left(\sqrt{n\mathbf{P}_{k-1}^{+}}\right)=n\mathbf{P}_{k-1}^{+}$, which can be obtained from the Cholesky factorization 'CHOL' in MATLAB. $\left(\sqrt{n\mathbf{P}_{k-1}^{+}}\right)_{i}$ is the i-th row of $\left(\sqrt{n\mathbf{P}_{k-1}^{+}}\right)$. $\mathbf{Q}_{k}$ is the covariance matrix of the additive system process noise. $\mathbf{R}_{k}$ is the covariance matrix of the additive measurement noise.

2.2. *Unscented Particle Filter*. In the proposed UPF algorithm, the first two moments of the UKF estimates are used to form the importance density function $\mathcal{N}(\hat{\mathbf{x}}_{k}^{+}, \mathbf{P}_{k}^{+})$ for generating particles. The first 7 steps of the UPF algorithm are the same as those of the UKF. Only the extra steps are listed in Table 2. As shown in the Table 1 and 2 equations, the particles of the UPF are generated from the importance density function formed by the UKF *a posteriori* estimates. The importance weights of particles are then computed and normalized through the likelihood function. The refined *a posteriori* mean and covariance estimates are calculated in step 11, which are

Table 1. Summary of the UKF algorithm.

Stochastic discrete-time system state space models	$\mathbf{x}_k = \mathbf{f}_{k-1}(\mathbf{x}_{k-1}) + \mathbf{w}_{k-1}$ $\mathbf{y}_k = \mathbf{h}_k(\mathbf{x}_k) + \mathbf{v}_k$ where $\mathbf{w}_k \sim \mathcal{N}(0, \mathbf{Q}_k)$ and $\mathbf{v}_k \sim \mathcal{N}(0, \mathbf{R}_k)$
1) Initialization	$\hat{\mathbf{x}}_0^+ = E(\mathbf{x}_0), \; \mathbf{P}_0^+ = E[(\mathbf{x}_0 - \hat{\mathbf{x}}_0^+)(\mathbf{x}_0 - \hat{\mathbf{x}}_0^+)^T]$
2) Draw sigma points	$\hat{\mathbf{x}}_{k-1,i}^+ = \hat{\mathbf{x}}_{k-1}^+ + \left(\sqrt{n\mathbf{P}_{k-1}^+}\right)_i^T, \; \hat{\mathbf{x}}_{k-1,i+n}^+ = \hat{\mathbf{x}}_{k-1}^+ - \left(\sqrt{n\mathbf{P}_{k-1}^+}\right)_i^T, \; i = 1, ..., n$
3) Time update	$\hat{\mathbf{x}}_k^- = \dfrac{1}{2n} \sum\limits_{i=1}^{2n} \mathbf{f}_{k-1}(\hat{\mathbf{x}}_{k-1,i}^+),$
	$\mathbf{P}_k^- = \dfrac{1}{2n} \sum\limits_{i=1}^{2n} \left[\mathbf{f}_{k-1}(\hat{\mathbf{x}}_{k-1,i}^+) - \hat{\mathbf{x}}_k^- \right] \left[\mathbf{f}_{k-1}(\hat{\mathbf{x}}_{k-1,i}^+) - \hat{\mathbf{x}}_k^- \right]^T + \mathbf{Q}_{k-1}$
4) Draw sigma points	$\hat{\mathbf{x}}_{k,i}^- = \hat{\mathbf{x}}_k^- + \left(\sqrt{n\mathbf{P}_k^-}\right)_i^T, \; \hat{\mathbf{x}}_{k,i+n}^- = \hat{\mathbf{x}}_k^- - \left(\sqrt{n\mathbf{P}_k^-}\right)_i^T, \; i = 1, ..., n$
5) Predict measurements	$\hat{\mathbf{y}}_k = \dfrac{1}{2n} \sum\limits_{i=1}^{2n} \mathbf{h}_k(\hat{\mathbf{x}}_{k,i}^-)$
6) Update covariance	$\mathbf{P}_k^{yy} = \dfrac{1}{2n} \sum\limits_{i=1}^{2n} \left[\mathbf{h}_k(\hat{\mathbf{x}}_{k,i}^-) - \hat{\mathbf{y}}_k \right] \left[\mathbf{h}_k(\hat{\mathbf{x}}_{k,i}^-) - \hat{\mathbf{y}}_k \right]^T + \mathbf{R}_k$
	$\mathbf{P}_k^{xy} = \dfrac{1}{2n} \sum\limits_{i=1}^{2n} \left[\hat{\mathbf{x}}_{k,i}^- - \hat{\mathbf{x}}_k^- \right] \left[\mathbf{h}_k(\hat{\mathbf{x}}_{k,i}^-) - \hat{\mathbf{y}}_k \right]^T$
7) Measurement update	$\mathbf{K}_k = \mathbf{P}_k^{xy}(\mathbf{P}_k^{yy})^{-1}, \; \hat{\mathbf{x}}_k^+ = \hat{\mathbf{x}}_k^- + \mathbf{K}_k(\tilde{\mathbf{y}}_k - \hat{\mathbf{y}}_k), \; \mathbf{P}_k^+ = \mathbf{P}_k^- - \mathbf{K}_k \mathbf{P}_k^{yy} \mathbf{K}_k^T$

Table 2. Additional steps in the UPF algorithm.

8) Draw particles	$\hat{\mathbf{x}}_{k,i}^+ \sim \mathcal{N}(\hat{\mathbf{x}}_k^+, \mathbf{P}_k^+), \; i = 1...n$
9) Calculate weights	$\mathrm{w}(\hat{\mathbf{x}}_{k,i}^+) = \dfrac{p(\tilde{\mathbf{y}}_k \vert \hat{\mathbf{x}}_{k,i}^+) \mathcal{N}(\hat{\mathbf{x}}_{k,i}^+ \vert \hat{\mathbf{x}}_k^-, \mathbf{P}_k^-)}{\mathcal{N}(\hat{\mathbf{x}}_{k,i}^+ \vert \hat{\mathbf{x}}_k^+, \mathbf{P}_k^+)}$
10) Normalize the weights	$\bar{w}(\hat{\mathbf{x}}_{k,i}^+) = \dfrac{\mathrm{w}(\hat{\mathbf{x}}_{k,i}^+)}{\sum_{i=1}^{N} \mathrm{w}(\hat{\mathbf{x}}_{k,i}^+)}, \; \text{where} \; \sum\limits_{i=1}^{N} \bar{w}(\hat{\mathbf{x}}_{k,i}^+) = 1$
11) Update estimates	$\hat{\mathbf{x}}_k^+ = \sum\limits_{i=1}^{N} \bar{w}(\hat{\mathbf{x}}_{k,i}^+)\hat{\mathbf{x}}_{k,i}^+, \; \mathbf{P}_k^+ = \sum\limits_{i=1}^{N} \bar{w}(\hat{\mathbf{x}}_{k,i}^+)\left[\hat{\mathbf{x}}_{k,i}^+ - \hat{\mathbf{x}}_k^+ \right]\left[\hat{\mathbf{x}}_{k,i}^+ - \hat{\mathbf{x}}_k^+ \right]^T$

recursively used to generate sigma points in the next epoch (step 2). The whole process repeats until the end of the trajectory.

3. ADAPTIVE ESTIMATION METHODS. The Kalman filtering methods can provide reliable *a posteriori* state estimates only when the prior statistics of the system process and measurement noises are known. In practice, these error statistics are difficult to know exactly, because they are strongly related to the type of application at hand and may infrequently be non-Gaussian and usually be non-stationary due to the changing of the physical environment. In order to obtain the accurate system state estimates, prompt reflection of changes from external influences in noise statistics is required. In this paper, the residual-based covariance

matching technique is used in the algorithm to monitor and track the changes in the noise sources and to adaptively adjust its parameters to maintain the system estimation accuracy.

3.1. *Innovation-based Adaptive Kalman Filtering.* The innovation sequence in the KF is the difference between the incoming measurements and the *a priori* estimated measurements. It represents the additional information available to the filter as a consequence of the incoming new measurements. Hence, it is considered as the most relevant information for the filter adaptation (Mehra, 1971), which is computed as:

$$\mathbf{z}_k = \tilde{\mathbf{y}}_k - \hat{\mathbf{y}}_k, \text{ where } \hat{\mathbf{y}}_k = \mathbf{h}_k(\hat{\mathbf{x}}_k^-). \tag{1}$$

As derived in Mohamed and Schwarz (1999), the system process and measurement noise error covariance matrices are adaptively updated as:

$$\hat{\mathbf{R}}_k = \hat{\mathbf{C}}_{v_k} - \mathbf{H}_k \mathbf{P}_k^- \mathbf{H}_k^T, \text{ and } \hat{\mathbf{Q}}_k = \mathbf{K}_k \hat{\mathbf{C}}_{v_k} \mathbf{K}_k^T, \tag{2}$$

where $\hat{\mathbf{C}}_{v_k}$ in Equation (2) is the estimated innovation covariance matrix, which is computed as:

$$\hat{\mathbf{C}}_{v_k} = \frac{1}{N} \sum_{j=j_0}^{k} \mathbf{z}_j \mathbf{z}_j^T, \tag{3}$$

where N is the window size (a fixed number of epochs); $\mathbf{z}_j$ is the innovation sequence at time instant j; $j_0 = k - N + 1$ is the first epoch of the estimation.

