Extended Kalman Filtering and Interacting Multiple Model for Tracking Maneuvering Targets in Sensor Networks.

Seham Mouawad Aly*, Raafat El Fouly**, Hoda Baraka***

Abstract: This paper considers the nonlinear state estimate problem for tracking maneuvering targets. Two methods are introduced to overcome the difficulty of non-linear model. The first method uses Interacting Multiple Model (IMM) which includes 2, 3, 4 and 10 models. These models are linear, each model stands for an operation point of the nonlinear model. Two model sets are designed using Equal-Distance Model-Set Design for each. The effect of increasing the number of models, separation between them and noise effect on the accuracy is introduced. The second method uses Second order Extended Kalman Filter (EKF2) which is a single nonlinear filter. Both methods are evaluated by simulation using two scenarios. A comparison between them is evaluated by computing their accuracy, change of operation range and computational complexity (computational time) at different measurement noise. Based on this study for small range of variation of nonlinear parameter, and low noise the EKF2 introduced quick and accurate tracking. For a large range of nonlinearity and good separation between models of IMM, at minimum noise large and small numbers of models of IMM introduced best accuracy but as the noise increase large number keeps higher accuracy until the large numbers and small numbers of IMM introduced bad accuracy. At high noise optimizing number of models and separation between model sets, IMM introduces better accuracy.

Keywords: Interacting Multiple Model (IMM), Extended Kalman filter, Probabilistic Data Association, Sensor Networks.

1. Introduction
Filtering and estimation are two of the most pervasive tools of engineering. Whenever the state of a system must be estimated from noisy sensor information, some kind of state estimator is employed to fuse the data from different sensors to produce an accurate estimate of the true system state. When the system dynamics and observation models are linear, the minimum mean squared error (MMSE) estimate may be computed using the Kalman filter. However, in most applications of interest the system dynamics and observation equations are nonlinear and suitable extensions to the Kalman filter. Probably the most widely used estimator for nonlinear systems is the extended Kalman filter (EKF). The EKF is used in modeling different systems as control system in fault tolerance [1],[2] and [3] for radar system. The EKF applies the Kalman filter to nonlinear systems by simply linearising all the nonlinear models so that the traditional linear Kalman filter equations can be applied.

* Faculty of Engineering, Maadi Modern Academy, sehamaly@hotmail.com
** Faculty of Engineering, Cairo University, Ralfouly@mtci.eg
*** Faculty of Engineering, Cairo University
However, in practice, the use of the EKF has two well-known drawbacks [4],[5]. First, linearization can produce highly unstable filters if the assumptions of local linearity are violated. Second, the derivation of the Jacobian matrices is nontrivial in most applications and often lead to significant implementation difficulties.

There are different filtering methods that have been proposed as Monte Carlo method [6] and recently Unscented transformation (UT)[7],[8]. The unscented transformation is a new method for calculating the statistics of a random variable which undergoes a nonlinear transformation. It is founded on the intuition that it is easier to approximate a Gaussian distribution than it is to approximate an arbitrary nonlinear function or transformation. The approach is illustrated in [4]. A set of points (or sigma points) are chosen so that they lay around the sample mean and sample covariance. The nonlinear function is applied to each point in turn to yield a cloud of transformed points. Although this method bares a superficial resemblance to Monte Carlo-type methods, there is an extremely important and fundamental difference. The samples are not drawn at random but rather according to a specific, deterministic algorithm. The Unscented Kalman Filter (UKF) for n dimension state space it takes (2n+1) sigma points which introduce a large time in calculations. A comparison between EKF and UKF is introduced in [4] and [9] which concluded that the choice between them is application dependent since not all the cases EKF is better than UKF or vs. versa.

Multiple-Model (MM) algorithm in general runs set of filters that models system dynamics. Then, the algorithm fuses the output of those filters for an overall estimate. Many MM target tracking algorithms have been proposed and each of them fuses its estimates differently. In [10] a comparison between different types of IMM illustrates their concept of working and their performance at targets maneuvering for the three proposed scenarios. The IMM introduced good accuracy at minimum execution time. For a detailed treatment of the MM algorithms, concerning their theoretical background, underlying assumptions, structures, implementations, and applications, a vast amount of references, and more, the reader is referred to [11],[12],[13].

