Nonlinear Filtering with IMM Algorithm for Ultra-Tight GPS/INS Integration

Dah-Jing Jwo¹,², Chia-Wei Hu² and Chien-Hao Tseng³

¹ Department of Communications, Navigation and Control Engineering, National Taiwan Ocean University, Keelung, Taiwan
² Department of Planning and Evaluation, National Science Council, Taipei, Taiwan
³ National Centre for High-Performance Computing, Taichung, Taiwan

* Corresponding author E-mail: djjwo@mail.ntou.edu.tw

Received 25 Apr 2012; Accepted 5 Mar 2013

Abstract This paper conducts a performance evaluation for the ultra-tight integration of a Global positioning system (GPS) and an inertial navigation system (INS), using nonlinear filtering approaches with an interacting multiple model (IMM) algorithm. An ultra-tight GPS/INS architecture involves the integration of in-phase and quadrature components from the correlator of a GPS receiver with INS data. An unscented Kalman filter (UKF), which employs a set of sigma points by deterministic sampling, avoids the error caused by linearization as in an extended Kalman filter (EKF). Based on the filter structural adaptation for describing various dynamic behaviours, the IMM nonlinear filtering provides an alternative for designing the adaptive filter in the ultra-tight GPS/INS integration. The use of IMM enables tuning of an appropriate value for the process of noise covariance so as to maintain good estimation accuracy and tracking capability. Two examples are provided to illustrate the effectiveness of the design and demonstrate the effective improvement in navigation estimation accuracy. A performance comparison among various filtering methods for ultra-tight integration of GPS and INS is also presented. The IMM based nonlinear filtering approach demonstrates the effectiveness of the algorithm for improved positioning performance.

Keywords GPS, INS, Ultra-Tight Integration, Interacting Multiple Model, Unscented Kalman Filter

1. Introduction

Global positioning systems (GPS) [1] are capable of providing accurate position information. Unfortunately, the data is prone to jamming or being lost due to the limitations of electromagnetic waves, which form the fundamental of their operation. The inertial navigation system (INS) is a self-contained system that integrates three acceleration components and three angular velocity components. However, the error in position coordinates increases unboundedly as a function of time. Due to the inherent complementary operational characteristics of GPS and INS systems, the synergy of both systems has been widely explored [1, 2]. The GPS/INS integration gives an adequate solution to provide a navigation system that has superior performance in comparison with either a GPS or an INS stand-alone system. The design of GPS/INS integrated navigation systems heavily depends on the design of the sensor fusion method.

The Kalman filter (KF) [2, 3] is a well-known sequential data assimilation scheme for solving the Wiener problem.
The Kalman filter not only works well in practice, but also is theoretically attractive because it has been shown that the filter that minimizes the variance of the estimation mean square error. While employed as the navigational state estimator, the extended Kalman filter (EKF), the nonlinear version of the KF, has been one of the promising approaches for estimating vehicle state variables and suppressing navigation measurement noise. In designing a navigation filter, there exists some model uncertainty, which cannot be expressed by a linear state-space model. The linear model includes modelling errors since the actual vehicle motions are non-linear processes.

It should be noted that the Kalman filter attempts to minimize the variance of the estimation errors and provides an optimal result only if the system model, the system initial conditions and the noise characteristics can be specified a priori, the perfect a priori knowledge of both the process noise and measurement noise are Gaussian white processes and the noise statistics for both disturbances are completely known. In Kalman filter designs, cases where the theoretical behaviour of a filter and its actual behaviour do not agree may lead to divergence problems. In various circumstances where there are uncertainties 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. It is very often the case that little a priori knowledge is available concerning the manoeuvre. One way to take them into account is to consider a nominal model affected by uncertainty.

As for system nonlinearity, the EKF might suffer from performance degradation and divergence problems due to the linearization process. To address the nonlinearity, other filters such as the unscented Kalman filter (UKF) [4-7] are becoming popular. Proposed by Julier et al. [4] to address nonlinear state estimation in the context of control theory, the UKF is a nonlinear, distribution approximation method that uses a finite number of carefully chosen sigma points to propagate the probability of state distribution. Unlike the conventional EKF, which achieves first-order accuracy with the linearization process using Jacobian matrices, the UKF employs a minimal set of sigma points using a deterministic sampling approach and at least the second order accuracy of the posterior mean and covariance can be captured. The UKF approximates the Gaussian distribution by a set of deterministically selected samples called sigma points, which are propagated through the true non-linear models to capture the true mean and covariance of the transformed distribution. The UKF makes a Gaussian approximation with a limited number of sigma points through the Unscented Transform (UT) [5, 6]. Through the nonlinear dynamics of the system, the true mean and covariance of the Gaussian random variable (GRV) are completely captured with a minimal set of samples. The basic premise behind the UKF is it is easier to approximate a Gaussian distribution than it is to approximate an arbitrary nonlinear function. The posterior mean and covariance can be captured accurately to the second order of Taylor series expansion for any nonlinear system. One of the remarkable merits is that the overall computational complexity of the UKF is the same as that of the EKF.

An adaptive mechanism that dynamically identifies uncertainties or modelling errors can be adopted for the various circumstances where there are uncertainties in the system model and noise description and the assumptions of 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. For example, when in the modelling step, some phenomena are disregarded. The suboptimal configuration is typically based on a simplified error state dynamic/measurement model. One way to take this into account is to consider a nominal model affected by uncertainty. To fulfil the requirement of achieving filter optimality or to prevent the divergence problems of the Kalman filter, the so-called adaptive Kalman filter (AKF) approach [8-10] has been a promising strategy and has been widely explored for dynamically adjusting the parameters of the supposedly optimum filter, based on the estimates of the unknown parameters for on-line estimation of motion, as well as the available signal and noise statistics data. Many efforts have been made to improve the estimation of the covariance matrices based on the innovation-based estimation approach. The two major approaches that have been proposed for AKF are the multiple model adaptive estimate (MMAE) and the innovation adaptive estimation (IAE).