3.2. *Residual-based Adaptive Kalman Filtering.* For the innovation-based adaptive filtering method, the subtraction operation in Equation (2) may cause numerical problems in the calculation of the measurement error covariance matrix $\hat{\mathbf{R}}_k$. That is, if there are negative parameters in the main diagonal of $\hat{\mathbf{R}}_k$, the filter will suddenly diverge. This problem can be easily tackled using the residual-based adaptive filtering method. The residual sequence is computed as the difference between the incoming measurements and the *a posteriori* estimated measurements, as shown in Equation (4).

$$\mathbf{z}_k = \tilde{\mathbf{y}}_k - \hat{\mathbf{y}}_k^+, \text{ where } \hat{\mathbf{y}}_k^+ = \mathbf{h}_k(\hat{\mathbf{x}}_k^+). \tag{4}$$

The $\hat{\mathbf{R}}_k$ is calculated by Equation (5), which guarantees the positive definite of the matrix. The detail derivation of this equation can be found in Mohamed and Schwarz (1999), and it is omitted here.

$$\hat{\mathbf{R}}_k = \hat{\mathbf{C}}_{v_k} + \mathbf{H}_k \mathbf{P}_k^+ \mathbf{H}_k^T, \text{ where } \hat{\mathbf{C}}_{v_k} = \frac{1}{N} \sum_{j=j_0}^{k} \mathbf{z}_j \mathbf{z}_j^T, \tag{5}$$

where $\hat{\mathbf{C}}_{v_k}$ is the residual sequence covariance matrix at time instant k.

In Equation (5), the computation of $\hat{\mathbf{R}}_k$ is on the basis of the *a posteriori* state error covariance matrix $\mathbf{P}_k^+$. The EKF measurement update equations are given here for convenience:

$$\begin{aligned} \mathbf{K}_k &= \mathbf{P}_k^- \mathbf{H}_k^T (\mathbf{H}_k \mathbf{P}_k^- \mathbf{H}_k^T + \mathbf{R}_k)^{-1} \\ \hat{\mathbf{x}}_k^+ &= \hat{\mathbf{x}}_k^- + \mathbf{K}_k(\tilde{\mathbf{y}}_k - \hat{\mathbf{y}}_k) \\ \mathbf{P}_k^+ &= (\mathbf{I} - \mathbf{K}_k \mathbf{H}_k)\mathbf{P}_k^-(\mathbf{I} - \mathbf{K}_k \mathbf{H}_k)^T + \mathbf{K}_k \mathbf{R}_k \mathbf{K}_k^T. \end{aligned} \tag{6}$$

As shown above, the derivation of $\mathbf{P}_k^+$ contains the measurement error covariance matrix $\mathbf{R}_k$ at time instant k. In order to estimate $\hat{\mathbf{R}}_k$ using Equation (5), a pre-defined error covariance $\mathbf{R}_k$ must be known. This problem can be solved by the recursive EKF algorithm. Namely, the $\hat{\mathbf{R}}_{k-1}$ can be used to approximate the error covariance $\mathbf{R}_k$ at time instant k. This approximation is not restricting. That is, the $\hat{\mathbf{R}}_k$ is varying slowly over time. It is calculated on the basis of $\hat{\mathbf{C}}_{v_k}$, which comes from the average of the accumulated residual sequence covariance over a sliding window size.

4. ADAPTIVE UNSCENTED PARTICLE FILTER. For the INS/GPS integrated navigation system, the IMU measurements are normally used in the system model, and their error statistics, i.e., the sensor turn-on biases, temperature related variations in biases and noises, can be found in the product specifications, and most of them can be detected through the self-alignment and calibration processes. The residual accumulated sensor errors will be estimated and corrected by the GPS data on the frequency of GPS measurement update rate. Therefore, the parameters in the system error covariance matrix $\mathbf{Q}$ can be approximately given, especially for short-time applications, for instance the 68 s UAV trajectory used in this paper. Nonetheless, the GPS measurements used in the system observation model are strongly related to the external influences. When the UAV is manoeuvring under highly reflective signal environments, the error statistics may vary dramatically. In this paper, we focus on the adaptive estimation of the measurement error covariance $\hat{\mathbf{R}}$ under the assumption that the statistical information about the process noise error covariance $\mathbf{Q}$ is approximately known.

Equation (5) is derived based on the EKF algorithm. For the UKF, a modification is required:

$$\hat{\mathbf{R}}_k = \hat{\mathbf{C}}_{v_k} + \mathbf{P}_k^{yy+}, \quad \text{where} \quad \hat{\mathbf{C}}_{v_k} = \frac{1}{N} \sum_{j=j_0}^{k} \mathbf{z}_j \mathbf{z}_j^T, \tag{7}$$

where $\mathbf{P}_k^{yy+}$ is calculated as:

$$\mathbf{P}_k^{yy+} = \frac{1}{2n} \sum_{i=1}^{2n} \left[\mathbf{h}_k(\hat{\mathbf{x}}_{k,i}^+) - \hat{\mathbf{y}}_k^+ \right] \left[\mathbf{h}_k(\hat{\mathbf{x}}_{k,i}^+) - \hat{\mathbf{y}}_k^+ \right]^T, \quad \text{where} \quad \hat{\mathbf{y}}_k^+ = \frac{1}{2n} \sum_{i=1}^{2n} \mathbf{h}_k(\hat{\mathbf{x}}_{k,i}^+). \tag{8}$$

The $\hat{\mathbf{x}}_{k,i}^+$ represent the sigma points drawn from the UKF *a posteriori* estimates, i.e., $\hat{\mathbf{x}}_k^+$, $\mathbf{P}_k^+$, as:

$$\hat{\mathbf{x}}_{k,i}^+ = \hat{\mathbf{x}}_k^+ + \left(\sqrt{n\mathbf{P}_k^+} \right)_i^T, \quad \hat{\mathbf{x}}_{k,i+n}^+ = \hat{\mathbf{x}}_k^+ - \left(\sqrt{n\mathbf{P}_k^+} \right)_i^T, \quad i = 1, \ldots, n. \tag{9}$$

Figure 1 shows the flowchart of the proposed AUPF algorithm.

5. TIGHTLY-COUPLED INS/GPS INTEGRATED KALMAN FILTER.

5.1. *Inertial Mechanization and INS Error Modelling.* In the state-of-the-art INS/GPS integrated navigation systems, the INS error propagation model is widely used as the system model (Farrell and Barth, 1999; Li et al., 2008). The INS navigation solution errors and IMU sensor errors are commonly chosen to form the system state vector. For the low-cost MEMS-based IMU, the sensor errors have the

Initialize $\hat{\mathbf{x}}_0^+, \mathbf{P}_0^+, \hat{\mathbf{R}}_0, \mathbf{Q}_0$

$\hat{\mathbf{x}}_{k-1}^+, \mathbf{P}_{k-1}^+, \hat{\mathbf{R}}_{k-1}$

Generate sigma points:

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

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

Delay $\hat{\mathbf{R}}_k$

Propagate ahead state estimates:

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

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

Refine a posterior state estimates:

$$\hat{\mathbf{x}}_k^+ = \sum_{i=1}^{N}\overline{\mathrm{w}}(\hat{\mathbf{x}}_{k,i}^+)\hat{\mathbf{x}}_{k,i}^+$$

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

Generate sigma points:

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

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

$\hat{\mathbf{x}}_k^-, \mathbf{P}_k^-$

Update and normalize the particle importance weights:

$$\mathrm{w}(\hat{\mathbf{x}}_{k,i}^+) = \frac{p(\tilde{\mathbf{y}}_k \mid \hat{\mathbf{x}}_{k,i}^+)\mathcal{N}(\hat{\mathbf{x}}_{k,i}^+ \mid \hat{\mathbf{x}}_k^-, \mathbf{P}_k^-)}{\mathcal{N}(\hat{\mathbf{x}}_{k,i}^+ \mid \hat{\mathbf{x}}_k^+, \mathbf{P}_k^+)}$$

$$\overline{\mathrm{w}}(\hat{\mathbf{x}}_{k,i}^+) = \frac{\mathrm{w}(\hat{\mathbf{x}}_{k,i}^+)}{\sum_{i=1}^{N}\mathrm{w}(\hat{\mathbf{x}}_{k,i}^+)}, with \sum_{i=1}^{N}\overline{\mathrm{w}}(\hat{\mathbf{x}}_{k,i}^+) = 1$$

Predict measurements and covariance matrices:

$$\hat{\mathbf{y}}_k = \frac{1}{2n}\sum_{i=1}^{2n}\mathbf{h}_k(\hat{\mathbf{x}}_{k,i}^-)$$

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

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

Importance sampling:

$$\hat{\mathbf{x}}_{k,i}^+ \sim \mathcal{N}(\hat{\mathbf{x}}_k^+, \mathbf{P}_k^+)$$

$\hat{\mathbf{x}}_k^+, \mathbf{P}_k^+$

$Window\ Size = N$

Kalman filter:

$$\mathbf{K}_k = \mathbf{P}_k^{xy}(\mathbf{P}_k^{yy})^{-1}$$

$$\hat{\mathbf{x}}_k^+ = \hat{\mathbf{x}}_k^- + \mathbf{K}_k(\tilde{\mathbf{y}}_k - \hat{\mathbf{y}}_k)$$

$$\mathbf{P}_k^+ = \mathbf{P}_k^- - \mathbf{K}_k\mathbf{P}_k^{yy}\mathbf{K}_k^T$$