There are two major directions to improve the MM approach: Develop a better MM algorithm and design a better model set. Methods of developing a better MM are illustrated in [14],[15],[16],[17],[18]. In [14],[15] invested a variant of the IMM that included an adaptive set of models. Improving the performance of IMM by successive evaluation of likely hood as in [17],[18], or by smoothing as in [16]. Designing a better model set is illustrated in [19]. Three classes of general methods for optimal design of model sets first by minimizing distribution mismatch, second minimizing modal distance, third moment matching, respectively are proposed. It is shown theoretically that the use of too many models is performance-wise as bad as that of too few models [19]. In this paper effect of changing number of models is introduced. In our work, we try to overcome the nonlinearity of the maneuvering targets which always represented by EKF by IMM. Instead of using single nonlinear model (EKF) a number of linear models using IMM is introduced. Each model stands for an operating point of this nonlinear model. These operating points are chosen according to the first class in [19]. Choosing the range and the separation between models is a hard decision due to the construction and operation of IMM. This point is illustrated by introducing two model sets.

The system dynamics includes two models linear motion and coordinated turn rate at constant turning angle. The linear model is considered for quick maneuvering. The complexity and
accuracy of a fixed set of IMM with 2, 3, 4 and 10 models is introduced. A comparison between their execution time, accuracy is introduced at different noisy measures.

Sections 2 describes the design of IMM algorithm. Section 3 describe EKF2 algorithm. The simulation scenarios are in section 4. Results and their analysis are presented in Section 5. Section 6 provides a conclusion.

2. The design of IMM algorithm
The IMM estimator is a suboptimal hybrid filter that has been shown to be one of the most cost-effective hybrid state estimation schemes. The main feature of this algorithm is the ability to estimate the state of a dynamic system with several behavior modes which can “switch” from one to another. In particular, the IMM estimator can be a self-adjusting variable-bandwidth filter, which makes it natural for tracking maneuvering targets [14].

The state dynamics of target is modeled as:

\[ x(k+1) = F(k,m(k+1))x(k) + G(k,m(k+1))u(k,m(k+1)) \]  

\[ z(k) = H(k)x(k) + w(k), \ldots \]  (1) (2)

where \( x \in \mathbb{R}^n \) is the system state vector, \( z \in \mathbb{R}^n \) is the measurement vector, \( u \in \mathbb{R}^n \) and \( w \in \mathbb{R}^n \) are mutually uncorrelated, white zero mean Gaussian noises with covariances \( Q_u \) and \( R_w \) respectively. The parameter \( m_k \) presents the current system model. \( F \) is the system dynamic matrix, and \( H \) is the measuring matrix. Because the accurate system model is unknown, the system is described by a number of models. The event that the \( i^{th} \) model \( m_i \) is actual at time \( k \) is denoted as \( M_i(k) = \{ m(k) = m_i \} \). It is assumed that the system model sequence is a Markov chain with transition probabilities

\[ P[M_j(k+1) | m_i(k)] = P_{ij}(k) \quad \text{where} \quad \sum_{j=1}^{r} P_{ij}(k) = 1, \quad i=1,2,\ldots,r \]  (3)

Scheme of the IMM algorithm is shown in figure one.

A Markov transition matrix is used to specify the probability that the target is in one of the modes of operation. The model probabilities are updated at each new measurement, and the resulting weighting factors are used in calculating the state. One cycle of a practical IMM algorithm consists of the following steps [20]

**Step 1: Calculate the mixing probabilities**
The probability that mode \( M_i \) was effect at time \( k-1 \) given that \( M_j \) is in effect at time \( k \) conditioned on \( Z^{k-1} \) is

\[ \mu_{ij}(k-1 | k-1) = P[M_i(k-1) | M_j(k), Z^{k-1}] = \frac{1}{\bar{c}_j} P_{ij} \mu_j(k-1) \]  (4)

where \( \bar{c}_j = \sum_{i=1}^{r} P_{ij} \mu_i(k-1) \) is the predicted mode probabilities and \( r \) different modes.
**Step 2: Calculate the mixed initial condition**

Starting with previous state estimates \( \hat{x}^i(k-1|k-1) \) and \( P^i(k-1|k-1) \) corresponding covariance matrices obtained as output from the \( r \) different Kalman filters (acting as the \( r \) different modes).

