

GPS/INS Integration Accuracy Enhancement Using the Interacting Multiple Model Nonlinear Filters

D.J. Jwo^{*1}, F.C. Chung², K.L. Yu³

¹ Department of Communications, Navigation and Control Engineering
National Taiwan Ocean University, Keelung 20224, Taiwan

*djwo@mail.ntou.edu.tw

² Inventec Appliances, Wugu Industrial Park
Wugu, New Taipei City 24890, Taiwan

³ Systems & Technology Corp., Hsichih, New Taipei City 22101, Taiwan

ABSTRACT

In this paper, performance evaluation for various single model nonlinear filters and nonlinear filters with interacting multiple model (IMM) framework is carried out. A high gain (high bandwidth) filter is needed to response fast enough to the platform maneuvers while a low gain filter is necessary to reduce the estimation errors during the uniform motion periods. Based on a soft-switching framework, the IMM algorithm allows the possibility of using highly dynamic models just when required, diminishing unrealistic noise considerations in non-maneuvering situations. The IMM estimator obtains its estimate as a weighted sum of the individual estimates from a number of parallel filters matched to different motion modes of the platform. The use of an IMM allows exploiting the benefits of high dynamic models in the problem of vehicle navigation. Simulation and experimental results presented in this paper confirm the effectiveness of the method.

Keywords: Interacting multiple model, Unscented Kalman filter, Unscented particle filter, Integrated navigation.

1. Introduction

The synergy of Global Positioning System (GPS) [1,2] and inertial navigation system (INS) [1,3] has been widely explored due to their complementary operational characteristics. The GPS/INS integrated navigation system is typically carried out through the Kalman filter [1-2] to estimate the vehicle state variables and suppress the navigation measurement noise. In the Kalman filter or its nonlinear version, the extended Kalman filter (EKF), the state distribution is approximated by a Gaussian random variable (GRV), which is then propagated analytically through the first-order linearization of the nonlinear system.

Proposed by Julier, et al. [4], the unscented Kalman filter (UKF) is a nonlinear, distribution approximation method, which uses a finite number of carefully chosen sigma points to propagate the probability of state distribution through the nonlinear dynamics of system so as to completely capture the true mean and covariance of the GRV with a limited number sigma points by using the Unscented Transform (UT) [5-7]. The basic premise behind the UKF is it is easier to

approximate a Gaussian distribution than it is to approximate an arbitrary nonlinear function. When the sample points are propagated through the true nonlinear system, the posterior mean and covariance can be captured accurately to the second order of Taylor series expansion for any nonlinear system. The particle filter (PF) [8-12] is a probability-based estimator. The Bayesian estimation is the foundation for particle filters. A large number of particles are required to obtain reasonable accuracy as the latest measurement is totally ignored. Henceforth, many variants of these filters are suggested by a number of researchers. To improve the estimate accuracy, the EKF or UKF can be used to generate the true mean and covariance of the proposal distribution.

The implementation of Kalman filter requires the *a priori* knowledge of both the process and measurement models. In various circumstances where there are uncertainties [13] in the system model and noise description, and the assumptions on the statistics of disturbances are violated due to the fact that in a number of practical situations, the

availability of a precisely known model is unrealistic. The adaptive Kalman filters (AKF) [14-15] is used based on an on-line estimation of motion as well as the signal and noise statistics available data. Many efforts have been made to improve the estimation of the covariance matrices based on the innovation-based estimation approach, resulting in the innovation adaptive estimation (IAE).

The other approach is the multiple model adaptive estimate (MMAE). Proposed by Bar-Shalom and Blom, the interacting multiple model (IMM) algorithm [12,16-21] has the configuration that runs in parallel several model-matched state estimation filters, which exchange information (interact) at each sampling time. The IMM algorithm has been originally applied to target tracking [17-19], and recently extended to GPS navigation [20,21]. This algorithm carries out a soft-switching between the various modes by adjusting the probabilities of each mode, which are used as weightings in the combined global state estimate. The AKF approach is based on parametric adaptation, while the IMM approach is based on filter structural adaptation (model switching). In the IMM approach, multiple models are developed to describe various dynamic behaviors.

The remainder of this paper is organized as follows. In Section 2, the nonlinear filters are discussed. The IMM filter algorithm is introduced in Section 3. In Section 4, numerical simulation and experiment on navigation processing are carried out to evaluate the performance for various nonlinear filters in single-filter framework mode as well as in multiple-model framework. Conclusions are given in Section 5.

2. The Nonlinear filters

Given a single model equation in discrete time

$$\mathbf{x}_{k+1} = \mathbf{f}_k(\mathbf{x}_k) + \mathbf{w}_k \quad (1)$$

$$\mathbf{z}_k = \mathbf{h}_k(\mathbf{x}_k) + \mathbf{v}_k \quad (2)$$

where the state vector $\mathbf{x}_k \in \mathfrak{R}^n$, process noise vector $\mathbf{w}_k \in \mathfrak{R}^m$, measurement vector $\mathbf{z}_k \in \mathfrak{R}^m$, measurement noise vector $\mathbf{v}_k \in \mathfrak{R}^m$, \mathbf{Q}_k is the

process noise covariance matrix and \mathbf{R}_k is the measurement noise covariance matrix.

2.1 The extended Kalman filter

The vectors \mathbf{w}_k and \mathbf{v}_k in Eqs. (1) and (2) are zero mean Gaussian white sequences having zero cross-correlation with each other:

$$\begin{aligned} E[\mathbf{w}_k \mathbf{w}_i^T] &= \mathbf{Q}_k \delta_{ik}; \quad E[\mathbf{v}_k \mathbf{v}_i^T] = \mathbf{R}_k \delta_{ik}; \\ E[\mathbf{w}_k \mathbf{v}_i^T] &= \mathbf{0} \quad \text{for all } i \text{ and } k \end{aligned} \quad (3)$$

where $E[\cdot]$ represents expectation, and superscript "T" denotes matrix transpose. The symbol δ_{ik} stands for the Kronecker delta function:

$$\delta_{ik} = \begin{cases} 1, & i = k \\ 0, & i \neq k \end{cases}$$

The discrete-time extended Kalman filter algorithm is summarized as follow:

(1) Initialize state vector and state covariance matrix: $\hat{\mathbf{x}}_{0|0}$ and $\mathbf{P}_{0|0}$

(2) Compute Kalman gain matrix:

$$\mathbf{K}_k = \mathbf{P}_{k|k-1} \mathbf{H}_k^T [\mathbf{H}_k \mathbf{P}_{k|k-1} \mathbf{H}_k^T + \mathbf{R}_k]^{-1}$$

(3) Update state vector:

$$\hat{\mathbf{x}}_{k|k} = \hat{\mathbf{x}}_{k|k-1} + \mathbf{K}_k [\mathbf{z}_k - \mathbf{h}_k(\hat{\mathbf{x}}_{k|k-1})]$$

(4) Update error covariance $\mathbf{P}_{k|k} = [\mathbf{I} - \mathbf{K}_k \mathbf{H}_k] \mathbf{P}_{k|k-1}$