$\hat{\mathbf{x}}_k^+$

Residual sequence:

$$\mathbf{z}_k = \tilde{\mathbf{y}}_k - \mathbf{h}_k(\hat{\mathbf{x}}_k^+)$$

Estimate measurement error covariance:

$$\hat{\mathbf{C}}_{v_k} = \frac{1}{N}\sum_{j=j_0}^{k}\mathbf{z}_j\mathbf{z}_j^T$$

$$\hat{\mathbf{R}}_k = \hat{\mathbf{C}}_{v_k} + \mathbf{P}_k^{yy+}$$

$\hat{\mathbf{x}}_k^+, \mathbf{P}_k^+$

$\mathbf{P}_k^{yy+}$

Generate a posteriori sigma points:

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

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

Update measurements and covariance matrix:

$$\hat{\mathbf{y}}_k^+ = \frac{1}{2n}\sum_{i=1}^{2n}\mathbf{h}_k(\hat{\mathbf{x}}_{k,i}^+)$$

$$\mathbf{P}_k^{yy+} = \frac{1}{2n}\sum_{i=1}^{2n}\left[\mathbf{h}_k(\hat{\mathbf{x}}_{k,i}^+) - \hat{\mathbf{y}}_k^+\right]\left[\mathbf{h}_k(\hat{\mathbf{x}}_{k,i}^+) - \hat{\mathbf{y}}_k^+\right]^T$$
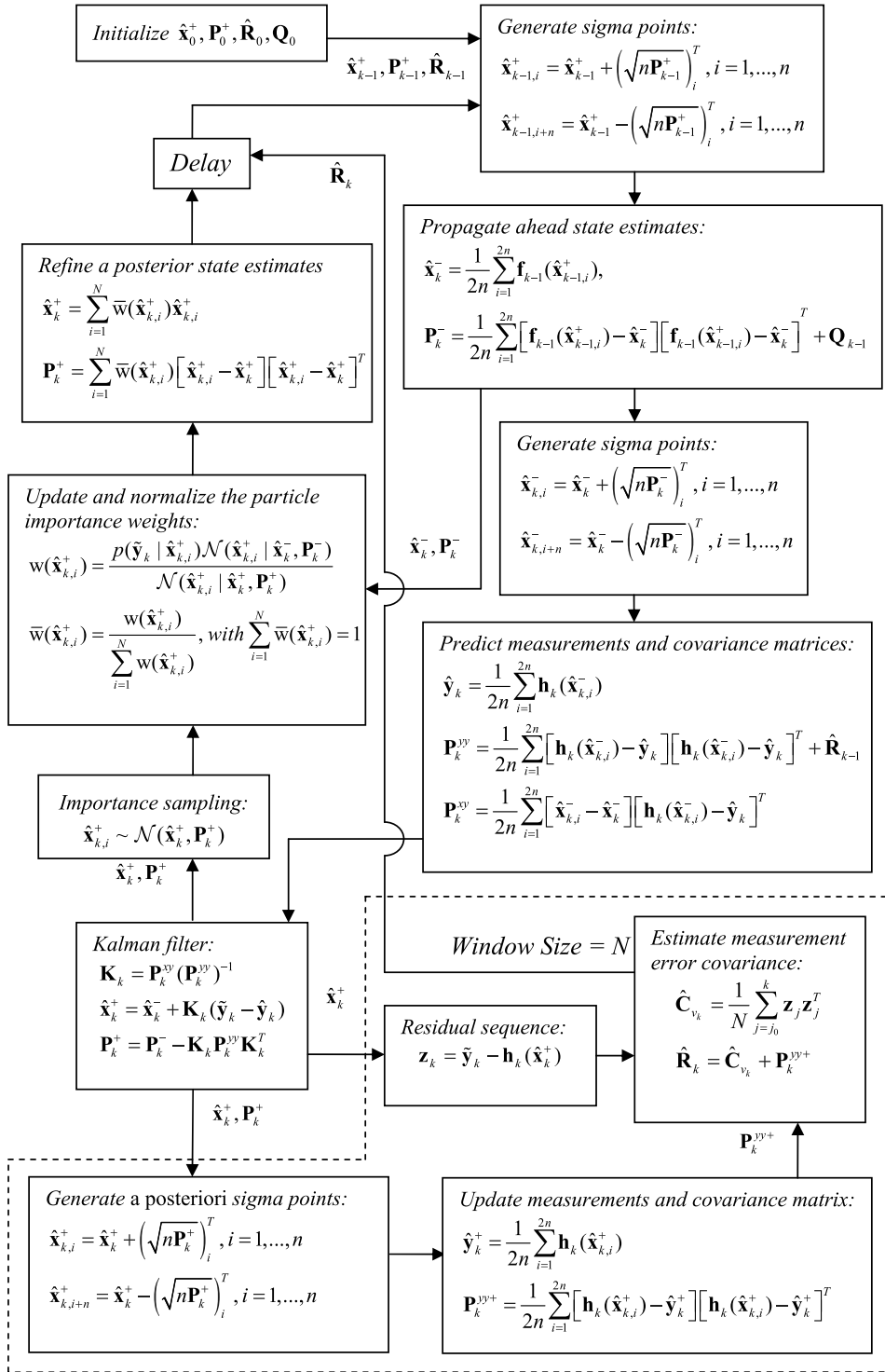
Figure 1. AUPF algorithm flowchart.

non-Gaussian characteristics and must be modelled. For instance, the bias errors in the gyroscope measurements will be integrated to form the Euler angles, which are used to build the vector transformation matrix for transforming the accelerometer measurements from the body frame to the navigation frame. After integrations, these errors turn out to be the position and velocity drift in the navigation frame.

For the EKF, the usage of error states or total states presents similar system estimation accuracy in INS/GPS tightly-coupled integration. The mutual derivation between the total states and error states can be found in Brown and Hwang (1997). However, for the UKF, using the total states with non-linear system model, the system strapdown processing must be repeated on every sigma point at each IMU data output instant. This will introduce significantly increased computational burden. Therefore, in this paper, the error states are used, as shown in Equation (10). They represent the difference between the estimated values and true values, i.e., $\delta\mathbf{x} = \hat{\mathbf{x}}_{estimated} - \mathbf{x}_{true}$.

$$\delta\mathbf{x} = \begin{cases} \delta\mathbf{p}_n = [\delta x_n \ \delta x_e \ \delta x_d]^T & (Position\ errors\ in\ NED\ frame) \\ \delta\mathbf{v}_n = [\delta\dot{x}_n \ \delta\dot{x}_e \ \delta\dot{x}_d]^T & (Velocity\ errors\ in\ NED\ frame) \\ \delta\boldsymbol{\psi} = [\delta\phi \ \delta\theta \ \delta\varphi]^T & (Attitude\ errors) \\ \delta\mathbf{f}_b = [\delta f_x \ \delta f_y \ \delta f_z]^T & (Accel\ bias\ errors\ in\ body\ frame) \\ \delta\boldsymbol{\omega}_b = [\delta\omega_x \ \delta\omega_y \ \delta\omega_z]^T & (Gyro\ bias\ errors\ in\ body\ frame) \\ c\delta t & (Receiver\ clock\ bias\ error\ in\ distance) \\ c\delta\dot{t} & (Receiver\ clock\ drift\ error\ in\ distance) \end{cases} \quad (10)$$

The INS mechanization we use is that which is often seen for low-cost IMU. A low-cost IMU is insensitive to the Earth rotation. Moreover, in the strapdown processing and in the system model for the KF, the transport rate, coriolis terms and the lever-arm effects are neglected for simplicity. The simplified mechanization model can then be expressed in NED frame as:

$$\begin{aligned} \mathbf{p}_{n,k+1} &= \mathbf{p}_{n,k} + \mathbf{v}_{n,k} \cdot \Delta t \\ \mathbf{v}_{n,k+1} &= \mathbf{v}_{n,k} + [\mathbf{R}_{b2n,k} \cdot \mathbf{f}_{b,k} + \mathbf{g}_n] \cdot \Delta t \\ \boldsymbol{\psi}_{k+1} &= \boldsymbol{\psi}_k + \mathbf{E}_{b2n,k} \cdot \boldsymbol{\omega}_{b,k} \cdot \Delta t \end{aligned} \quad (11)$$

where $\mathbf{f}_b$ is measurement vector of specific force; $\boldsymbol{\omega}_b$ represents the angular rate vector; $\mathbf{g}_n$ is the acceleration vector due to the local gravity; $\boldsymbol{\psi}$ contains the Euler angles; $\mathbf{R}_{b2n}$ is the rotation matrix from body frame to NED navigation frame; $\mathbf{E}_{b2n}$ is the rotation rate transformation matrix between the body and navigation frame, as shown in Equation (12).

$$\mathbf{R}_{b2n} = \begin{pmatrix} c(\varphi)c(\theta) & c(\varphi)s(\theta)s(\phi) - s(\varphi)c(\phi) & c(\varphi)s(\theta)c(\phi) + s(\varphi)s(\phi) \\ s(\varphi)c(\theta) & s(\varphi)s(\theta)s(\phi) + c(\varphi)c(\phi) & s(\varphi)s(\theta)c(\phi) - c(\varphi)s(\phi) \\ -s(\theta) & c(\theta)s(\phi) & c(\theta)c(\phi) \end{pmatrix}$$

$$(12)$$

$$\mathbf{E}_{b2n} = \begin{pmatrix} 1 & s(\phi)t(\theta) & c(\phi)t(\theta) \\ 0 & c(\phi) & -s(\phi) \\ 0 & s(\phi)/c(\theta) & c(\phi)/c(\theta) \end{pmatrix},$$

where $c(x)$, $s(x)$ and $t(x)$ represent the mathematical operations of $cos(x)$, $sin(x)$ and $tan(x)$ respectively.