The AKF approach is based on parametric adaptation, while the interacting multiple model (IMM) approach [11] is based on filter structural adaptation (model switching). The IMM algorithm has a configuration that runs several model-matched state estimation filters in parallel, which exchange information (interact) at each sampling time. In the IMM approach, multiple models are developed. In each model a filter runs and the IMM algorithm makes uses of model probabilities to weight the inputs and output of a bank of parallel filters at each time instant. Based on the filter structural adaptation (model switching) for describing various dynamic behaviours, the IMM algorithm was originally applied to target tracking [11, 12] and was only recently extended to GPS navigation [13, 14] and the integration of GPS and INS [15-21]. The IMM algorithm based on the augmented Kalman filter (AUKF) has been proposed to increase the accuracy of IMM algorithm in the presence of low and medium manoeuvres [19]. The AUKF considers the acceleration as an additive state term in the state space.
The received satellite signal can be presented as:

\[ y(t) = A \cdot CA(t) \cdot D(t) \cos \left( 2\pi f_c (t - \tau) + \theta_0 \right) + \eta \]  

where \( A \) is the signal amplitude, \( CA \) is the C/A code sequence, \( D(t) \) is the navigation data, \( \tau \) is the propagation delay between the satellite and receiver, \( f_c \) is the carrier frequency, \( \theta_0 \) is the initial carrier phase and \( \eta \) is the Gaussian noise. The signals from the receiver correlator, in-phase (I) and quadrature (Q), form the measurements of the filters:

\[ I = \int_{kT}^{(k+1)T} [\cos(\dot{\theta} + \Delta)] A \cos(\omega_c t + \theta) + \eta_I \]  

\[ Q = \int_{kT}^{(k+1)T} [\sin(\dot{\theta} + \Delta)] A \cos(\omega_c t + \theta) + \eta_Q \]  

After simplification, we have:

\[ I = \frac{A}{2\omega_c} [\sin(\omega_c (k+1)T + \theta_e) - \sin(\omega_c kT + \theta_e)] + \eta_I \]  

\[ Q = -\frac{A}{2\omega_c} [\cos(\omega_c (k+1)T + \theta_e) - \cos(\omega_c kT + \theta_e)] + \eta_Q \] 

where \( \eta_I \) and \( \eta_Q \) represent the noise components for the I and Q channels, respectively, \( \omega_c = f - \omega = 2\pi(\bar{f} - f) \) is the frequency error tracked by the PLL, \( \theta_e = \hat{\theta} - \theta \) is the phase error tracked by the PLL and \( T \) is the pre-detection integration time. The expectation of the integral leads to:

\[ E[I] = \frac{A}{2\omega_c} [\sin(\omega_c (k+1)T + \theta_e) - \sin(\omega_c kT + \theta_e)] \]  

\[ E[Q] = -\frac{A}{2\omega_c} [\cos(\omega_c (k+1)T + \theta_e) - \cos(\omega_c kT + \theta_e)] \] 

Equation (4) shows that \( E[I] \) and \( E[Q] \) depend on the errors of the carrier frequency and phase. These errors can be described in terms of the position and velocity:

\[ \omega_c = \frac{2\pi f_c}{c} |\hat{v}_u - v_u| = \frac{\alpha_c}{c} V_e \]  

\[ \theta_e = \frac{2\pi f_c}{c} \left( |\hat{v}_u - v_u| t - |\hat{R}_u - R_u| \right) = \frac{\alpha_c}{c} (V_e - R_e) \] 

where \( R_u \) and \( v_u \) are the measured position and velocity of the receiver, \( \hat{R}_u \) and \( \hat{v}_u \) are the receiver estimates of position and velocity and \( R_e = |\hat{R}_u - R_u| \) and \( V_e = |\hat{v}_u - v_u| \) represent the errors of the position and velocity, given by:

\[ R_e = \sqrt{x_e^2 + y_e^2 + z_e^2} = \sqrt{(\hat{x} - x)^2 + (\hat{y} - y)^2 + (\hat{z} - z)^2} \]  

\[ V_e = \sqrt{x_e^2 + y_e^2 + z_e^2} = \sqrt{(\hat{x} - \hat{x})^2 + (\hat{y} - \hat{y})^2 + (\hat{z} - \hat{z})^2} \]
The INS equations describing the three-dimensional inertial navigation state are:

\[ \mathbf{V}_n = \mathbf{C}_m f - \left[ \Omega (\omega_{in}^0) + 2 \Omega (\omega_{in}^0) \right] \mathbf{V}_n + \mathbf{g}_n^0 \]  \tag{7}