Mixed initial condition for the filter \( M_j \) at time \( k \) is:

\[
\hat{x}^j(k-1|k-1) = \sum_{i=1}^{r} \hat{x}^i(k-1|k-1).\mu_{ji}(k-1|k-1) \quad (5)
\]

\[
P^j(k-1|k-1) = \sum_{i=1}^{r} \mu_{ji}(k-1|k-1).\{P^i(k-1|k-1) + \left[\hat{x}^i(k-1|k-1) - \hat{x}^j(k-1|k-1)\right]^T\} \quad (6)
\]

**Step 3: Perform mode-matched filtering and calculate the likelihood function corresponding to the \( r \) filter**

Use the estimate \( \hat{x}^j(k-1|k-1) \) and corresponding covariance \( P^j(k-1|k-1) \) as inputs to the filter matched to \( M_j(k) \), which uses \( z(k) \) to yield \( \hat{x}^j(k|k) \) and \( P^j(k|k) \).

Kalman filter equations

\[
\hat{x}_i(k|k) = \hat{x}_i(k|k-1) + K_f^i(k)v_i(k) \quad (7)
\]

\[
\hat{x}_i(k|k) = \hat{x}_i(k|k-1) + K_f^j(k)v_i(k) \quad (8)
\]

\[
v_i(k) = z(k) - H_i(k)\hat{x}_i(k|k-1) \quad (9)
\]

\[
P_i(k|k-1) = F_i^i(k-1|k-1)P_i(k-1|k-1)F_i^T_i(k-1) + GQ(k)G^T \quad (10)
\]

\[
S_j(k) = H_i(k)P_i(k|k-1)H_i^T(k) + R(k) \quad (11)
\]

\[
K_j^i(k) = P_i(k|k-1)H_i^T(k)S_j^{-1}(k) \quad (12)
\]

\[
P_i(k|k) = P_i(k|k-1) - K_j^i(k)S_j(k)K_j^T(k) \quad (13)
\]

\( v_i, S_i \) are the innovation process and its covariance matrix; \( K_{f,i} \) the filter gain.

The likelihood functions for filter \( j \) is as follows:

\[
\Lambda_j(k) = N[v_j(k);0,S_j(k)]
\]

\[
= 2\pi S_j(k)^{1/2} \exp\left[-\frac{1}{2}v_j^2(k)S_j^{-1}(k)v_j(k)\right] \quad (14)
\]

where \( v_j(k) = z(k) - \hat{z}(k|k-1) \) is the innovation for filter \( j \) and \( S_j(k) \) is the covariance matrix associated with \( v_j(k) \)

**Step 4: Update model probability**

\[
\mu_j(k) = \frac{1}{c}\Lambda_j(k)\sigma, \text{ where } c = \sum_{j=1}^{r} \Lambda_j(k)\sigma \quad (15)
\]

**Step 5: Combine model-conditioned estimates and covariance.**

For output purposes only, \( \tilde{x}(k|k) \) and \( P(k|k) \) are computed according to

\[
\tilde{x}(k|k) = \sum_{i=1}^{r} \tilde{x}_i^i(k|k)\mu_i(k) \quad (16)
\]

\[
P(k|k) = \sum_{i=1}^{r} \mu_i(k)\{P^i(k|k) + [\tilde{x}^i(k|k) - \tilde{x}(k|k)][\tilde{x}^i(k|k) - \tilde{x}(k|k)]^T\} \quad (17)
\]
The nonlinear model has system dynamics of the form