The simplified linear INS error model is given in Equation (13). Its derivation can be found in Titterton and Weston (2004) and Sukkarieh (2000). It is worth noting that this model uses approximations. It is not the exact linearized model from Equation (5). To use this model, the IMU self-alignment and calibration processes must be carefully conducted before the experiment starts.

$$
\begin{bmatrix} \delta\dot{\mathbf{p}}_n \\ \delta\dot{\mathbf{v}}_n \\ \delta\dot{\boldsymbol{\psi}} \\ \delta\dot{\mathbf{f}}_b \\ \delta\dot{\boldsymbol{\omega}}_b \\ c\delta\dot{t} \\ c\delta\ddot{t} \end{bmatrix} = \underbrace{\begin{bmatrix} \mathbf{O}_{3\times3} & \mathbf{I}_{3\times3} & \mathbf{O}_{3\times3} & \mathbf{O}_{3\times3} & \mathbf{O}_{3\times3} & \mathbf{0} & \mathbf{0} \\ \mathbf{O}_{3\times3} & \mathbf{O}_{3\times3} & [\mathbf{f}_n\times] & \mathbf{R}_{b2n} & \mathbf{O}_{3\times3} & \mathbf{0} & \mathbf{0} \\ \mathbf{O}_{3\times3} & \mathbf{O}_{3\times3} & \mathbf{O}_{3\times3} & \mathbf{O}_{3\times3} & -\mathbf{R}_{b2n} & \mathbf{0} & \mathbf{0} \\ \mathbf{O}_{3\times3} & \mathbf{O}_{3\times3} & \mathbf{O}_{3\times3} & \mathbf{O}_{3\times3} & \mathbf{O}_{3\times3} & \mathbf{0} & \mathbf{0} \\ \mathbf{O}_{3\times3} & \mathbf{O}_{3\times3} & \mathbf{O}_{3\times3} & \mathbf{O}_{3\times3} & \mathbf{O}_{3\times3} & \mathbf{0} & \mathbf{0} \\ \mathbf{0}^T & \mathbf{0}^T & \mathbf{0}^T & \mathbf{0}^T & \mathbf{0}^T & 0 & 1 \\ \mathbf{0}^T & \mathbf{0}^T & \mathbf{0}^T & \mathbf{0}^T & \mathbf{0}^T & 0 & 0 \end{bmatrix}}_{\mathbf{F}} \cdot \begin{bmatrix} \delta\mathbf{p}_n \\ \delta\mathbf{v}_n \\ \delta\boldsymbol{\psi} \\ \delta\mathbf{f}_b \\ \delta\boldsymbol{\omega}_b \\ c\delta t \\ c\delta\dot{t} \end{bmatrix} + \mathbf{w}, \quad (13)
$$

where $[\mathbf{f}_n\times]$ is the skew-symmetric matrix of $\mathbf{f}_n = \mathbf{R}_{b2n}\mathbf{f}_b$; the unknown gyroscope and accelerometer slowly time-varying biases $\delta\mathbf{f}_b$ and $\delta\boldsymbol{\omega}_b$ are modelled as constant plus random-walk processes; the receiver clock drift error $c\delta\dot{t}$ is modelled as random walk, while the receiver clock bias $c\delta t$ is modelled as the integral of the clock drift error.

With this simplified linear error model, the UKF performs the same time-update procedure as the EKF algorithm. Thus, a large amount of computational burden can be saved. It is implemented in continuous-time domain and can be transformed to its equivalent discrete-time domain using Equation (14), where equivalent means that the discrete and the continuous-time model can predict the same system state at specified discrete time instants.

$$
\mathbf{A}_k = e^{\mathbf{F}(k\Delta t)\cdot\Delta t} = \mathbf{I} + \mathbf{F}(k\Delta t)\cdot\Delta t + \frac{\mathbf{F}(k\Delta t)^2\cdot\Delta t^2}{2!} + \text{h.o.t} \quad (14)
$$

In this paper, only the first order term is considered. The other higher order terms are neglected. The functional flowchart of INS/GPS tightly-coupled integration is given in Figure 2.

5.2. *Derivation of GPS Observation Model.* In INS/GPS tightly-coupled integration, the observation models are heavily non-linear. For UKF and UPF algorithms, non-linear Equations (15) and (16) are used. For the EKF algorithm, the Jacobian matrix in Equation (17) is exploited.

The predicted pseudorange measurements are related with error states δx_n, δx_e, δx_d, $c\delta t$ as:

$$
\hat{\rho}^{sv(n)} = \sqrt{(x_n^{sv(n)} - \hat{x}_n^{ins} + \delta x_n)^2 + (x_e^{sv(n)} - \hat{x}_e^{ins} + \delta x_e)^2 + (x_d^{sv(n)} - \hat{x}_d^{ins} + \delta x_d)^2} + c(\hat{t} - \delta t),
$$
$$(15)$$

where $\hat{\rho}^{sv(n)}$ $(n=1,\ldots,N)$ is the INS predicted pseudorange measurement from the n-th satellite; $x_n^{sv(n)}$, $x_e^{sv(n)}$ and $x_d^{sv(n)}$ are the n-th satellite position coordinates expressed in NED navigation frame; $\hat{x}_n^{ins}$, $\hat{x}_e^{ins}$ and $\hat{x}_d^{ins}$ are the INS estimated position in NED navigation frame.
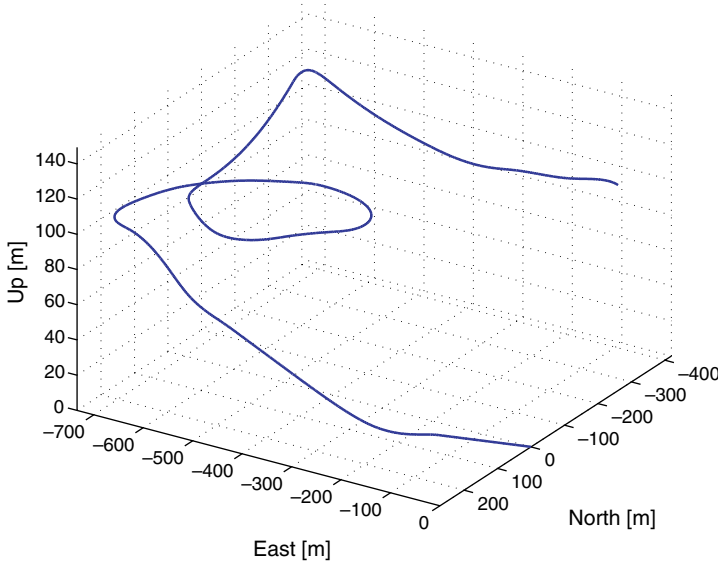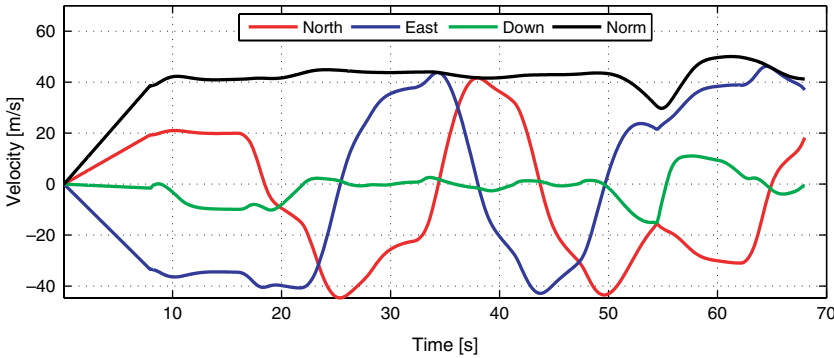
Figure 2. Functional flowchart of INS/GPS tightly-coupled integration.

The INS predicted delta pseudorange measurements are related with error states $\delta\dot{x}_n$, $\delta\dot{x}_e$, $\delta\dot{x}_d$, $c\delta\dot{t}$ as

$$\hat{\dot{\rho}}^{sv(n)} = a_n^{sv(n)}(\dot{x}_n^{sv(n)} - \hat{\dot{x}}_n^{ins} + \delta\dot{x}_n) + a_e^{sv(n)}(\dot{x}_e^{sv(n)} - \hat{\dot{x}}_e^{ins} + \delta\dot{x}_e) + a_d^{sv(n)}(\dot{x}_d^{sv(n)} - \hat{\dot{x}}_d^{ins} + \delta\dot{x}_d)$$
$$+ c(\hat{\dot{t}} - \delta\dot{t}),$$

where

$$a_n^{sv(n)} = \frac{x_n^{sv(n)} - \hat{x}_n^{ins} + \delta x_n}{\hat{d}^{sv(n)}}, \quad a_e^{sv(n)} = \frac{x_e^{sv(n)} - \hat{x}_e^{ins} + \delta x_e}{\hat{d}^{sv(n)}}, \quad a_d^{sv(n)} = \frac{x_d^{sv(n)} - \hat{x}_d^{ins} + \delta x_d}{\hat{d}^{sv(n)}},$$

$$\hat{d}^{sv(n)} = \sqrt{(x_n^{sv(n)} - \hat{x}_n^{ins} + \delta x_n)^2 + (x_e^{sv(n)} - \hat{x}_e^{ins} + \delta x_e)^2 + (x_d^{sv(n)} - \hat{x}_d^{ins} + \delta x_d)^2}. \quad (16)$$

For the EKF algorithm, the Jacobian observation matrix is calculated as

$$\mathbf{H} = \begin{bmatrix} l_n^{sv(n)} & l_e^{sv(n)} & l_d^{sv(n)} & 0 & 0 & 0 & ... & 0 & 1 & 0 \\ 0 & 0 & 0 & l_n^{sv(n)} & l_e^{sv(n)} & l_d^{sv(n)} & 0 & ... & 0 & 1 \end{bmatrix}_{2N \times 17}, \quad (17)$$

where $l_n^{sv(n)}$, $l_e^{sv(n)}$, $l_d^{sv(n)}$ ($n = 1,..., N$) denote the unit direction vectors pointing from the n-th satellite position in the navigation frame to the INS estimated user position $\hat{x}_n^{ins}$, $\hat{x}_e^{ins}$, $\hat{x}_d^{ins}$ in navigation frame.