The error model employed for INS is a terrestrial INS psi-angle error model [22]:

\[
\begin{align*}
\delta t &= -\omega_{en} \times \delta r + \delta \nu \\
\delta \nu &= -(\omega_{in} + \omega_{in}) \times \delta \nu - \psi \times f + \epsilon_a \\
\psi &= \omega_{in} \times \nu + \epsilon_g
\end{align*}
\]  \tag{8}

where \( \delta r \) is the position error vector, \( \delta \nu \) is the velocity error vector, \( \psi \) is the attitude error vector, \( \omega_{en} \) is the true rate of the local geographic frame with respect to the Earth frame, \( \omega_{in} \) is the earth rate vector with respect to the local geographic frame, \( \omega_{in} \) is the true rate of the local geographic frame with respect to the inertial frame, \( f \) is the specific force vector, \( \epsilon_a \) is the accelerometer error vector and \( \epsilon_g \) is the gyro drift rate vector. The state vector includes 17 states: 3 inertial error states each in position, velocity, attitude, accelerometer bias, gyro bias and 1 state each for receiver clock bias and drift:

\[ \mathbf{x}_k = [x, y, z, x', y', z', \psi, \nu_x, \nu_y, \nu_z, \alpha_x, \alpha_y, \alpha_z, \beta_x, \beta_y, b, d]^T \]

As shown in Figure 1, the signals from the receiver correlators, I and Q, form the measurements of the filters. The parameters I/Q and the position/velocity can be related by the following expressions:

\[
\begin{align*}
\text{d}E[I] &= \frac{1}{2} \left[ \frac{\partial E[I]}{\partial \theta_e} \frac{\partial \theta_e}{\partial x} + \frac{\partial E[I]}{\partial \omega_e} \frac{\partial \omega_e}{\partial x} \right] \text{d}x \\
\text{d}E[Q] &= \frac{1}{2} \left[ \frac{\partial E[Q]}{\partial \theta_e} \frac{\partial \theta_e}{\partial x} + \frac{\partial E[Q]}{\partial \omega_e} \frac{\partial \omega_e}{\partial x} \right] \text{d}x
\end{align*}
\]  \tag{9a-b}

In the case that EKF is utilized, the measurements are:

\[ \mathbf{z}_k = [I + \text{d}I, Q + \text{d}Q], \quad [-I - \eta_i, Q - \eta_Q], = [\text{d}I + \eta_i, \text{d}Q + \eta_Q] \]

with the corresponding measurement matrix:

\[
\mathbf{H}_k = \begin{bmatrix}
\mathbf{h}_{i1} & \mathbf{h}_{i2} & \mathbf{h}_{i3} & \mathbf{h}_{i4} & \mathbf{h}_{i5} & \mathbf{h}_{i6} & \mathbf{h}_{i7} \\
\vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\
\mathbf{h}_{Q1} & \mathbf{h}_{Q2} & \mathbf{h}_{Q3} & \mathbf{h}_{Q4} & \mathbf{h}_{Q5} & \mathbf{h}_{Q6} & \mathbf{h}_{Q7} \\
\vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\
\mathbf{h}_{Q17} & \mathbf{h}_{Q18} & \mathbf{h}_{Q19} & \mathbf{h}_{Q20} & \mathbf{h}_{Q21} & \mathbf{h}_{Q22} & \mathbf{h}_{Q23} \\
\end{bmatrix}_{1 \times 17} \tag{11}
\]

In Equation (10), \( \text{d}I \) and \( \text{d}Q \) denote the deviations in the INS predicted I and Q measurements caused by the inertial sensor errors, \( \eta_i \) and \( \eta_Q \) are the noise components of I and Q in the GPS signal measurements and \( i \) (i = 1...m) represents the number of channels being tracked. The derivatives in the Jacobian matrix of Equation (11) are given by:

\[
\begin{align*}
\frac{\partial E[I]}{\partial \theta_e} &= \frac{A}{2 \omega_e} \left[ \cos(\omega_k (k+1)T + \theta_e) - \cos(\omega_k kT + \theta_e) \right] \\
\frac{\partial E[I]}{\partial \omega_e} &= -\frac{A}{2 \omega_e} \left[ \sin(\omega_k (k+1)T + \theta_e) - \sin(\omega_k kT + \theta_e) \right] \\
\frac{\partial E[Q]}{\partial \theta_e} &= \frac{A}{2 \omega_e} \left[ \cos(\omega_k (k+1)T + \theta_e) - \cos(\omega_k kT + \theta_e) \right] \\
\frac{\partial E[Q]}{\partial \omega_e} &= -\frac{A}{2 \omega_e} \left[ \sin(\omega_k (k+1)T + \theta_e) + \sin(\omega_k kT + \theta_e) \right] \tag{12-15}
\end{align*}
\]
Taking derivatives of $\theta$ and $\omega$ with respect to the three components of position gives:

$$\frac{\partial \theta}{\partial x} = \frac{\omega x_c}{c R_e}; \quad \frac{\partial \theta}{\partial y} = \frac{\omega y_c}{c R_e}; \quad \frac{\partial \theta}{\partial z} = \frac{\omega z_c}{c R_e}$$

and:

$$\frac{\partial \omega_x}{\partial x} = 0; \quad \frac{\partial \omega_y}{\partial y} = 0; \quad \frac{\partial \omega_z}{\partial z} = 0$$

respectively. Furthermore, taking derivatives of $\theta$ and $\omega$ with respect to the velocities gives:

$$\frac{\partial \theta}{\partial x} = -\omega \dot{x}_c T; \quad \frac{\partial \theta}{\partial y} = -\omega \dot{y}_c T; \quad \frac{\partial \theta}{\partial z} = -\omega \dot{z}_c T$$

and:

$$\frac{\partial \omega_x}{\partial x} = -\omega \dot{x}_c; \quad \frac{\partial \omega_y}{\partial y} = -\omega \dot{y}_c; \quad \frac{\partial \omega_z}{\partial z} = -\omega \dot{z}_c$$

respectively.

where $E[\cdot]$ represents expectation and superscript “T” denotes matrix transpose. $Q_k$ is the process noise covariance matrix and $R_k$ is the measurement noise covariance matrix. The symbol $\delta_{ik}$ 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 follows.