\[ x(k + 1) = f(k, x(k)) + Gu(k) \]  \hspace{1cm} (18)

\[ z(k + 1) = H(k, x(k)) + w(k) \]  \hspace{1cm} (19)

where the process noise, \( u[k-1] \), is zero-mean, white noise with covariance \( Q[k-1] \). The measurement equation in (19) has the measurement noise, \( w[k] \) zero-mean, white noise with covariance \( R[k] \). The EKF provides a method for linearly approximating the predicted state \( \hat{x}(k + 1 | k) \), and the associated state prediction covariance \( P(k+1|k) \) of the original nonlinear model. To obtain \( \hat{x}(k + 1 | k) \), the nonlinear function in Equation (18) is expanded in a vector Taylor series around the latest state estimate \( \hat{x}(k | k) \) using equation 20. Truncating after terms up to the first and second order yield the first (EKF1) and second order (EKF2) Extended Kalman Filter, respectively. The derivation here will be for EKF2 from which the EKF1 is a particular case.

\[ f(x) = f(\hat{x}) + \nabla_x f(\hat{x})(x - \hat{x}) + \frac{1}{2} \nabla^2_x f(\hat{x})(x - \hat{x})^2 + H.O.T \]  \hspace{1cm} (20)

Expanding Equation (18) about the latest state estimate, up to second order terms gives

\[ x[k] = f[k-1, \hat{x}(k-1 | k-1)] + f'_x[k-1][x(k-1) - \hat{x}(k-1)] + \]

\[ \frac{1}{2} \sum_{j=1}^{n} e_j(x(k-1) - \hat{x}(k-1))^T f^j_x[k-1][x(k-1) - \hat{x}(k-1)(k-1)] + H.O.T + G(k-1)u(k-1) \]  \hspace{1cm} (21)
where \( n \) is the dimension of \( x \), \( e_i \) is the \( i^{th} \) \( n \) dimension Cartesian basis vector.

\[
f_i(k-1) \Delta \nabla e_i[f(k-1, x)]^T \bigg|_{x=x(k-1)} \tag{22}
\]
is the Jacobian of the vector \( f[k-1; x] \), evaluated at the latest state estimate, and

\[
f_{x,x}^i(k-1) \Delta \nabla^2 e_i^j \bigg|_{x=x(k-1)} \tag{23}
\]
is the Hessian of the \( i^{th} \) component of \( f \), and \( H. \ O. \ T. \) represents the Higher Order Terms.

Neglecting the \( H.O.T. \) and taking the expectation of Equation (21) conditioned on the observations up to time \( k-1 \) gives

\[
\tilde{x}(k | k-1) = f(k-1, \tilde{x}(k-1) | k-1) + \frac{1}{2} \sum_{i=1}^n e_i \text{tr}[f_{x,x}^i(k-1)P(k-1 | k-1)] \tag{24}
\]
where \( P(k-1 | k-1) = E[(x(k-1) - \tilde{x}(k-1) | k-1)](x(k-1) - \tilde{x}(k-1) | k-1)^T \) is the latest state error covariance and we have used the fact \( E[x^T A x] = tr(AP) \).

To obtain the \( P(k|k-1) \) subtract equation (24) from (21) again ignoring the \( H.O.T. \). Then, by multiplying the difference by its transpose and taking the expectation conditioned on the observations up to time \( k-1 \), the state prediction error covariance is given by

\[
P(k | k-1) = f_i(k-1)P(k-1 | k-1)f_i^T(k-1) + \\
\frac{1}{2} \sum_{i=1}^n \sum_{j=1}^n e_i e_j^T \text{tr}[f_{x,x}^i(k-1)P(k-1 | k-1)f_{x,x}^j(k-1)P(k-1 | k-1)] + Q(k-1) \tag{25}
\]

The EKF has two parameters, \( Q \) and \( R \), which represent the process noise covariance and the measurement noise covariance. \( R \) is determined empirically and accounts for the uncertainty in the tracking data. Setting these matrices properly goes a long way toward making the filters robust. We determine \( Q \) using the continuous process noise matrix \( Q^* \) which assumes that the process noise always enters the process model on the highest derivative [4].