(5) Predict new state vector and state covariance matrix

$$\hat{\mathbf{x}}_{k+1|k} = \mathbf{f}_k(\hat{\mathbf{x}}_{k|k})$$

$$\mathbf{P}_{k+1|k} = \Phi_k \mathbf{P}_{k|k} \Phi_k^T + \mathbf{Q}_k$$

where the linear approximation equations for system and measurement matrices are obtained through the relations

$$\Phi_k \approx \left. \frac{\partial \mathbf{f}_k}{\partial \mathbf{x}} \right|_{\mathbf{x}=\hat{\mathbf{x}}_{k|k-1}}; \mathbf{H}_k \approx \left. \frac{\partial \mathbf{h}_k}{\partial \mathbf{x}} \right|_{\mathbf{x}=\hat{\mathbf{x}}_{k|k-1}}$$

2.2 The unscented Kalman filter

Suppose the mean $\bar{\mathbf{x}}$ and covariance \mathbf{P} of vector \mathbf{x} are known, a set of deterministic vector called sigma points can then be found. The ensemble mean and covariance of the sigma points are equal to $\bar{\mathbf{x}}$ and \mathbf{P} . The nonlinear function $\mathbf{y} = f(\mathbf{x})$ is applied to each deterministic vector to obtain transformed vectors. The ensemble mean and covariance of the transformed vectors will give a good estimate of the true mean and covariance of \mathbf{y} , which is the key to the unscented transformation.

Consider an n dimensional random variable \mathbf{x} , having the mean $\hat{\mathbf{x}}$ and covariance \mathbf{P} , and suppose that it propagates through an arbitrary nonlinear function f . The unscented transform creates $2n+1$ sigma vectors \mathbf{X} (a capital letter) and weighted points W

$$\mathbf{X}_{(0)} = \hat{\mathbf{x}} \quad (4)$$

$$\mathbf{X}_{(i)} = \hat{\mathbf{x}} + (\sqrt{(n+\lambda)\mathbf{P}})_i^T, \quad i = 1, \dots, n \quad (5)$$

$$\mathbf{X}_{(i+n)} = \hat{\mathbf{x}} - (\sqrt{(n+\lambda)\mathbf{P}})_i^T, \quad i = 1, \dots, n \quad (6)$$

where $(\sqrt{(n+\lambda)\mathbf{P}})_i$ is the i th row (or column) of the matrix square root. $\sqrt{(n+\lambda)\mathbf{P}}$ can be obtained from the lower-triangular matrix of the Cholesky factorization; $\lambda = \alpha^2(n+k) - n$ is a scaling parameter; α determines the spread of the sigma points around $\hat{\mathbf{x}}$ and is usually set as $1e-4 \leq \alpha \leq 1$; k is a secondly scaling parameter usually set as 0; β is used to incorporate prior knowledge of the distribution of $\bar{\mathbf{x}}$; $W_i^{(m)}$ and $W_i^{(c)}$ are the weights for the mean and covariance, respectively, associated with the i th point.

$$W_0^{(m)} = \frac{\lambda}{(n+\lambda)} \quad (7)$$

$$W_0^{(c)} = W_0^{(m)} + (1 - \alpha^2 + \beta) \quad (8)$$

$$W_i^{(m)} = W_i^{(c)} = \frac{1}{2(n+\lambda)}, \quad i = 1, \dots, 2n \quad (9)$$

The sigma vectors are propagated through the nonlinear function to yield a set of transformed sigma points,

$$\mathbf{y}_i = f(\mathbf{X}_i), \quad i = 0, \dots, 2n \quad (10)$$

The mean and covariance of \mathbf{y}_i are approximated by a weighted average mean and covariance of the transformed sigma points as follows:

$$\bar{\mathbf{y}}_u = \sum_{i=0}^{2n} W_i^{(m)} \mathbf{y}_i \quad (11)$$

$$\bar{\mathbf{P}}_u = \sum_{i=0}^{2n} W_i^{(c)} (\mathbf{y}_i - \bar{\mathbf{y}}_u)(\mathbf{y}_i - \bar{\mathbf{y}}_u)^T \quad (12)$$

The implementation algorithm of UKF is summarized as follows:

(1) The transformed set is given by instantiating each point through the process model

$$\boldsymbol{\zeta}_{i,k|k-1} = \mathbf{f}_{k-1}(\mathbf{X}_{i,k-1}), \quad i = 0, \dots, 2n$$

(2) Predicted mean

$$\hat{\mathbf{x}}_{k|k-1} = \sum_{i=0}^{2n} W_i^{(m)} \boldsymbol{\zeta}_{i,k|k-1}$$

(3) Predicted covariance

$$\begin{aligned} \mathbf{P}_{k|k-1} &= \sum_{i=0}^{2n} W_i^{(c)} [\boldsymbol{\zeta}_{i,k|k-1} - \hat{\mathbf{x}}_{k|k-1}][\boldsymbol{\zeta}_{i,k|k-1} - \hat{\mathbf{x}}_{k|k-1}]^T + \mathbf{Q}_{k-1} \end{aligned} \quad (13)$$

(4) Instantiate each of the prediction points through observation model

$$\mathbf{Z}_{i,k|k-1} = \mathbf{h}_k(\boldsymbol{\zeta}_{i,k|k-1})$$

(5) Predicted observation

$$\hat{\mathbf{z}}_{k|k-1} = \sum_{i=0}^{2n} W_i^{(m)} \mathbf{z}_{i,k|k-1}$$

(6) Innovation covariance

$$\begin{aligned} \mathbf{P}_{zz} &= \sum_{i=0}^{2n} W_i^{(c)} [\mathbf{z}_{i,k|k-1} - \hat{\mathbf{z}}_{k|k-1}] [\mathbf{z}_{i,k|k-1} - \hat{\mathbf{z}}_{k|k-1}]^T + \mathbf{R}_k \end{aligned} \quad (14)$$

(7) Cross covariance

$$\mathbf{P}_{xz} = \sum_{i=0}^{2n} W_i^{(c)} [\mathbf{x}_{i,k|k-1} - \hat{\mathbf{x}}_{k|k-1}] [\mathbf{z}_{i,k|k-1} - \hat{\mathbf{z}}_{k|k-1}]^T \quad (15)$$

(8) Performing update

$$\mathbf{K}_k = \mathbf{P}_{xz} \mathbf{P}_{zz}^{-1}$$

$$\hat{\mathbf{x}}_{k|k} = \hat{\mathbf{x}}_{k|k-1} + \mathbf{K}_k (\mathbf{z}_k - \hat{\mathbf{z}}_{k|k-1})$$

$$\mathbf{P}_{k|k} = \mathbf{P}_{k|k-1} - \mathbf{K}_k \mathbf{P}_{zz} \mathbf{K}_k^T$$