1. Initialize state vector and state covariance matrix: $\hat{x}_0$ and $P_0$

2. Compute Kalman gain matrix from state covariance and estimated measurement covariance:

$$K_k = P_k \Phi_k^T [H_k \Phi_k^T + R_k]^{-1}$$

(17)

3. Multiply prediction error vector by Kalman gain matrix to obtain state correction vector and update state vector:

$$\hat{x}_k = \hat{x}_k + K_k [z_k - \hat{z}_k], \text{ with } \hat{z}_k = h(\hat{x}_k, k)$$

(18)

4. Update error covariance

$$P_k = [I - K_k H_k] P_k$$

(19)

5. Predict new state vector and state covariance matrix

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

(20)

$$P_{k+1} = \Phi_k P_k \Phi_k^T + Q_k$$

(21)

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

$$\Phi_k = \frac{\partial f_k}{\partial x} \bigg|_{\hat{x}_k}$$

(22)

Equations (17-19) are measurement update equations and Equations (20-21) are the time update equations of the algorithm from $k$ to step $k+1$. These equations incorporate a measurement value into a priori estimation to obtain an improved a posteriori estimation. In the above equations, $P_k$ is the error covariance matrix defined by $E[(x_k - \hat{x}_k)(x_k - \hat{x}_k)^T]$, in which $\hat{x}_k$ is an estimation of the system state vector $x_k$ and the weighting matrix $K_k$ is generally referred to as the Kalman gain matrix. The Kalman filter algorithm starts with an initial condition value, $\hat{x}_0$ and $P_0$. When new measurement $z_k$ becomes available with the progression of time, the estimation of states and the corresponding error covariance would follow recursively ad infinity. Further detailed discussion on the EKF can be referred to in [2, 3].

---

**Figure 1.** Illustration of the I and Q components as the measurements of the ultra-tightly coupled GPS/INS integration.

3. The Interacting Multiple Model Unscented Kalman Filter

The nonlinear system governed by the nonlinear stochastic difference equations can be written as:

$$x_{k+1} = f(x_k, k) + w_k$$

(16a)

$$z_k = h(x_k, k) + v_k$$

(16b)

where $x_k$ and $w_k$ represent the state vector and process noise vector, respectively and $z_k$ and $v_k$ are the measurement vector and the measurement noise vector, respectively. The vectors $w_k$ and $v_k$ are zero mean Gaussian white sequences and have zero cross-correlation with each other:

$$E[w_i v_j^T] = Q_k \delta_{ik}; \quad E[v_i v_j^T] = R_k \delta_{ik};$$

$$E[w_i v_j^T] = 0 \quad \text{for all } i \text{ and } k$$

www.intechopen.com

Dah-Jing Jwo, Chia-Wei Hu and Chien-Hao Tseng: Nonlinear Filtering with IMM Algorithm for Ultra-Tight GPS/INS Integration
3.1 The Unscented Kalman Filter

A The unscented transformation

The first step in the UKF is to sample the prior state distribution, i.e., generate the sigma points through unscented transformation (UT). Figure 2 illustrates the true means and covariances as compared to those obtained by the mapping of the UKF versus that of the EKF. The dot-line ellipse represents the true covariance. The UKF is implemented through the transformation of the nonlinear function \( f() \), shown as the solid-line ellipse in the top portion of the figure. The EKF is accomplished through the Jacobian \( F = \frac{\partial f}{\partial x} \), shown as the solid-line ellipse in the bottom portion of the figure. The UKF approach estimates are expected to be closer to the true values than the EKF approach.

Several UT’s are available. One of the popular approaches is the scaled unscented transformation. Consider an \( n \) dimensional random variable \( x \), having the mean \( \hat{x} \) and covariance \( P \) and suppose that it propagates through an arbitrary nonlinear function \( f() \). The unscented transform creates \( 2n + 1 \) sigma vectors \( X \) (a capital letter) and weighted points \( W \), given by:

\[
X_{(0)} = \hat{x}
\]
\[
X_{(i)} = \hat{x} + \left( \sqrt{(n + \lambda)}P \right) \times i, \quad i = 1, \ldots, n
\]
\[
X_{(i+n)} = \hat{x} - \left( \sqrt{(n + \lambda)}P \right) \times i, \quad i = 1, \ldots, n
\]

where \( \sqrt{(n + \lambda)}P \) is the \( i \) th row of the matrix square root. \( \sqrt{(n + \lambda)}P \) can be obtained from the lower-triangular matrix of the Cholesky factorization. \( \lambda = \alpha^2(n + \kappa) - n \) is a scaling parameter, \( \alpha \) determines the spread of the sigma points around \( \hat{x} \) and is usually set to a small positive (e.g., 1e-4 \( \leq \alpha \leq 1 \)), \( \kappa \) is a second scaling parameter (usually set to 0) and \( \beta \) is used to incorporate prior knowledge of the distribution of \( \bar{x} \) (when \( x \) is normally distributed, \( \beta = 2 \) is an optimal value). \( W^{(m)} \) is the weight for the mean associated with the \( i \) th point and \( W^{(c)} \) is the weight for the covariance associated with the \( i \) th point, as follows:

\[
W^{(m)}_0 = \frac{\lambda}{(n + \lambda)}
\]

\[
W^{(c)}_0 = W^{(m)}_0 + (1 - \alpha^2 + \beta)
\]

\[
W^{(m)}_i = W^{(c)}_i = \frac{1}{2(n + \lambda)}, \quad i = 1, \ldots, 2n
\]