6. EXPERIMENTAL SETUP. In the following experiments, a 3D UAV trajectory is used as shown in Figure 3. The velocity dynamic profiles in NED navigation frame are plotted in Figure 4. For the majority of time, the UAV is flying at a velocity of 40 m/s, which is 144 km/h. In the north and east directions, the velocities vary dramatically from approximately $+40$ m/s to -40 m/s in just a few seconds.

This trajectory is derived from the strapdown processing of the field collected high quality IMU raw data from the University of Sydney. It is considered as the reference

Figure 3. UAV trajectory.

Figure 4. Trajectory dynamics.

for generating GPS measurements (i.e., pseudorange and Doppler, where the Doppler measurement is derived from the first order central difference of the carrier phase data $d_k^{(n)} \approx (\phi_{k+1}^{(n)} - \phi_{k-1}^{(n)})/(2 \cdot \Delta t)$) using the SATNAV toolbox from GPSoft LLC. It needs to be noted that such an approximation of instantaneous Doppler measurement using carrier phase data at low output rate (i.e., 1 or 5 Hz) is very poor for high dynamic applications. In order to tackle this problem, 50 Hz carrier phase data are used to derive the Doppler measurements.

Under the assumption that the DGPS mode is used, the majority of GPS measurement errors can be corrected or minimized to small values, i.e., the ionospheric and tropospheric delays, satellite clock errors, and errors in satellite ephemeris predictions. Therefore, they are not considered in the simulation. For the INS/GPS integration, we assume that a NovAtel DL-4plus GPS receiver is used.

Table 3. Error characteristics of the simulated LandMark^{TM} 20 MEMS-based IMU (1σ).

Gyroscopes (Angular rates)	Bias stability [°/h] 30	Noise (ARW) [°/s/$\sqrt{Hz}$] 0·028
Accelerometers (Specific forces)	Bias stability [mg] 0·5	Noise (VRW) [mg/$\sqrt{Hz}$] 0·07
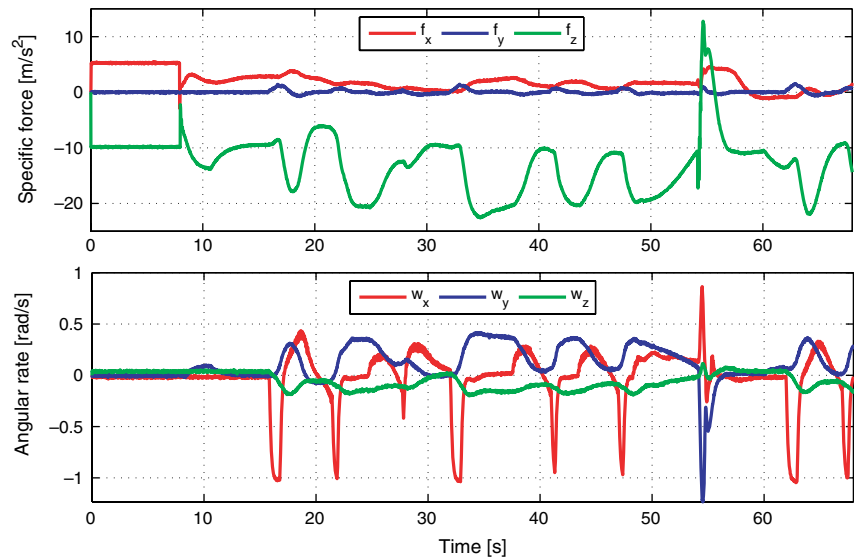
Figure 5. Simulated LandMark^{TM} 20 MEMS IMU raw data.

From the specification, we know when the receiver is navigating in a benign signal environment, in DGPS mode, the typical values for position and velocity estimation errors are 0·45 m CEP and 0·03 m/s RMS respectively. Hence, we reasonably simulate 0·5 m (1σ) thermal noise errors on the pseudorange, and 0·02 m/s (1σ) thermal noise errors on the Doppler measurements. For the multipath, we use the SATNAV toolbox from GPSoft LLC with minor modifications to generate the errors. The basic idea is to generate the time-correlated Gaussian noises in horizontal plane and project them onto the line-of-sight directions from the satellites to receiver through the cosine of the elevation angles (non-linear operation on Gaussian noises). The parameters we have chosen are 0·4 m and 0·01 m/s for generating multipath errors on pseudorange and Doppler measurements. Time constant is chosen to be 15 s.

An automotive level LandMark^{TM} 20 MEMS-IMU from Gladiator Technologies Inc. is simulated for the IMU. The main sensor errors are simulated according to the specification shown in Table 3. The IMU raw data (simulated errors plus the field collected high quality IMU raw data) in the body frame are plotted in Figure 5.

The integration is accomplished by using the aforementioned state space error modelling. In such a manner, the three design schemes, i.e., EKF, UKF and UPF, share the same time-update procedures. An ideal time-synchronization has been assumed for the GPS and INS data processing. Flexure and vibrations have not been
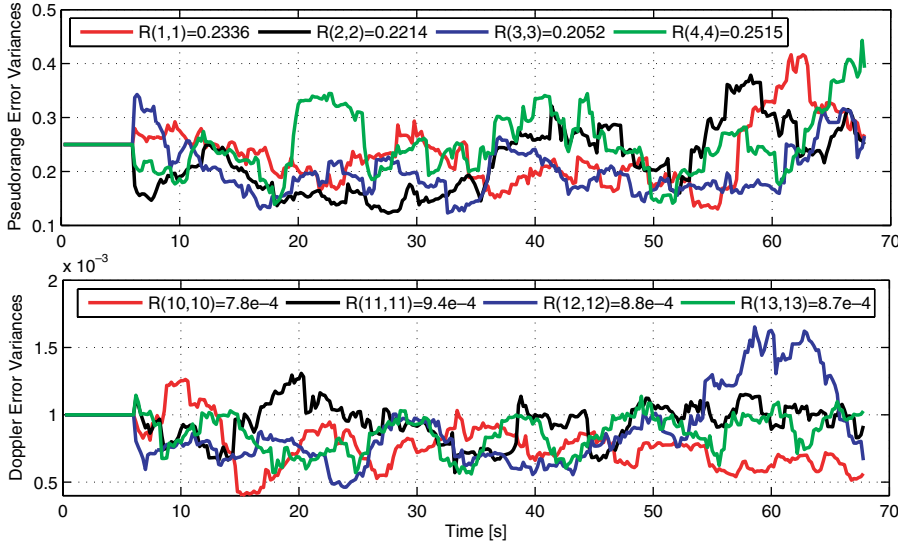
Figure 6. Adaptively updated measurement error variance parameters under benign signal environment.

taken into consideration. For the integrated KF, the system time-update happens at the IMU measurements update rate of 50 Hz, while the measurement-update happens at the GPS measurement update rate of 5 Hz. The simulation is conducted with eight satellites in view. The program is implemented in MATLAB 7.1 using the Intel Core 2 Duo T7600 CPU and 2GB RAM.

7. SIMULATION & RESULTS. Four groups of tests with well-controlled signal environments are conducted in this section to illustrate the concept of the AUPF algorithm. In Section 7.1, the adaptive optimization of the INS/GPS system performance under benign signal environment is investigated and in Section 7.2, highly reflective signal environments are simulated. The advantages of using the adaptive filtering technique can be readily seen. In Section 7.3, system performances from the AEKF, AUKF and AUPF are compared using one group of IMU and GPS data. In Section 7.4, Monte Carlo experiments are conducted to verify the improved system performance from the AUPF algorithm. The position and velocity errors analyzed in the following experiments represent the norm of position and velocity errors (i.e., $\|\Delta \mathbf{p}\|$ and $\|\Delta \mathbf{v}\|$) in north, east and down directions, which is calculated as $\|\Delta \mathbf{x}\| = \sqrt{\Delta x_n^2 + \Delta x_e^2 + \Delta x_d^2}$.