2.3 The particle filter

Although the EKF and UKF can deal well with some nonlinear filtering problems, however, they always approximate $p(\mathbf{x}_k | \mathbf{Z}_k)$ to be Gaussian, where $p(\mathbf{x}_k | \mathbf{Z}_k) = \text{pdf}$ of \mathbf{x}_k conditioned on measurements $\mathbf{z}_1, \mathbf{z}_2, \dots, \mathbf{z}_k$. The particle filter is a probability-based estimator. If the true density is non-Gaussian, particle filters may lead to better performances in comparison to that of EKF or UKF. Figure 1 shows the recursive Bayesian state estimation, which is based on the Bayes' rule

$$p(\mathbf{x}_k | \mathbf{Z}_k) = \frac{p(\mathbf{y}_k | \mathbf{x}_k) p(\mathbf{x}_k | \mathbf{Z}_{k-1})}{p(\mathbf{y}_k | \mathbf{Z}_{k-1})} \quad (16)$$

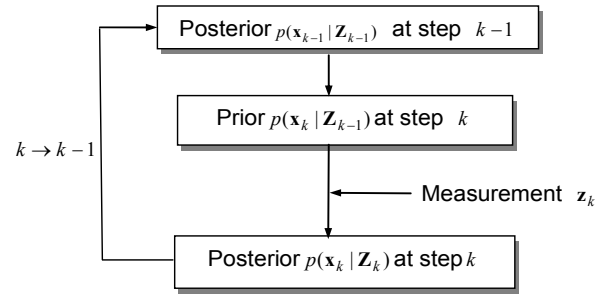


Figure 1. The recursive Bayesian state estimation.

In order to compute $p(\mathbf{x}_k | \mathbf{Z}_k)$, the equation $p(\mathbf{x}_k | \mathbf{Z}_{k-1})$ needs to be found from the Chapman-Kolmogorov equation and the marginal density function:

$$p(\mathbf{x}_k | \mathbf{Z}_{k-1}) = \int p(\mathbf{x}_k | \mathbf{x}_{k-1}) p(\mathbf{x}_{k-1} | \mathbf{Z}_{k-1}) d\mathbf{x}_{k-1} \quad (17)$$

The pdf $p(\mathbf{z}_k | \mathbf{Z}_{k-1})$ can be obtained to be

$$p(\mathbf{z}_k | \mathbf{Z}_{k-1}) = \int p(\mathbf{z}_k | \mathbf{x}_k) p(\mathbf{x}_k | \mathbf{Z}_{k-1}) d\mathbf{x}_k \quad (18)$$

The basic PFs combine two principles namely the Monte Carlo (MC) principle and the Importance Sampling (IS) method where the transition prior density $p(\mathbf{x}_k | \mathbf{x}_{k-1})$ is taken as the importance distribution:

$$q(\mathbf{x}_k | \mathbf{x}_{k-1}, \mathbf{z}_k) = p(\mathbf{x}_k | \mathbf{x}_{k-1}) \quad (19)$$

The weights of these particles are evaluated according to:

$$w_k \propto w_{k-1} p(\mathbf{z}_k | \mathbf{x}_k) \quad (20)$$

and

$$w_k \propto \frac{p(\mathbf{x}_k | \mathbf{Z}_k)}{q(\mathbf{x}_k | \mathbf{Z}_k)} \quad (21)$$

where $q(\mathbf{x}_k | \mathbf{x}_{k-1}, \mathbf{z}_k)$ is the importance density function, w_{k-1} are the importance weights of the previous epoch particles and w_k are the importance weights of the current epoch.

2.4 The extended Kalman particle filter and unscented particle filter

The extended Kalman particle filtering (EKPF) is formed when the EKF is used for the importance proposal generation [9]; the unscented particle filter (UPF) is formed when the UKF is used for the importance proposal generation [8]. In a local linearization technique, such as the EKPF and UPF, each particle is drawn from the local Gaussian approximation of the optimal importance distribution $p(\mathbf{x}_k | \mathbf{x}_{k-1}, \mathbf{z}_k)$ that is conditioned on the current state and the latest measurement

$$q(\mathbf{x}_k | \mathbf{x}_{k-1}, \mathbf{z}_k)_{\text{Optimal}} = q_N(\mathbf{x}_k | \mathbf{Z}_k) \quad (22)$$

where q_N denotes the Gaussian approximation of importance density and is obtained from EKF/UKF.

An improvement in the choice of proposal distribution over the simple transition prior can be accomplished by moving particles towards the regions of high likelihood, based on the most recent measurement. An effective approach to accomplish this is to use an EKF or UKF generated Gaussian approximation optimal proposal [4,6]

$$\hat{\mathbf{x}}_k^i \sim q(\mathbf{x}_k^i | \mathbf{x}_{k-1}^i, \mathbf{z}_k) \cong N(\mathbf{x}_k^i, \mathbf{P}_k^i) \quad (23)$$

where \mathbf{x}_k^i and \mathbf{P}_k^i are the mean and covariance of the i th particle generated by the EKF/UKF, respectively. This is accomplished by using a separated EKF or UKF to generate and propagate a Gaussian proposal distribution for each particle.

One cycle of the EKPF/UPF (i.e., the PF that uses EKF/UKF as proposal distribution) algorithm can be summarized as follows.

(1) Initialization. Assume $\{\mathbf{x}_0^i\}_{i=1}^N$ be a set of particles sampled from the prior $\mathbf{x}_0^i \sim p(\mathbf{x}_0)$ at $k=0$ and set

$$\begin{aligned} \hat{\mathbf{x}}_0^i &= E[\mathbf{x}_0^i] \\ \mathbf{P}_0^i &= E[(\mathbf{x}_0^i - \hat{\mathbf{x}}_0^i)(\mathbf{x}_0^i - \hat{\mathbf{x}}_0^i)^T] \end{aligned}$$

$$w_0^i = 1/N \quad (24)$$

where $i = 1 \dots N$

(2) Importance sampling

(i) Update each particle with the EKF/UKF to obtain mean $\bar{\mathbf{x}}_k^i$ and covariance \mathbf{P}_k^i .

(ii) Sample the particles from

$$\hat{\mathbf{x}}_k^i \sim q(\mathbf{x}_k^i | \mathbf{x}_{k-1}^i, \mathbf{z}_k) \cong N(\mathbf{x}_k^i, \mathbf{P}_k^i)$$

where $\hat{\mathbf{x}}_k^i$ \mathbf{P}_k^i are the estimation of mean and covariance of EKF/UKF, $i = 1, \dots, N$.

(iii) The recursive estimate for the importance weights can be written as follows

$$w_k^i = w_{k-1}^i \frac{p(\mathbf{z}_k | \mathbf{x}_k^i) p(\mathbf{x}_k^i | \mathbf{x}_{k-1}^i)}{q(\mathbf{x}_k^i | \mathbf{x}_{k-1}^i, \mathbf{z}_k)} \quad (25)$$

where N is the number of samples. Normalized the importance weights by

$$w_k^i = w_k^i / \sum_{i=1}^N w_k^i \quad (26)$$

(3) Selection or re-sampling. Compute the effective weights according to

$$N_{\text{eff}} = \frac{1}{\sum_{i=1}^N (w_k^i)^2} \quad (27)$$

If $N_{\text{eff}} < N$, resample particles $\{\mathbf{x}_k^i\}_{i=1}^N$ and assign equal weights to them $w_k^i = 1/N$.