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

\[
y_i = f(X^{(i)}), \quad i = 0, \ldots, 2n
\]

The mean and covariance of \( y_i \) are approximated by a weighted average mean and covariance of the transformed sigma points as follows:

\[
y_u = \sum_{i=0}^{2n} W^{(m)}_i y_i
\]

\[
\bar{P}_u = \sum_{i=0}^{2n} W^{(c)}_i (y_i - \bar{y_u})(y_i - \bar{y_u})^T
\]

Figure 2. Illustration of properties of UKF and EKF [27]

The first step in the UKF is to sample the prior state distribution, i.e., generate the sigma points through the UT, which is a method for calculating the statistics of a random variable, which undergoes a nonlinear transformation. The basic premise is that to approximate a probability distribution is easier than to approximate an arbitrary nonlinear transformation. The samples are propagated through true nonlinear equations and linearization of the model is not required.

Suppose the mean \( \bar{x} \) and covariance \( P \) of vector \( x \) are known, a set of deterministic vectors called sigma points can then be found. The ensemble mean and covariance of the sigma points are equal to \( \bar{x} \) and \( P \). The nonlinear function \( y = f(x) \) 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 \( y \), which is the key to the unscented transformation. The UKF approach estimates are expected to be closer to the true values than the EKF approach. As compared to the EKF’s linear approximation, the unscented transformation is accurate to the second order for any nonlinear function.

B The unscented Kalman filter

The implementation algorithm of UKF is summarized as follows:

1. The transformed set is given by instantiating each point through the process model
2. Predicted mean
\[ \hat{x}_{k|k-1} = \frac{1}{\sum_{i=0}^{2n} W_i^{(c)}} \sum_{i=0}^{2n} W_i^{(c)} \zeta_i(k_{k-1}) \] (28)

3. Predicted covariance
\[ P_{k|k-1} = \sum_{i=0}^{2n} W_i^{(c)} [\zeta_i(k_{k-1}) - \hat{x}_{k|k-1}] [\zeta_i(k_{k-1}) - \hat{x}_{k|k-1}]^T + Q_{k-1} \] (29)

4. Instantiate each of the prediction points through the observation model
\[ Z_{i,k|k-1} = h(\zeta_i(k_{k-1})) \]

5. Predicted observation
\[ \hat{z}_{k|k-1} = \sum_{i=0}^{2n} W_i^{(c)} Z_{i,k|k-1} \]

6. Innovation covariance
\[ P_{zz} = \sum_{i=0}^{2n} W_i^{(c)} [Z_{i,k|k-1} - \hat{z}_{k|k-1}] [Z_{i,k|k-1} - \hat{z}_{k|k-1}]^T + R_k \] (30)

7. Cross covariance
\[ P_{xz} = \sum_{i=0}^{2n} W_i^{(c)} [\zeta_i(k_{k-1}) - \hat{x}_{k|k-1}] [Z_{i,k|k-1} - \hat{z}_{k|k-1}]^T \] (31)

8. Performing update
\[ K_k = P_{xz} P_{zz}^{-1} \]
\[ \hat{x}_k = \hat{x}_{k|k-1} + K_k (z_k - \hat{z}_{k|k-1}) \]
\[ P_k = P_{k|k-1} - K_k P_{zz} K_k^T \] (32)

3.2 The Interacting Multiple Model Unscented Kalman Filter

The IMM approach takes into account a set of models to represent the system behaviour patterns or system model. The estimator carries out soft switching among various models by the model probability. The overall estimate is obtained by a combination of the estimates from the filters running in parallel, based on the individual models that match the system modes. The measurements can be obtained from one or more sensors and the model-matched filters can be linear or nonlinear. The algorithm of IMM-nonlinear filters is introduced to deal with the noise uncertainty and system nonlinearity simultaneously.

Let a general system for multiple models in discrete time be described by
\[ x_{k+1} = f(x_k, k, M_k) + w_k(x_k, M_k) \] (33a)
\[ z_k = h(x_k, k, M_k) + v_k(x_k, M_k) \] (33b)

where \( f() \) and \( h() \) are the parameterized state transition and measurement functions, \( x_k \) and \( z_k \) are the dynamical state and measurement of the system in model \( M_k \) and \( w_k, v_k \) are the process noise and measurement noise with covariances \( Q_k \) and \( R_k \), respectively.

The IMMUKF algorithm uses model (Markov chain state) probabilities to weight the input and output of a bank of parallel UKFs at each time instant. The approach takes into account a set of models to represent the system behaviour patterns or system model. The overall estimate is obtained by a combination of the estimates from the filters running in parallel, based on the individual models that match the system modes. An IMM cycle consists of four major stages: interaction (mixing), filtering, model probability calculation and estimate combination. Figure 3 shows a block diagram of the IMMUKF algorithm with two models. One cycle of IMMUKF can be written as follows.