7.1. *System performance comparison between the UKF and AUKF algorithms under benign signal conditions.* The benign signal environment is simulated according to the parameters given in Section 6. The simulated measurement error variances on pseudorange measurements are about 0·25 and for Doppler, they approach 0·001. The computational burden of the UKF is 10·65 s whilst for the AUKF, 17·26 s is required. This extra time is caused by the additional estimation block as marked by the dashed-line box in Figure 1.
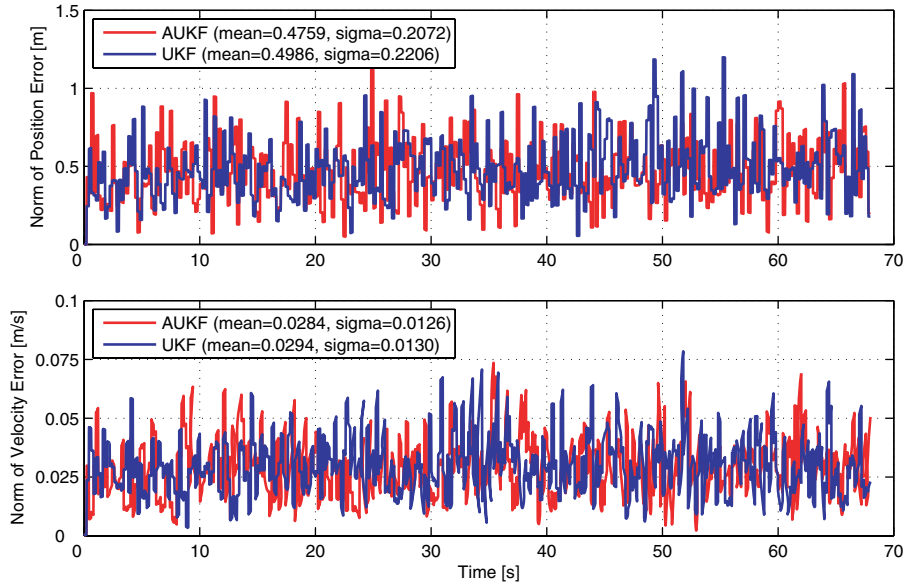
Figure 7. Performance comparison between the UKF and AUKF under benign signal environment.

The four out of eight pseudorange and Doppler measurement error variance parameters are chosen and depicted in Figure 6. They approximately match the prior given statistical information, so the UKF and AUKF algorithms yield comparable system performances, as shown in Figure 7.

7.2. *System performance comparison between the UKF and AUKF algorithms under highly reflective noisy signal environment.* For stationary applications, the multipath errors are strongly related to the geometry between the receiver antenna and the satellites, and the errors are time-correlated. Nevertheless, for a UAV, which is manoeuvring in the air with velocity of 40 m/s from one GPS measurement update epoch to the next, the environment will be changed dramatically (i.e., the geometry between the GPS antenna on the UAV with the buildings or trees nearby), and it is usually not possible to relate the multipath errors from one epoch to the next. Thus, we consider the multipath errors to be time-uncorrelated. To simulate such highly reflective multipath effects, we choose 5 m and 0·1 m/s error parameters (1σ) to generate multipath errors on pseudorange and Doppler measurements (typical values according to Misra and Enge (2007)) using the SATNAV toolbox with minor modifications as introduced in Section 6. They are added onto the GPS measurements randomly without reference to the time. A scale factor is introduced, which changes from 0 to 1 over time. It is used to control the impact of the noises (thermal noises plus multipath errors) to approximate the case that the UAV navigates from the open sky environment into the highly reflective noisy signal environment.

The system performance from the UKF and AUKF algorithms are compared as shown in Figure 8. The residual-based adaptive covariance matching technique improves the system estimation accuracy by adaptively adjusting the weights to the
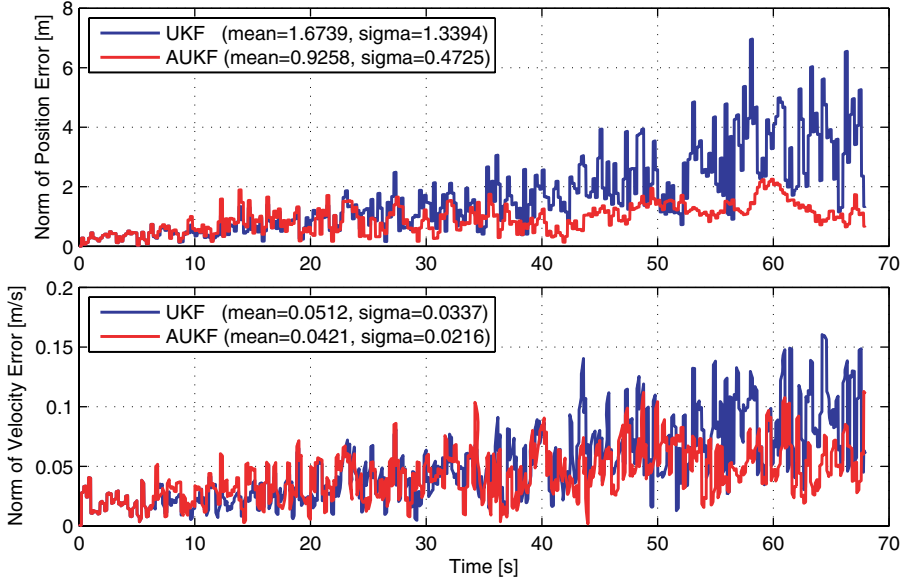
Figure 8. Performance comparison between the UKF and AUKF under highly reflective signal environment.
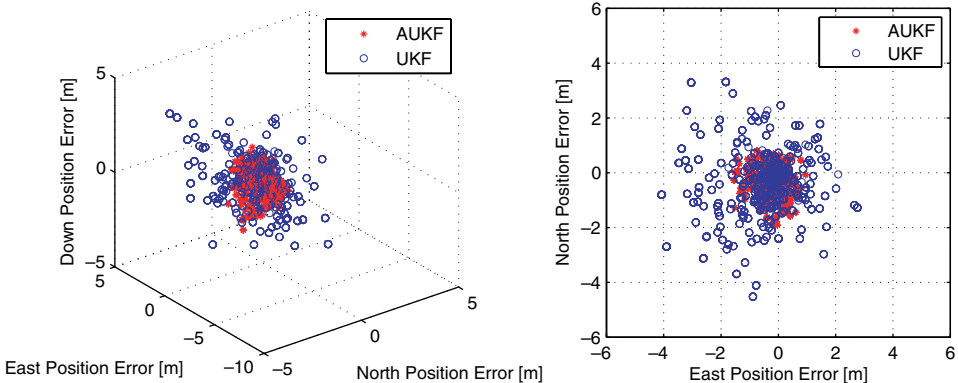
Figure 9. Position error comparisons in north, east and down directions between the AUKF and UKF.

measurements on the basis of the incoming residual sequences. For the measurement error variance parameters, the prior given data fit the signal environment only in the first few epochs. After that, the measurement noises grow steadily over time due to the increased thermal noises and multipath effects, which turn the prior error statistics to be much smaller than the practical ones. Large estimation errors from the UKF algorithm (blue curves) are readily seen in the figure.

The position estimation errors in north, east and down directions are given in Figure 9, where substantial improvements are obtained using the residual-based AUKF algorithm. It also shows that for the residual-based AUKF, the perfect
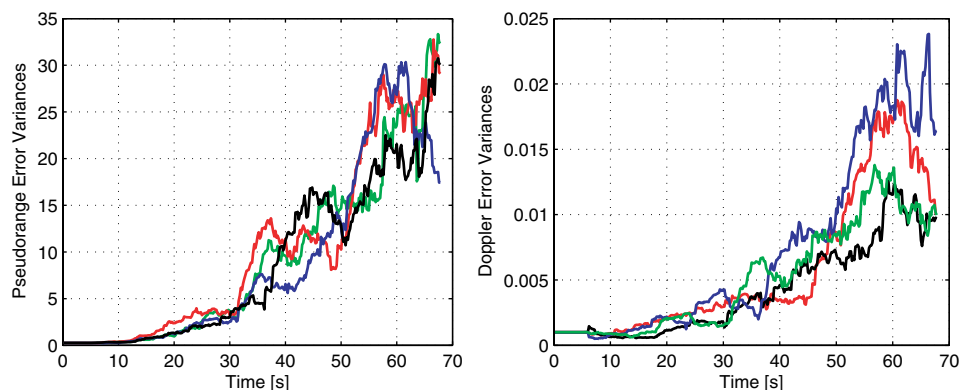
Figure 10. Adaptively updated measurement error variances under highly reflective signal environment.

a priori knowledge of the measurement noises is not of the main importance. The adaptively updated measurement error variance parameters from the randomly chosen four out of eight pseudorange and Doppler measurements are plotted in Figure 10. The window size N is chosen to be 30. Therefore, in the first 30 epochs, the initial statistical parameters are used.

7.3. *System performance comparison between the AEKF, AUKF and AUPF algorithms.* For comparison purpose, one set of IMU and GPS data is used for all three filters. The signal environment was the same as that introduced in Section 7.2. Three comparisons are made: (1) compare the system performances between the AEKF, AUKF and AUPF algorithms using one group of GPS and IMU data; (2) use different numbers of particles in the AUPF algorithm to investigate the changes in its estimation accuracy, and (3) summarize the simulations conducted in this Section.