(4) Output

$$\mathbf{x}_k = \sum_{i=1}^N \mathbf{x}_k^i w_k^i \quad (28)$$

$$\hat{\mathbf{P}}_k = \sum_{i=1}^N w_k^i (\mathbf{x}_k^i - \hat{\mathbf{x}}_k)(\mathbf{x}_k^i - \hat{\mathbf{x}}_k)^T \quad (29)$$

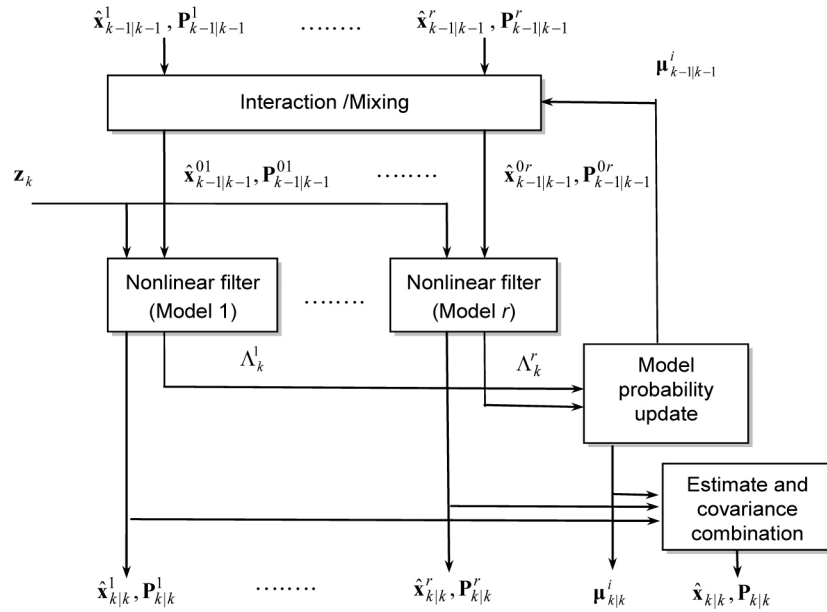


Figure 2. The block diagram of the IMM nonlinear filter algorithm (one cycle with r models).

3. The interacting multiple model (IMM) filter framework

The IMM approach carries out a 'soft switching' among various models by the model probability. The algorithm of IMM-nonlinear filters is introduced to deal with the noise uncertainty and system nonlinearity simultaneously. Figure 2 describes the structure of the IMM estimator for the case of r models.

Let a general system for multiple models in discrete time be described by

$$\mathbf{x}_{k+1} = \mathbf{f}_k(\mathbf{x}_k, M_k) + \mathbf{w}_k(\mathbf{x}_k, M_k) \quad (30)$$

$$\mathbf{z}_k = \mathbf{h}_k(\mathbf{x}_k, M_k) + \mathbf{v}_k(\mathbf{x}_k, M_k) \quad (31)$$

where $\mathbf{f}_k(\cdot)$ and $\mathbf{h}_k(\cdot)$ are the parameterized state transition and measurement functions, \mathbf{x}_k and \mathbf{z}_k are the dynamical state and measurement of the system in model M_k , and the system itself is a Markov chain, \mathbf{w}_k , \mathbf{v}_k are the process noise and measurement noise with means $\bar{\mathbf{w}}_k$, $\bar{\mathbf{v}}_k$ and covariances \mathbf{Q}_k and \mathbf{R}_k , respectively.

3.1 IMM-EKF

An IMM cycle consists of four major stages. The IMM-EKF algorithm is summarized as follows.

(1) Model interaction/mixing. For given states $\mathbf{x}_{k-1}^j = \mathbf{x}_{k-1|k-1}^j$ with corresponding covariances $\mathbf{P}_{k-1}^j = \mathbf{P}_{k-1|k-1}^j$ and mixing probabilities $\mu_{k-1|k-1}^{i|j}$ for every model, the initial condition for the model j is

$$\hat{\mathbf{x}}_{k-1|k-1}^{0j} = \sum_{i=1}^r \hat{\mathbf{x}}_{k-1|k-1}^i \mu_{k-1|k-1}^{i|j}, \quad j = 1, 2, \dots, r \quad (32)$$

The covariance corresponding to the above is:

$$\mathbf{P}_{k-1|k-1}^{0j} = \sum_{i=1}^r \mu_{k-1|k-1}^{i|j} \{ \mathbf{P}_{k-1|k-1}^i + [\hat{\mathbf{x}}_{k-1|k-1}^i - \hat{\mathbf{x}}_{k-1|k-1}^{0j}][\hat{\mathbf{x}}_{k-1|k-1}^i - \hat{\mathbf{x}}_{k-1|k-1}^{0j}]^T \} \quad (33)$$

Calculating the mixing probabilities with mode switching probability matrix π_{ij} and the Gaussian mixing probabilities are computed via the equations:

$$\mu_{k-1|k-1}^{i|j} = \frac{1}{c_j} \pi_{ij} \mu_{k-1}^i, \quad i, j = 1, \dots, r \quad (34)$$

where $\bar{\mathbf{c}}_j$ is a normalization factor is

$$\bar{\mathbf{c}}_j = \sum_{i=1}^r \pi_{ij} \mathbf{P}_{k-1}^i$$

where $\hat{\mathbf{x}}_{k-1|k-1}^{0j}$ and $\hat{\mathbf{P}}_{k-1|k-1}^{0j}$ are the mixed initial condition for mode-matched filter j at time $k-1$. Note that, in this context, i and j denote the corresponding model index respectively; n is the total number of sampling particles.

(2) Filtering. The state estimate equation and covariance equation are used as input to the filter matched to \mathbf{M}_k^j , which uses \mathbf{z}_k to yield $\hat{\mathbf{x}}_{k|k}^j$ and $\mathbf{P}_{k|k}^j$. The likelihood function corresponding to the r filters

$$\Lambda_k^j = p[\mathbf{z}_k | \mathbf{M}_k^j, \mathbf{Z}_{k-1}] \quad (35)$$

are computed using the mixed initial condition as

$$\Lambda_k^j = p[\mathbf{z}_k | \mathbf{M}_k^j, \hat{\mathbf{x}}_{k-1|k-1}^{0j}, \mathbf{P}_{k-1|k-1}^{0j}] \quad (36)$$

(3) Model probability update. Model probability is updated according to the model likelihood and model transition probability governed by the finite-state Markov chain:

$$\mu_k^j = \frac{1}{c} \Lambda_k^j \bar{\mathbf{c}}_j \quad (37)$$

where

$$c = \sum_{j=1}^r \bar{\mathbf{c}}_j \Lambda_k^j \quad (38)$$

and Λ_k^j is a likelihood function given by

$$\Lambda_k^j = \frac{1}{\sqrt{2\pi | \mathbf{P}_{zz}^j |}} \exp \left[-\frac{1}{2} \mathbf{v}_k^{jT} (\mathbf{P}_{zz}^j)^{-1} \mathbf{v}_k^j \right] \quad (39)$$

(4) Combination of state estimation and covariance combination. Combination of the model-conditioned estimates and covariances is done according to the mixture equations

$$\hat{\mathbf{x}}_{k|k} = \sum_{j=1}^r \mu_k^j \hat{\mathbf{x}}_{k|k}^j \quad (40)$$

$$\mathbf{P}_{k|k} = \sum_{j=1}^r \mu_k^j \{ \mathbf{P}_k^j + [\hat{\mathbf{x}}_{k|k}^j - \hat{\mathbf{x}}_{k|k}] [\hat{\mathbf{x}}_{k|k}^j - \hat{\mathbf{x}}_{k|k}]^T \} \quad (41)$$

3.2 IMM-UKF

One cycle of the IMM-UKF is similar to that of the IMM-EKF except that the EKF used in the IMM-EKF is replaced by the UKF, as follows.