1. Model interaction/mixing

For given states \( x_{k|k-1}^i = x_{k|k-1}^{i,k} \) with corresponding covariances \( P_{k|k-1}^i = P_{k|k-1}^{i,k} \) and mixing probabilities \( \mu_{k|k-1}^{i,j} \) for every model, the initial condition for the model \( j \) is:
\[ x_{k|k-1}^j = \sum_{i=1}^{2n} \mu_{k|k-1}^{i,j} x_{k|k-1}^i \] \( j = 1, 2, ..., \tau \) (34)

The covariance corresponding to the above is:
\[ P_{k|k-1}^j = \sum_{i=1}^{2n} \mu_{k|k-1}^{i,j} P_{k|k-1}^i + \mu_{k|k-1}^{j,j} \] (35)

The model transition probabilities, which are related to Markov chain, are defined as:
\[ \pi_{ij} = P(M_k^i | M_k^j) = \begin{bmatrix} \pi_{11} & \pi_{12} & \cdots & \pi_{1j} \\ \pi_{21} & \pi_{22} & \cdots & \pi_{2j} \\ \vdots & \vdots & \ddots & \vdots \\ \pi_{i1} & \pi_{i2} & \cdots & \pi_{ij} \end{bmatrix} \] (36)

where \( i, j = 1, 2, ..., \tau \) and \( \tau \) is the number of sub-models. The mixing probabilities with mode switching probability matrix \( \pi_{ij} \) and the Gaussian mixing probabilities are computed via the equation:
\[ \mu_{k|k-1}^{i,j} = \frac{1}{\tau} \pi_{ij} k_{k-1} \] (37)

where \( \tau \) is a normalization factor:
\[ \tau = \sum_{i=1}^{2n} \pi_{ij} k_{k-1} \] (38)
2. Model individual filtering

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

$$X_{0,k-1} = x_{k-1}^0$$

$$X_{i,n,k-1}^l = x_{k-1}^0 + \sqrt{(n+\lambda)}P_{k-1}^{0i,l} \quad , \quad i = 1,...,n$$

$$X_{i,n,k-1}^j = x_{k-1}^0 - \sqrt{(n+\lambda)}P_{k-1}^{0i,l}^T \quad , \quad i = 1,...,n$$

- **Step 2 in UKF loop.** Time update (prediction steps):

$$\mathbf{P}_{i,k-1} = \sum_{i=0}^{2n} W_i^m \mathbf{C}_{i}^k$$

$$\mathbf{x}_{i,k-1} = \sum_{i=0}^{2n} W_i^m \mathbf{x}_{i,k-1}$$

- **Step 3 in UKF loop.** Measurement update (correction steps):

$$\mathbf{P}_{i,z}^z = \sum_{i=0}^{2n} W_i^m [\mathbf{Z}_{i,k-1} - \mathbf{z}_{k-1}][\mathbf{Z}_{i,k-1} - \mathbf{z}_{k-1}]^T + \mathbf{R}_k$$

$$\mathbf{P}_{i,z}^z = \sum_{i=0}^{2n} W_i^m [\mathbf{C}_{i}^k - \mathbf{x}_{i,k-1}][\mathbf{C}_{i}^k - \mathbf{x}_{i,k-1}]^T + \mathbf{Q}_k$$

$$\mathbf{K}_k = \mathbf{P}_{i,z}^z (\mathbf{P}_{i,z}^z)^{-1}$$

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

$$P_{k|k} = P_{k|k-1} - \mathbf{K}_k P_{i,z}^z \mathbf{K}_k^T$$

The samples are propagated through true nonlinear equations; the linearization is unnecessary (calculation of the Jacobian matrix is not required).

3. Model probabilities update

The model probability $\mu_k$ is updated according to the model likelihood and model transition probability governed by the finite-state Markov chain:

$$\mu_k = \frac{1}{c} \Lambda_k$$

where the normalized constant:

$$c = \sum_{i=1}^{r} \Lambda_k$$

and $\Lambda_k$ is a likelihood function given by:

$$\Lambda_k = \frac{1}{\sqrt{2\pi|\mathbf{P}_{i,z}^z|}} \exp \left[ -\frac{1}{2} \frac{[\mathbf{z}_k - \hat{x}_{k|k}]}{(\mathbf{P}_{i,z}^z)^{-1}} \right]$$

4. Combination of state estimation and covariance combination

The model-individual estimates and covariances are combined to an overall state and covariance.

$$\hat{x}_{k|k} = \sum_{i=1}^{r} \hat{x}_{i,k} \Lambda_k$$

$$P_{k|k} = \sum_{i=1}^{r} P_{i,k} \left[ \hat{x}_{i,k} - \hat{x}_{k|k} \right] \left[ \hat{x}_{i,k} - \hat{x}_{k|k} \right]^T$$

**Figure 3.** Block diagram of the IMMUKF algorithm (one cycle with two models)
4. Results and discussion

Simulation examples were carried out to evaluate the performance of the proposed approach in comparison with the conventional approaches. The computer codes were developed by the authors using Matlab® software. The commercial software Satellite Navigation (SATNAV) toolbox [28] by GPSSoft LLC was employed for generating the satellite positions and pseudoranges. It is assumed that the differential GPS mode is used and only the multipath and receiver thermal noise are included.

Assume that the differential GPS mode is used and most of the errors can be corrected, but the multipath and receiver thermal noise cannot be eliminated. The measurement noise variances \( \nu_i \) are set at 15m². Let each of the white-noise spectral amplitudes that drive the random walk position states be \( S_p = 0.003(\text{m} / \text{sec}^2) / \text{rad} / \text{sec} \). Furthermore, let the clock model spectral amplitudes be \( S_i = 0.4(10^{-18}) \text{sec} \) and \( S_g = 1.58(10^{-18}) \text{sec}^{-1} \). The data for INS error specifications is taken from Crista IMU specifications [29], summarized in Table 1. For the two models of the IMM algorithms, two process noise covariance matrices are used. Model 1 represents the system with smaller process noise (\( Q_{\text{small}} \)), while the system with larger process noise (\( Q_{\text{large}} \)) is represented by Model 2. In this paper, the ratio of process noise is 100, i.e., \( Q_{\text{large}} = 100 Q_{\text{small}} \). This ratio depends on the designer’s knowledge of the dynamic characteristics. For a very high dynamic manoeuvring situation, a larger ratio should be used. The model transition probability matrix is given by:

\[
\pi = \begin{bmatrix}
0.99 & 0.01 \\
0.01 & 0.99
\end{bmatrix}
\]  

(55)

The UKF parameters are set as \( \alpha = 1, \ \kappa = 0, \ \beta = 2 \).