7.3.1. *System performance comparison in highly reflective noisy signal environments.* Figure 11 plots the norm of position and velocity estimation errors from the AEKF, AUKF and AUPF algorithms using one group of GPS and IMU data. 100 particles are used in the AUPF algorithm. As shown in the figure, in the first few seconds, the introduced thermal noise and multipath effects are very small. The system is working with the prior given measurement error statistics which match the simulated signal environment at the start of the integration and hence, the estimation accuracies from three filters are comparable. After approximately 10 seconds, the noise and multipath errors grow steadily over time. It takes some time for the corrupted measurements to become less weighted. In this period, the magnitude of innovation sequence is increasing, which accordingly turns the non-linearity problem to be more critical. That is, the neglect of the higher order terms of Taylor series expansion may lead to the degrading of system estimation accuracy. Besides, the multipath errors simulated in the trajectory are not exactly Gaussian-distributed. From the figure, the AUPF presents the best estimation accuracy when the norms of position and velocity errors are compared. Nevertheless, for the AUPF, the particles are generated randomly, and the system estimation accuracy varies after each simulation run. Hence, its system performance can only be evaluated in the sense of statistics instead of just from one run of the simulation. For a thorough investigation
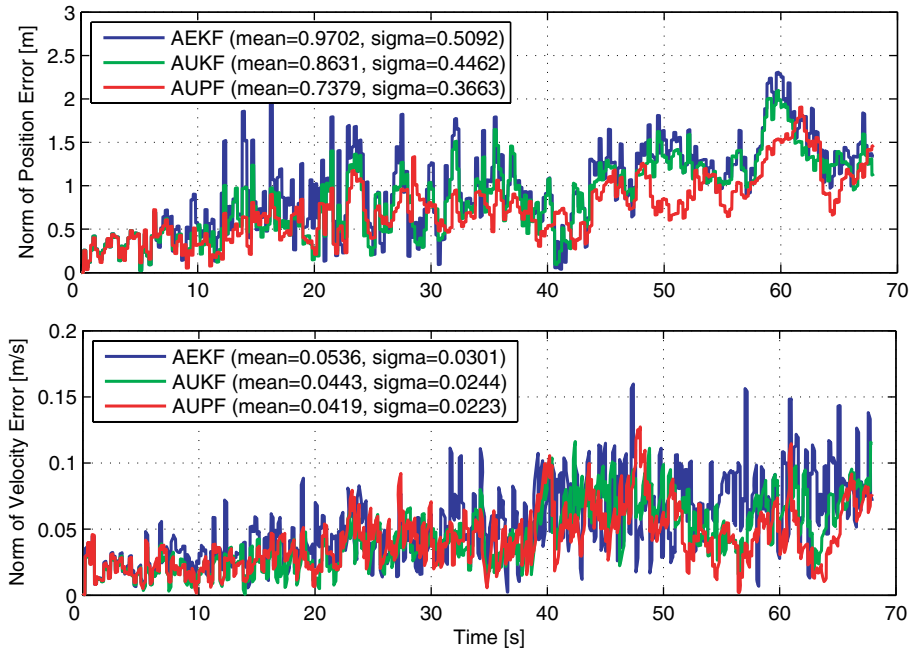
Figure 11. Performance comparison of AEKF, AUKF and AUPF under highly reflective signal environment.

of its estimation accuracy, in the next Section, 100 runs are conducted using different numbers of particles.

7.3.2. *Comparisons of the AUPF with different numbers of particles.* In this test, 25, 50, 100, and 500 particles are used. 100 simulation runs are conducted using the same group of GPS and IMU data. The relationship between the improved system estimation accuracy and the increased number of particles is illustrated in Figure 12.

Using 25 particles in the AUPF algorithm, the system diverges frequently. With 500 particles, the system estimation accuracy is much improved compared with the use of fewer particles. It can be expected that further improved system performance can be achieved when 1000 or more particles are used at the expense of significantly increased processing time. It is often the case that slightly improved estimation accuracy requires huge extra computational burden. In practice, the choice of the minimum number of particles is always a trade off between the increased processing load and the improved estimation accuracy, which should be discussed on the basis of the specific application at hand.

The relationship between the system performance and the number of particles is revealed in Figure 13. If ten particles are used, only one particle may be significantly weighted and the others are much less weighted. However, the estimates from this particle may still be far away from the true values. Only with greater numbers, for instance, 50 or 100 particles, will more particles be significantly weighted through the likelihood function, and the system *a posteriori* estimates are calculated from the normalized weighted sum of particle estimates, which is expected to be much closer to the true values.
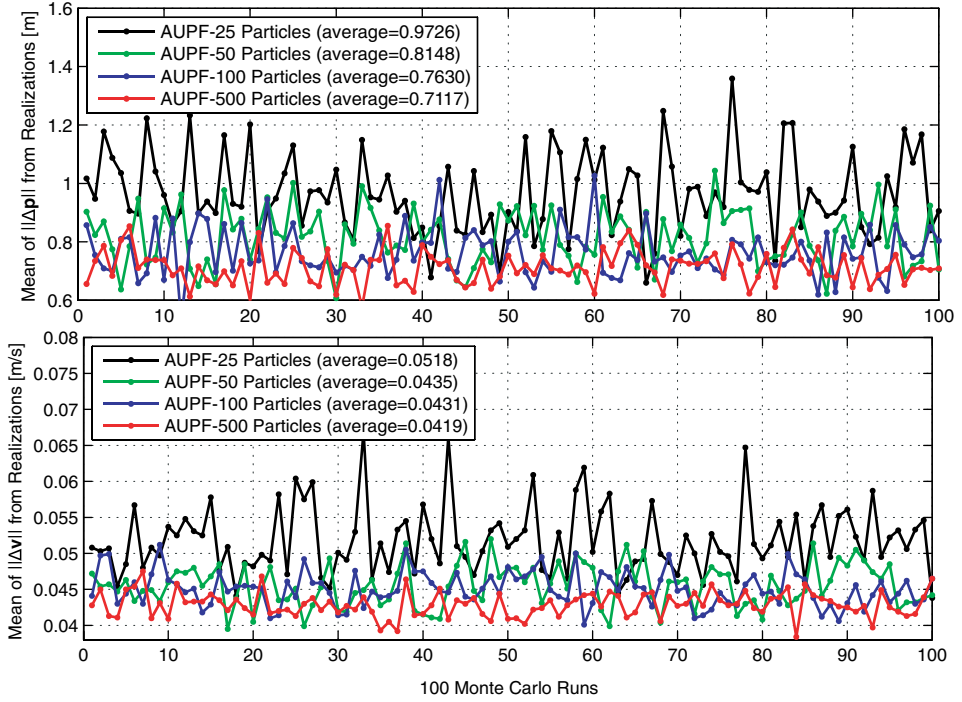
Figure 12. System performance analysis of the AUPF algorithm with different number of particles.

7.3.3. *Summary of the system performances from the AEKF, AUKF and AUPF algorithms.* The system estimation accuracy and processing time from simulations conducted in Section 7.3.1 and 7.3.2 are summarized in Table 4. All the filters are processing the same group of GPS and IMU data so that the differences are caused by the algorithms.

In Table 4, $m_{\|\Delta\mathbf{p}\|}$ and $m_{\|\Delta\mathbf{v}\|}$ represent the averaged mean of $\|\Delta\mathbf{p}\|$ and $\|\Delta\mathbf{v}\|$ errors from realizations, while $m_{\sigma(\|\Delta\mathbf{p}\|)}$ and $m_{\sigma(\|\Delta\mathbf{v}\|)}$ denote the averaged sigma values on $\|\Delta\mathbf{p}\|$ and $\|\Delta\mathbf{v}\|$ errors from realizations. The time represents the number of seconds elapsed in the processing of the whole GPS and IMU data from the 68 s UAV trajectory. It is worth mentioning that this is just an indication to show the computational cost, because it depends largely on the style of programming in MATLAB and the computational power from the CPU. For exact knowledge of the computational burden, the number of numerical operations, i.e., additions and multiplications, should be counted.

In Table 4, the simulation results from the AUPF with 1000 particles are included. Due to its large computational burden, the results are the average values from ten simulation runs, while for the other AUPF algorithms, i.e., with 25, 50, 100 and 500 particles, the results are the average values from 100 runs. For the AEKF and AUKF algorithms, using the same prior statistical parameters, i.e. $\mathbf{P}$, $\mathbf{Q}$ and $\mathbf{R}$, the system performance will be unchanged when the same group of GPS and IMU data is processed. Thus, one simulation run for each is performed. From the table, the AUPF with 100 particles is highly recommended due to its improved estimation accuracy

Table 4. System performance comparison between adaptive non-linear filtering methods.