(1) Step 1 in UKF. The unscented transform creates $2n+1$ sigma vectors \mathbf{X} (a capital letter) and weighted points W . For state estimation at instant $k-1$, sigma points are generated according to

$$\begin{cases} \mathbf{X}_{0,k-1}^j = \hat{\mathbf{x}}_{k-1}^{0j} \\ \mathbf{X}_{i,k-1}^j = \hat{\mathbf{x}}_{k-1}^{0j} + (\sqrt{(n+\lambda)\mathbf{P}_{k-1}^{0j}})_i^T, \quad i=1,\dots,n \\ \mathbf{X}_{i+n,k-1}^j = \hat{\mathbf{x}}_{k-1}^{0j} - (\sqrt{(n+\lambda)\mathbf{P}_{k-1}^{0j}})_i^T \end{cases} \quad (42)$$

$$W_0^{(m)} = \frac{\lambda}{(n+\lambda)} \quad (43)$$

$$W_0^{(c)} = W_0^{(m)} + (1 - \alpha^2 + \beta) \quad (44)$$

$$W_i^{(m)} = W_i^{(c)} = \frac{1}{2(n+\lambda)}, \quad i=1,\dots,2n \quad (45)$$

(2) Step 2 in UKF. Time update (prediction steps)

$$\zeta_{i,k|k-1}^j = f_j(\mathbf{X}_{i,k-1}^j), \quad i=0,\dots,2n \quad (46)$$

$$\hat{\mathbf{x}}_{k|k-1}^j = \sum_{i=0}^{2n} W_i^{(m)} \zeta_{i,k|k-1}^j \quad (47)$$

$$\begin{aligned} \mathbf{P}_{k|k-1}^j &= \sum_{i=0}^{2n} W_i^{(c)} [\zeta_{i,k|k-1}^j - \hat{\mathbf{x}}_{k|k-1}^j] [\zeta_{i,k|k-1}^j - \hat{\mathbf{x}}_{k|k-1}^j]^T + \mathbf{Q}_{k-1}^j \end{aligned} \quad (48)$$

$$\mathbf{Z}_{i,k|k-1}^j = h(\zeta_{i,k|k-1}^j) \quad (49)$$

$$\hat{\mathbf{z}}_{k|k-1}^j = \sum_{i=0}^{2n} W_i^{(m)} \mathbf{Z}_{i,k|k-1}^j \quad (50)$$

(3) Step 3 in UKF. Measurement update (correction steps)

$$\begin{aligned} \mathbf{P}_{zz}^j &= \sum_{i=0}^{2n} W_i^{(c)} [\mathbf{Z}_{i,k|k-1}^j - \hat{\mathbf{z}}_{k|k-1}^j][\mathbf{Z}_{i,k|k-1}^j - \hat{\mathbf{z}}_{k|k-1}^j]^T + \mathbf{R}_k^j \\ & \quad (51) \end{aligned}$$

$$\mathbf{P}_{xz}^j = \sum_{i=0}^{2n} W_i^{(c)} [\mathbf{Z}_{i,k|k-1}^j - \hat{\mathbf{z}}_{k|k-1}^j][\mathbf{Z}_{i,k|k-1}^j - \hat{\mathbf{z}}_{k|k-1}^j]^T \quad (52)$$

$$\mathbf{K}_k^j = \mathbf{P}_{xz}^j (\mathbf{P}_{zz}^j)^{-1} \quad (53)$$

$$\hat{\mathbf{x}}_{k|k}^j = \hat{\mathbf{x}}_{k|k-1}^j + \mathbf{K}_k^j \mathbf{v}_k^j \quad (54)$$

where $\mathbf{v}_k^j = \mathbf{Z}_k - \hat{\mathbf{z}}_{k|k-1}^j$.

$$\mathbf{P}_{k|k}^j = \mathbf{P}_{k|k-1}^j - \mathbf{K}_k^j \mathbf{P}_{zz}^j \mathbf{K}_k^{jT} \quad (55)$$

The samples are propagated through true nonlinear equations; the linearization is unnecessary, i.e., calculation of Jacobian is not required.

3.3 IMM-EPF

The extended particle filter (EPF) with IMM framework yields the IMM-EPF. Integrating a bank of extended Kalman particle filters with the IMM method, an IMM-EKPF is obtained [12]; integrating a bank of unscented particle filters with the IMM method, an IMM-UPF is obtained. One cycle of the IMM-EPF (IMM-EKPF or IMM-UPF) is similar to that of the IMM-EKF or IMM-UKF except that the filters are replaced by the extended particle filters (i.e., EKPF or UPF).

4. Results and discussion for GPS/INS integration

The differential equations describing the two-dimensional inertial navigation state, where two accelerometers and one gyroscope are involved, are:

$$\begin{bmatrix} \dot{n} \\ \dot{e} \\ \dot{v}_n \\ \dot{v}_e \\ \dot{\psi} \end{bmatrix} = \begin{bmatrix} v_n \\ v_e \\ a_n \\ a_e \\ \omega_r \end{bmatrix} = \begin{bmatrix} v_n \\ v_e \\ \cos(\psi)a_u - \sin(\psi)a_v \\ \sin(\psi)a_u + \cos(\psi)a_v \\ \omega_r \end{bmatrix} \quad (56)$$

where $[a_u, a_v]$ are the measured accelerations in the body frame, ω_r is the measured yaw rate in the body frame. The error model for INS is constructed by the navigation states augmented by the accelerometer biases and gyroscope drift:

$$\frac{d}{dt} \begin{bmatrix} \delta n \\ \delta e \\ \delta v_n \\ \delta v_e \\ \delta \psi \\ \delta a_u \\ \delta a_v \\ \delta \omega_r \end{bmatrix} = \begin{bmatrix} 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & -a_e & \cos(\psi) & -\sin(\psi) & 0 \\ 0 & 0 & 0 & 0 & -a_n & \sin(\psi) & \cos(\psi) & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} \delta n \\ \delta e \\ \delta v_n \\ \delta v_e \\ \delta \psi \\ \delta a_u \\ \delta a_v \\ \delta \omega_r \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ u_{acc} \\ u_{acc} \\ u_{gyro} \\ u_{acc}^b \\ u_{acc}^b \\ u_{gyro}^b \end{bmatrix} \quad (57)$$

which were utilized in the integration Kalman filter as the inertial error model. In Eq. (57), δn and δe represent the east, and north position errors; δv_n and δv_e represent the east, and north velocity errors; $\delta \psi$ represents yaw angle; δa_u , δa_v , and $\delta \omega_r$ represent the accelerometer biases and gyroscope drift, respectively. The measurement model is

$$\mathbf{z}_k = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} \delta n \\ \delta e \\ \delta v_n \\ \delta v_e \\ \delta \psi \\ \delta a_u \\ \delta a_v \\ \delta \omega_r \end{bmatrix} + \begin{bmatrix} v_{pn} \\ v_{pw} \\ v_{vn} \\ v_{ve} \\ v_{hdg} \end{bmatrix} \quad (58)$$

The GPS-derived yaw angle measurement can be obtained through

$$\psi = \tan^{-1}\left(\frac{v_e}{v_n}\right) \quad (59)$$

Simulation and experimental tests have been carried out to evaluate the navigation performance for the proposed approach in comparison with the conventional approaches.