<table>
<thead>
<tr>
<th></th>
<th>Gyros</th>
<th>Accelerometers</th>
</tr>
</thead>
<tbody>
<tr>
<td>Range</td>
<td>( \pm 300 )°/sec</td>
<td>( \pm 10 ) G</td>
</tr>
<tr>
<td>Scale Factor Error</td>
<td>(&lt; 3 )°/sec (( @ 25^\circ \text{C} ))</td>
<td>(&lt; 100 ) mG (0.98 m/s²) (( @ 25^\circ \text{C} ))</td>
</tr>
<tr>
<td>In-Run Bias Error:</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Fixed temperature</td>
<td>(&lt; 0.2 )°/sec (warmed up)</td>
<td>(&lt; 25 ) mG (0.245 m/s²)</td>
</tr>
<tr>
<td>Over temperature</td>
<td>(&lt; 0.6 )°/sec</td>
<td>(&lt; 51 ) mG (0.500 m/s²)</td>
</tr>
<tr>
<td>Noise (1o, no over-samples)</td>
<td>(&lt; \pm 0.7 )°/sec</td>
<td>(&lt; \pm 12 ) mG (0.120 m/s²)</td>
</tr>
</tbody>
</table>

Table 1. INS error specifications (from Crista IMU Specifications) [29]

4.1 Scenario 1 (Example 1)

Figure 4 shows the schematic illustration of a test trajectory for Scenario 1. The trajectory can be divided into ten intervals according to their dynamic characteristics. Table 2 presents a description of the vehicle motion to provide a better insight into vehicle dynamic information in each time interval. The trajectory can be divided into ten time intervals (or segments) according to the dynamic characteristics. The vehicle was simulated to conduct constant acceleration and level flight from 0 to 25 seconds, clockwise circular motion with a radius of 750m from 41 to 231 seconds and counter-clockwise turns from 283 to 374 seconds, where highly dynamic manoeuvring is involved. For all the other segments, constant-velocity straight-line flight is carried out.

Figure 4. The three-dimensional test trajectory for Scenario 1
<table>
<thead>
<tr>
<th>Segment number</th>
<th>Time interval (sec)</th>
<th>Motion</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>[0-25]</td>
<td>Constant acceleration level flight</td>
</tr>
<tr>
<td>2</td>
<td>[26-40]</td>
<td>Climbing</td>
</tr>
<tr>
<td>3</td>
<td>[41-231]</td>
<td>Counter-clockwise circular motion</td>
</tr>
<tr>
<td>4</td>
<td>[232-266]</td>
<td>Climbing</td>
</tr>
<tr>
<td>5</td>
<td>[267-282]</td>
<td>Constant velocity level flight</td>
</tr>
<tr>
<td>6</td>
<td>[283-374]</td>
<td>Constant acceleration clockwise turn</td>
</tr>
<tr>
<td>7</td>
<td>[375-389]</td>
<td>Descending</td>
</tr>
<tr>
<td>8</td>
<td>[390-424]</td>
<td>Constant velocity level flight</td>
</tr>
<tr>
<td>9</td>
<td>[425-449]</td>
<td>Descending</td>
</tr>
<tr>
<td>10</td>
<td>[450-484]</td>
<td>Constant velocity level flight</td>
</tr>
</tbody>
</table>

Table 2. Description of the vehicle motion for Scenario 1

Figures 5-7 provide the position accuracy for the results based on various approaches. Figure 5 shows the accuracy comparison for EKF and UKF in the north, east and altitude components, respectively. Figure 6 shows the position errors in the north, east and altitude components, respectively for IMMEKF as compared to EKF and for IMMUKF as compared to UKF. The use of UKF demonstrates performance improvement when compared to the EKF, due to better nonlinearity treatment. Comparisons of RMSEs are illustrated in Figure 7 for the designs without and with the IMM algorithm. It can be seen that the incorporation of the IMM mechanism can remarkably improve the filter estimation accuracy. Among the four approaches, the IMMUKF algorithm demonstrates superior navigation estimation accuracy as compared to the other three approaches.

The execution times for the four approaches are given in Table 3. The execution times for UKF and IMMEKF are about twice that of EKF; the execution time required for IMMUKF is also approximately twice those of UKF and IMMEKF and is about four times as much as EKF. The information is useful for making trade-offs between estimation accuracy and computational time when selecting a suitable algorithm for a specific application. The model probabilities calculated using the IMMEKF and IMMUKF are provided for inspecting the filter switching capability, as shown in Figure 8. For the intervals where highly dynamic manoeuvres are
involved, the model probability of Model 2 becomes larger than that of Model 1. In contrast, in the lower dynamic intervals, the model probability of Model 1 is larger than that of Model 2, confirming that Model 2 is more appropriate for highly dynamic regions. Moreover, the model probability for the IMMUKF demonstrates a stronger capability to accurately capture the dynamic behaviours than the IMMEKF, resulting in better estimation accuracy.