Filters	Runs	$m_{\|\Delta\mathbf{p}\|}$ [m]	$m_{\sigma(\|\Delta\mathbf{p}\|)}$ [m]	$m_{\|\Delta\mathbf{v}\|}$ [m/s]	$m_{\sigma(\|\Delta\mathbf{v}\|)}$ [m/s]	Time [s]
AEKF	1	0·9702	0·5092	0·0536	0·0301	3·4882
AUKF	1	0·8831	0·4462	0·0443	0·0244	17·6617
AUPF(25)	100	0·9726	0·6075	0·0518	0·0382	32·4063
AUPF(50)	100	0·8148	0·4674	0·0435	0·0245	45·9631
AUPF(100)	100	0·7630	0·4298	0·0431	0·0236	74·1883
AUPF(500)	100	0·7117	0·3848	0·0419	0·0227	308·5563
AUPF(1000)	10	0·6921	0·3736	0·0407	0·0218	598·5516
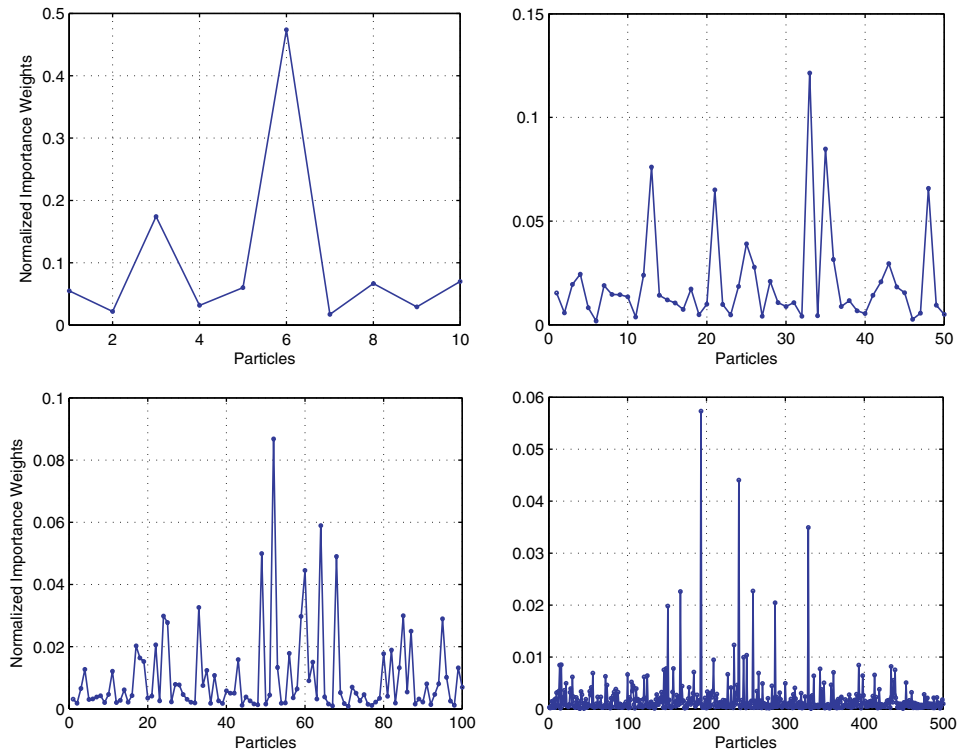
Figure 13. Normalized importance particle weights.

and affordable computational burden. Further increasing the number of particles yields very limited improved system estimation accuracy, but requires significantly increased processing load.

7.4.   *Comparison between the AEKF, AUKF and AUPF algorithms using 100 sets of GPS and IMU data.*   In this test, 100 groups of GPS and IMU data are simulated to test the system performances from the AEKF, AUKF and AUPF (100 particles) algorithms. The signal environments are the same as the ones used in Sections 7.2 and 7.3.

In Figure 14, improvements from the AUPF algorithm with 100 particles are readily seen, not only with system estimation accuracy, but also in the robustness of
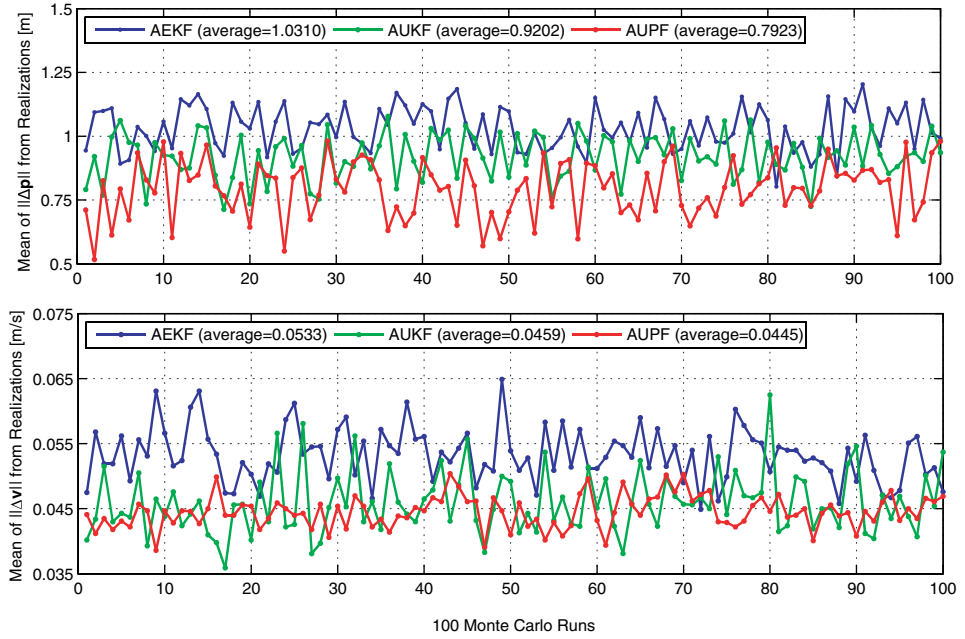
Figure 14. System performance comparison between the AEKF, AUKF and AUPF algorithms
under highly reflective signal environment with 100 Monte Carlo simulations.

the system performance. In the experiments, the position estimation errors mainly arise from the multipath effects introduced onto the pseudorange measurements. However, for the velocity estimation errors, the multipath errors added onto the Doppler measurements are relatively small. That is, the Doppler measurement noises are nearly Gaussian-distributed. And hence, the benefit from the AUPF to AUKF is small. Nevertheless, both the AUKF and AUPF use non-linear observation equations directly and yield improved estimation solutions as compared with the AEKF algorithm.

8. CONCLUSIONS. In this paper, the AUPF algorithm is proposed as a non-linear/non-Gaussian estimation method for the INS/GPS tightly-coupled integration. It outperforms the AEKF and AUKF algorithms when a sufficient number of particles are generated. The residual-based adaptive estimation technique is used in the filter to track the changes in the system observation data in an adaptive manner, which makes the system more robust. The simulation tests verify the benefits of the AUPF algorithm in comparison to the AEKF and AUKF algorithms. The computational burden is also analyzed and shows that the AUPF with 100 particles is of highest interest for the simulation conducted in this paper.

REFERENCES

Arulampalam, M. S., Maskell, S., Gordon N., Clapp, T. (2002). A Tutorial on Particle Filters for Online Nonlinear/Non-Gaussian Bayesian Tracking, *IEEE Trans. on Signal Processing*, **50**(2), 174–188.

Brown, R. G., Hwang P. Y. C. (1997). *Introduction to Random signals and Applied Kalman Filtering*, John Wiley & Sons, New York.

EI-Sheimy, N., Niu, X. J. (2007). The Promise of MEMS to the navigation community, *InsideGNSS*, March/April 2007, 46–56.

Farrell, J., and Barth, M. (1999). *The global positioning system and inertial navigation*, McGraw-Hill professional, New York.

Gordon, N., Salmond, D., Smith, A. F. M. (1993). Novel Approach to Nonlinear and Non-Gaussian Baysian State Estimation, in: *IEE Proceedings – Radar and Signal Processing*, **140**(2), 107–113.

Groves, P. D. (2008). *Principles of GNSS, Inertial and multisensor integrated navigation systems*, Artech House, Boston.

Haug, A. J., (2005). *A Tutorial on Bayesian Estimation and Tracking Techniques Applicable to Nonlinear and Non-Gaussian Processes*, MITRE technical report.

Hide, C., Moore, T., Smith, M. (2003). Adaptive Kalman Filtering for Low-cost INS/GPS, *The Journal of Navigation*, **56**, 143–152.

Julier, S. J., Uhlmann, J. K., Durrant-Whyte, H. F. (2000). A New Method for the Nonlinear Transformation of Means and Covariances in Filters and Estimators, *IEEE Trans. on automatic control*, **45**, 477–482.

Julier, S. J., Uhlmann, J. K., Durrant-Whyte, H. F. (1995). A new approach for filtering nonlinear systems, *Proceedings of the American Control Conference*, 1628–1632.

Knedlik, S., Zhou, J. C., Loffeld, O. (2009). On position and attitude estimation for remote sensing with bistatic SAR, *Geoscience and remote sensing*, Pei-Gee Peter Ho, Ed. Vukovar: In-Tech, 75–90.

Kotecha, J. H., Djuric, P. M. (2003). Gaussian particle filtering, *IEEE Trans. On signal processing*, **51**(10), 2592–2601.

Li, Y., Rizos, C., Wang, J. L., Mumford, P., Ding, W. D. (2008). Sigma-Point Kalman Filtering for Tightly Coupled INS/GPS Integration, *Journal of the Institute of Navigation*, **55**(3), 167–179.

Mehra, R. K. (1971). On the Identification of Variances and Adaptive Kalman Filtering, *IEEE Trans. on automatic control*, **AC-15**(**2**), 175–184.

Mohamed, A. H. and Schwarz, K. P. (1999). Adaptive Kalman filtering for INS/GPS, *Journal of Geodesy*, **73**, 193–203.

Misra, P., Enge, P. (2007). *Global Positioning System: Signals, Measurements, and Performance*, Ganga-Jamuna Press.

Simon, D. (2006). *Optimal State Estimation Kalman*, John Wiley & Son, New Jersey.

Sukkarieh, S. (2000). *Low Cost, High Integrity, Aided Inertial Navigation Systems for Autonomous Land Vehicles*, Ph.D Dissertation, University of Sydney.

Titterton, D. H., Weston, J. L. (2004). *Strapdown Inertial Navigation Technology* (*2nd Edition*). Institution of Engineering and Technology.