A. Simulation example. The navigation integration software was developed by the authors using the Matlab® software. The commercial software Satellite Navigation (SATNAV) toolbox [22] by GPSof LLC was employed for generating the satellite positions and pseudoranges.

Figure 3 shows the schematic illustration of simulated trajectory. The trajectory can be divided mainly into six time intervals (or segments) according the dynamic characteristics, as indicated in the figure. The vehicle was simulated to conduct constant-velocity straight-line during the three time intervals, 0-45, 135-180 and 225-270s, all at a constant speed of 10π m/s. Furthermore, it conducted circular motion with radius 450 meters during 45-135s (counterclockwise); with radius 900 meters during 180-225s (counterclockwise) and 270-450s (clockwise) where high dynamic and medium maneuverings are involved.

Assume that the differential GPS (DGPS) mode is used and only multipath and receiver thermal noise are considered. The measurement noise variances r_{ρ_i} are assumed *a priori* known, which is set as $9m^2$. Let each of the white-noise spectral amplitudes that drive the random walk position states be $S_p = 0.003 (m/sec^2)/rad/sec$. Also, let the clock model spectral amplitudes be $S_f = 0.4(10^{-18})sec$ and $S_g = 1.58(10^{-18})sec^{-1}$. The measurement noise covariance matrix is given by $R_k = 15 \times I_{m \times m}$.

100 particles were used whenever the particle filter (for both EKPF and UPF) is used. For the two models of the IMM algorithms, two process noise

covariance matrices are used: $Q_k^{small} = (1e-3) \times I_{8 \times 8}$, $Q_k^{large} = 5Q_k^{small}$. The mode transition matrix is given by

$$\pi = \begin{bmatrix} 0.99 & 0.01 \\ 0.01 & 0.99 \end{bmatrix} \quad (60)$$

The UKF parameters are: $\alpha = 1$, $\kappa = 0$, $\beta = 2$. Figure 4 gives the results for the simulated case for various nonlinear filters, in single-model and IMM frameworks. The corresponding model probability is shown in Figure 5. Table 1 provides the comparison of position errors using nonlinear filters and IMM nonlinear filters for the simulation case.

B. Road test. The road test was conducted in Hsinchu, Taiwan. The test drive results were obtained by post processing of logged sensor data using the Matlab® software. The GPS data was used as reference trajectory for comparative study on navigation performance for various dynamic environments using various nonlinear with/without IMM framework. The Micro-Electro-Mechanical-Systems (MEMS) inertial sensors used were the BOSCH SMB380 Triaxial acceleration sensors and the XV-8000CB Ultra Miniature Size Vibration Gyro Sensors. The GPS data was collected at 1 Hz and the IMU has a data rate of 10 Hz. The driving speed is around 16~25 km/hr for most of the time. Two process noise covariance matrices are used:

$$Q_k^{large} = 10Q_k^{small}$$

where

$$Q_k^{small} = \begin{bmatrix} Q_1 & & 0 \\ & Q_2 & \\ 0 & & Q_3 \end{bmatrix}, \quad Q_1 = (5e-3) \times I_{3 \times 3}, \\ Q_2 = (5e-6) \times I_{3 \times 3}, \quad Q_3 = (5e-3) \times I_{2 \times 2}.$$

In this experiment, Figure 6 shows the trajectory for the road test.

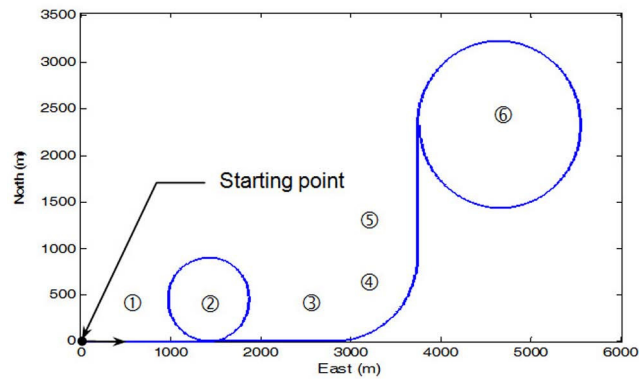


Figure 3. Schematic illustration of simulated trajectory.