Figure 6. Position errors for IMMEKF as compared to EKF (left) and for IMMUKF as compared to UKF (right) - Scenario 1

Figure 7. Comparison of RMSE for EKF and UKF based results (left) and for IMMEKF and IMMUKF based results (right) - Scenario 1
<table>
<thead>
<tr>
<th>Approach</th>
<th>Time</th>
</tr>
</thead>
<tbody>
<tr>
<td>EKF</td>
<td>8.784</td>
</tr>
<tr>
<td>UKF</td>
<td>16.152</td>
</tr>
<tr>
<td>IMMEKF</td>
<td>17.821</td>
</tr>
<tr>
<td>IMMUKF</td>
<td>33.056</td>
</tr>
</tbody>
</table>

Table 3. Comparison of execution time for various approaches (in seconds) - Scenario 1

4.2 Scenario 2 (Example 2)

Another example is provided for further confirmation of the effectiveness and justification of the performance. The test trajectory is shown in Figure 9. Table 4 provides a description of the vehicle motion. The trajectory can be divided into nine segments, where the vehicle conducts constant acceleration and level flight from 0 to 25 seconds, counter-clockwise turns from 51 to 231 seconds and clockwise circular motion from 283 to 463 seconds. In the three segments, highly dynamic manoeuvring is involved. Constant-velocity straight-line flight is involved in all the other segments, where low dynamic motion is considered.

The results are shown in Figures 10-12. The performance improvement by the IMM algorithm is essentially similar to that obtained in the previous example. Figure 10 provides the accuracy comparison for EKF and UKF and Figure 11 shows the position errors for IMMEKF as compared to EKF and for IMMUKF as compared to UKF. Comparisons of RMSEs are illustrated in Figure 12. Among the four approaches, the IMMUKF algorithm demonstrates superior navigation estimation accuracy. The model probabilities calculated using the IMMEKF and IMMUKF are depicted in Figure 13. When a designer lacks sufficient information to develop a precise model or the parameters change with time, the IMM approach provides a feasible alternative approach that can be adopted for designing an adaptive filter in the ultra-tight GPS/INS integration design.
Figure 10. Comparison of position error for EKF and UKF - Scenario 2

Figure 11. Position errors for IMMEKF as compared to EKF (left) and for IMMUKF as compared to UKF (right) - Scenario 2
<table>
<thead>
<tr>
<th>Segment number</th>
<th>Time interval (sec)</th>
<th>Motion</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>[0-25]</td>
<td>Constant acceleration level</td>
</tr>
<tr>
<td></td>
<td></td>
<td>flight</td>
</tr>
<tr>
<td>2</td>
<td>[26-50]</td>
<td>Climbing</td>
</tr>
<tr>
<td>3</td>
<td>[51-231]</td>
<td>Counter-clockwise circular</td>
</tr>
<tr>
<td></td>
<td></td>
<td>motion</td>
</tr>
<tr>
<td>4</td>
<td>[232-282]</td>
<td>Climbing</td>
</tr>
<tr>
<td>5</td>
<td>[283-463]</td>
<td>Clockwise circular motion</td>
</tr>
<tr>
<td>6</td>
<td>[464-488]</td>
<td>Descending</td>
</tr>
<tr>
<td>7</td>
<td>[489-512]</td>
<td>Constant velocity level flight</td>
</tr>
<tr>
<td>8</td>
<td>[513-547]</td>
<td>Descending</td>
</tr>
<tr>
<td>9</td>
<td>[548-573]</td>
<td>Constant velocity level flight</td>
</tr>
</tbody>
</table>

Table 4. Description of the vehicle motion for Scenario 2

Figure 12. Comparison of RMSE for EKF and UKF based results (left) and for IMMEKF and IMMUKF based results (right) - Scenario 2

Figure 13. Comparison of model probability for IMMEKF (left) and IMMUKF (right) - Scenario 2

5. Conclusions

The IMM-based architecture has been presented to improve the estimation accuracy. The problem of filter parametric adaptation can be regarded as the generalization of structural adaptation (model switching). Featured in structural adaptation, the interacting multiple model (IMM) nonlinear filtering approach has demonstrated good potential as an alternative state estimation technique for the ultra-tightly coupled GPS/INS navigation design. The IMM algorithm is employed for dynamically adjusting process noise by defining different models according to the dynamic situation in which the vehicle is involved and accordingly enhancing the estimation accuracy. The approach is designed to solve the possible degradation problems caused by noise uncertainty and modelling error, so as to improve the navigation accuracy in highly dynamic regions without sacrificing precision in the lower dynamic regions.

An unscented Kalman filtering approach has been demonstrated to be superior to the EKF, due to the fact that the UKF is able to deal with the nonlinear formulation, while the linear model does not reflect the actual dynamic behaviour when the vehicle is
manoeuvring. The nonlinear filters have been incorporated into the IMM framework, resulting in the IMMEKF, IMMUKF algorithms. The IMM algorithm has been employed for dynamically adjusting the process noise. The use of an IMM method allows the exploitation of the benefits of highly dynamic models in the problem of vehicle navigation. Two examples have been provided to illustrate the effectiveness of the design and demonstrate effective improvement in navigation estimation accuracy. Performance comparisons for various approaches: EKF, UKF, IMMEKF and IMMUKF have been presented. Among the four approaches, IMMUKF leads to very promising results in navigational accuracy improvement for an ultra-tightly coupled GPS/INS integrated navigation system. Nevertheless, selection of the nonlinear filters in the IMM framework leads to different levels of computational burden. The information is also provided, which is useful for making trade-offs between estimation accuracy and computational time when selecting a suitable algorithm for a specific application.

6. Acknowledgements

Funding for this work was provided by the National Science Council of the Republic of China under grant numbers NSC 98-2221-E-019-021-MY3 and NSC 101-2221-E-019-027-MY3.

7. References