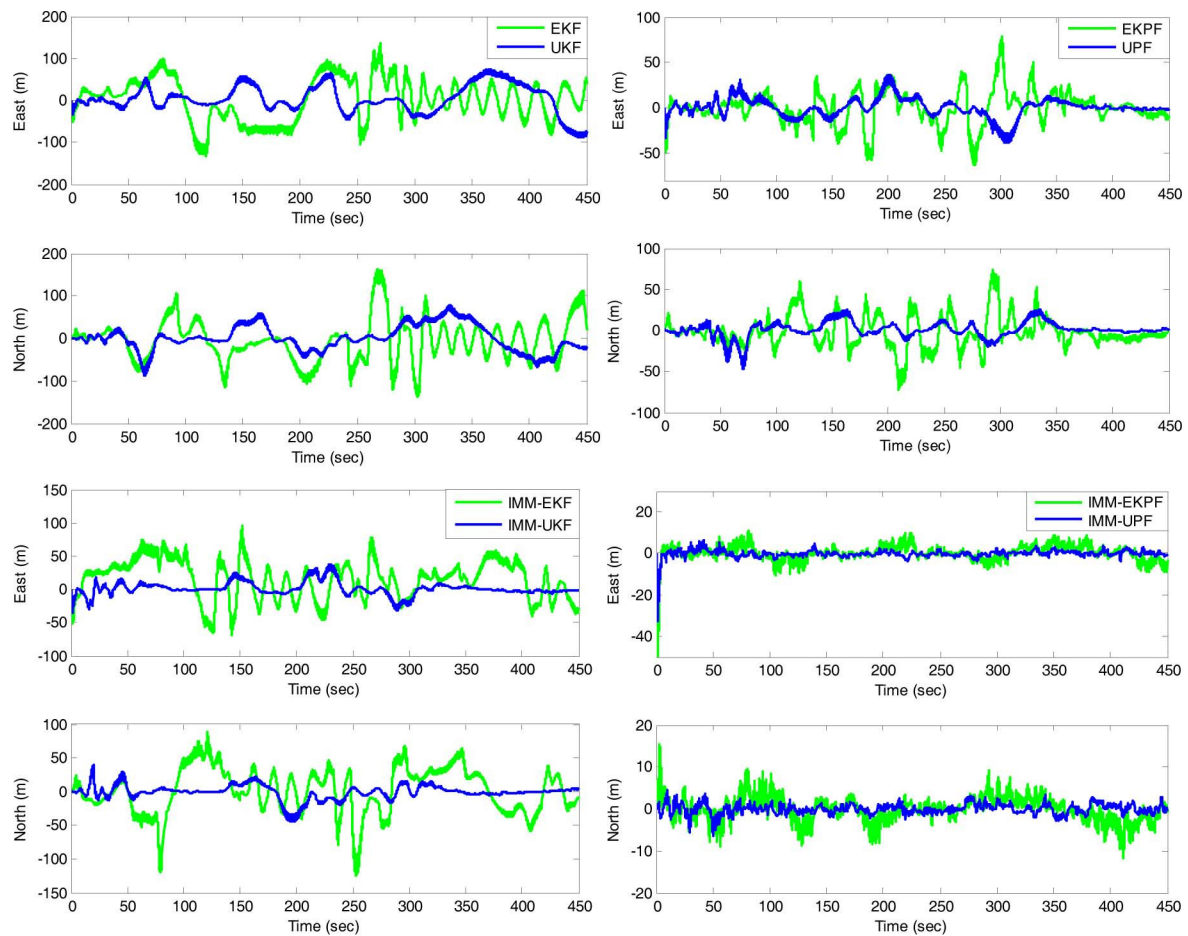


Figure 4. Results for the simulated case.

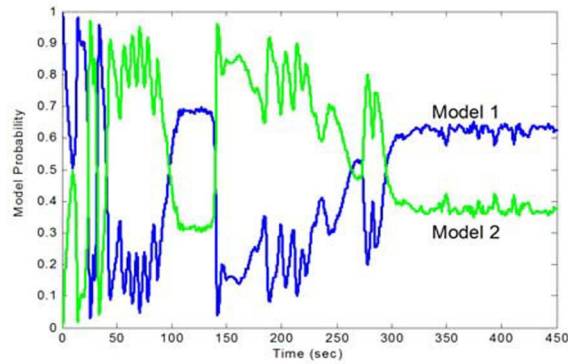


Figure 5. Model probability.

	MSE (m^2)		Time consumption (sec)
	North	East	
EKF	2309.4	2573.9	2.781
UKF	904.3693	1058.6	3.594
EKPF	374.1817	326.4786	73.719
UPF	89.6852	110.9177	297.515

	MSE (m^2)		Time consumption (sec)
	North	East	
IMM-EKF	1158.0	858.1319	9.969
IMM-UKF	112.0694	96.6876	6.672
IMM-EKPF	8.4843	15.6924	137.016
IMM-UPF	2.1015	5.0883	588.000

Table 1. Comparison of position errors using nonlinear filters and IMM nonlinear filters – simulation results.

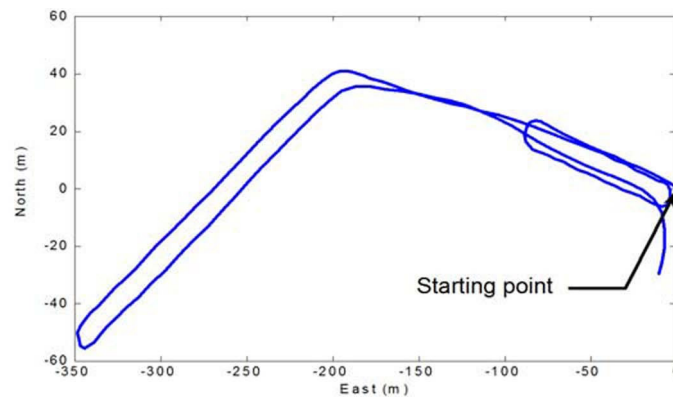


Figure 6. The trajectory for road test.

The results are illustrated in Figure 7. Table 2 shows comparison of position errors and computation time using various nonlinear filters and IMM nonlinear filters. When trials with sharp turns and abrupt maneuvers are performed, the performance enhancement becomes remarkable. It can also be seen that the computational load for the particle filter has been rapidly increased. The number of particles used in the test is 100. The number of sigma points in the UKF is $2n+1$, which is 17 in this paper.

5. Conclusions

This paper has presented comparative study on navigation performance for various nonlinear filters, including EKF UKF EKPF UPF.

Furthermore, the nonlinear filters have been incorporated into the interacting multiple model framework, resulting in the IMM-EKF IMM-UKF IMM-EKPF IMM-UPF algorithms. The problem of filter parametric adaptation can be regarded as the generalization of structural adaptation. To solve the possible degradation problem caused by noise uncertainty and modeling error, the IMM algorithm can be employed for dynamically adjusting the process noise, and accordingly enhancing the estimation accuracy. However, selection of the nonlinear filters in the IMM framework leads to different levels of computational burden. The IMM-PF requires significantly large computational time as compared to the other two algorithms. Trade-off needs to be made when selecting a suitable algorithm for a specific purpose.

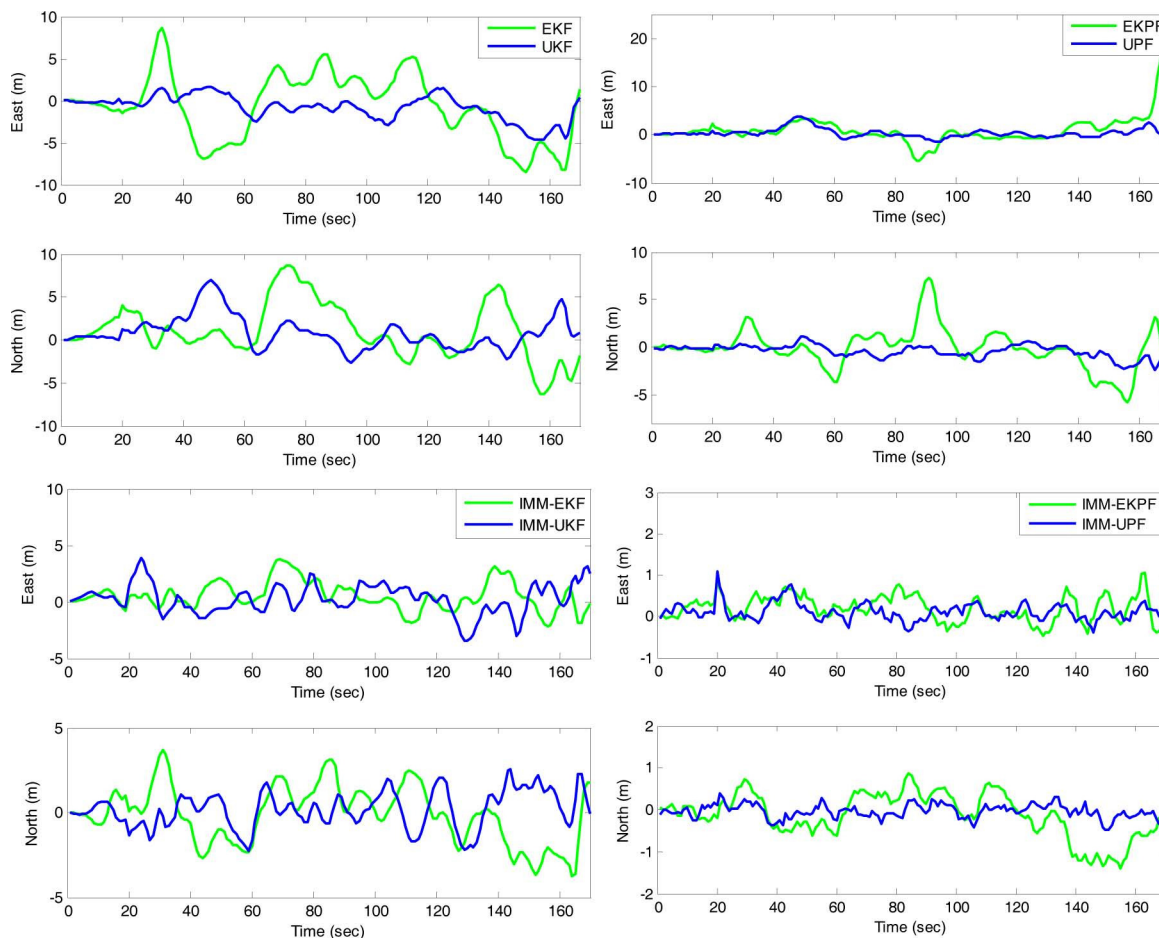


Figure 7. Results for the road test.

	MSE (m^2)		Time consumption (sec)
	North	East	
EKF	11.3684	16.0626	1.718
UKF	4.3407	2.9309	2.125
EKPF	3.4810	6.3444	34.016
UPF	1.0068	0.6104	157.766

	MSE (m^2)		Time consumption (sec)
	North	East	
IMM-EKF	2.9483	1.5776	6.672
IMM-UKF	1.1663	1.7736	3.719
IMM-EKPF	0.2832	0.0982	74.235
IMM-UPF	0.0271	0.0456	324.781

Table 2. Comparison of position errors using nonlinear filters and IMM nonlinear filters – experiment results.

Acknowledgements

Funding for this work was provided by the National Science Council of the Republic of China under grant numbers NSC 97-2221-E-019-012 and NSC 98-2221-E-019-021-MY3.

References

- [1] J. Farrell and M. Barth, "The Global Positioning System and Inertial Navigation", New York: McCraw-Hill, 1999.
- [2] R. G. Brown and P. Y. C. Hwang, "Introduction to Random Signals and Applied Kalman Filtering", New York: John Wiley & Sons, 1997.
- [3] L. T. Grigorie and R. M. Botez, "Modeling and Numerical Simulation of an Algorithm for the Inertial Sensors Errors Reduction and for the Increase of the Strapdown Navigator Redundancy Degree in a Low Cost Architecture," Transactions of the Canadian Society for Mechanical Engineering, vol. 34, no. 1, 2010, pp.1-16.
- [4] S. J. Julier et al., A New Approach for Filtering Nonlinear System, the American Control Conference, 1995, pp.1628-1632.
- [5] S. J. Julier, The Scaled Unscented Transformation, Proc. of the American Control Conf., Anchorage, 2002, pp.4555-4559.
- [6] S. J. Julier et al., "A New Method for the Nonlinear Transformation of Means and Covariances in Filters and Estimators," IEEE Transactions on Automatic Control, vol. 5, no. 3, 2000, pp. 477-482.
- [7] Wan, E. A., van der Merwe, R., The Unscented Kalman Filter for Nonlinear Estimation, Adaptive Systems for Signal Processing, Communication and Control (AS-SPCC) Symposium, Alberta, Canada, 2000, pp.153-156.
- [8] R. van der Merwe et al., "The Unscented Particle Filter", Technical Report CUED / F-INFENG / TR 380, Cambridge University Engineering Department, 2000.
- [9] P. Aggarwal et al., "Hybrid Extended Particle Filter (HEPF) for Integrated Inertial Navigation and Global Positioning Systems," Meas. Sci. Technol., vol. 20, 2009, (9pp), doi:10.1088/0957-0233/20/5/055203.
- [10] T. T. Duong and K. W. Chiang, Non-linear, Non-Gaussian Estimation for INS/GPS Integration, The First International Conference on Engineering and Technology Innovation (IJETI), Kenting, Taiwan, November 11-15, 2011.
- [11] A. Doucet et al., "An Introduction to Sequential Monte Carlo methods," in A. Doucet et al. (eds), Sequential Monte Carlo Methods in Practice, New York: Springer, 2001, pp.3-14.

- [12] N. Yang et al., "An Interacting Multiple Model Particle Filter for Manoeuvring Target Location," *Meas. Sci. Technol.*, vol. 17, 2006, pp.1307–1311.
- [13] R. A. López et al., Temperature Control of Continuous Chemical Reactors Under Noisy Measurements and Model Uncertainties, *Journal of Applied Research and Technology*, vol. 10, no. 3, 2012, pp.428-446.
- [14] R. K. Mehra, "On-line Identification of Linear Dynamic Systems with Applications to Kalman Filtering," *IEEE Transactions on Automatic Control*, vol. AC-16, 1970, pp.12-21.
- [15] A. H. Mohamed and K. P. Schwarz, "Adaptive Kalman Filtering for INS/GPS," *Journal of Geodesy*, vol. 73, 1999, pp.193-203.
- [16] X. R. Li and Y. Bar-Shalom, "Estimation with Applications to Tracking and Navigation", New York: John Wiley & Sons, 1993.
- [17] Y. S. Kim and K. S. Hong, "An IMM Algorithm for Tracking Maneuvering Vehicles in an Adaptive Cruise Control Environment," *International Journal of Control, Automation, and Systems*, vol. 2, no. 3, 2004, pp.310-318.
- [18] B. J. Lee et al., "Fuzzy-logic-based IMM Algorithm for Tracking a Manoeuvring Target," *IEE Proceedings-Radar Sonar Navig.*, vol. 152, no. 1, 2005, pp.16-22.
- [19] X. R. Li and Y. Bar-Shalom, "Design of an Interacting Multiple Model Algorithm for Air Traffic Control Tracking," *IEEE Transactions on Control System Technology*, vol. 1, no. 3, 1993, pp.186-194.
- [20] G. Chen and M. Harigae, Using IMM Adaptive Estimator in GPS Positioning, 40th SICE Annual Conference, SICE 2001, International Session Papers, Nagoya, Japan, 2001, pp.25-27.
- [21] X. Lin et al., "Enhanced Accuracy GPS Navigation Using the Interacting Multiple Model Estimator," *IEEE Aerospace Conference*, Montana, vol. 4, 2001, pp.1911-1923.
- [22] GPSoft LLC, Navigation System Integration and Kalman Filter Toolbox User's Guide, 2005.