\[
Q = \Phi_1 f_i(k-1)Q f_i(k-1) \tag{26}
\]

Therefore, \( \Phi_1 \) is a scaling parameter which acts as a confidence value for how sure we are that the process model is an accurate description of the true motion dynamics. The EKF also needs to be initialized on startup. However, in some practical situations linearization introduces significant biases or errors. In [5] introduce these defects but with good tuning and knowing the limitation of noise you can get good accuracy with less computation complexity than that introduced by Unsecented Kalman filter as in [4].

4. The simulation scenarios

Model set of IMM

Two kinematics models are introduced for quick maneuver detection. Constant Velocity Model (CV). This model is the most commonly used. The target is assumed to move with nearly constant velocity. Coordinated Turn Rate Model (CT) where set of accelerations modeled are those in the direction normal to the velocity, which model constant turns \( \omega \). The nonlinear model is the case of considering a fifth state \( \omega \) instead of constant value.
For notational simplicity, \( x_t \equiv \{x_t,x_{\dot{y}},y_{\dot{y}}\} \) refers to the state (coordinates and the velocities). All models are in the generic state-space form

\[
x_{t+1} = F \ x_t + G \ u_t
\]

\[
F_{CV} = \text{diag} \begin{bmatrix} 1 & \Delta T \ 0 & 1 \end{bmatrix}, \quad G = \text{diag} \begin{bmatrix} \Delta T^2 \ 2 \Delta T \end{bmatrix}, \quad F_{CT} = \begin{bmatrix} 1 & \frac{\sin \omega \Delta T}{\omega} & 0 & -1 - \frac{\cos \omega \Delta T}{\omega} \\ \omega & \cos \omega \Delta T & 0 & -\sin \omega \Delta T \\ 0 & 1 - \frac{\cos \omega \Delta T}{\omega} & 1 & \frac{\sin \omega \Delta T}{\omega} \\ 0 & \frac{\sin \omega \Delta T}{\omega} & 0 & \frac{\sin \omega \Delta T}{\omega} \end{bmatrix}
\]

where \( u_t \sim N(0, \text{diag}(\sigma_x^2, \sigma_y^2)) \).

**Model set design for IMM**

Equal-Distance Model-Set Design proposed in [19] is considered. The number of models required to resemble the nonlinear model depends on two parameters. First, consider the estimate range of operation of the nonlinear state changes from \( L_{\text{min}} \) to \( L_{\text{max}} \). Second the separation between models is \( \Delta \). In simulation two ranges are considers. The first range \( \omega \) changes from -0.05 to 0.05 with \( \Delta = 0.02 \) while the second from -5 to 5 with \( \Delta = 2 \).

**First model set design:**

For a range of operation \( L (-0.05 \text{ to } 0.05) \), change the number of models of IMM, first IMM2 includes two models CV and CT with the turning angle \( \omega = 0.02 \) radian/sec. The transition matrix

\[
P_t = \begin{bmatrix} 0.9 & 0.1 \\ 0.1 & 0.9 \end{bmatrix} \text{ and } \mu = \begin{bmatrix} 0.5 \\ 0.5 \end{bmatrix}, \quad Q=\text{diagonal}(0.5^2) \quad R=\text{diagonal}(50).
\]

The Second IMM3 includes three models one CV and two CT turning angle \( \omega = 0.02, -0.02 \) radian/sec respectively. The transition matrix

\[
P_t = \begin{bmatrix} 0.98 & 0.01 & 0.01 \\ 0.01 & 0.98 & 0.01 \\ 0.01 & 0.01 & 0.98 \end{bmatrix} \text{ and } \mu = \begin{bmatrix} 0.4 \\ 0.3 \\ 0.3 \end{bmatrix}, \quad Q=\text{diagonal} (0.5^2) \quad R=\text{diagonal} (70).
\]

The third IMM is IMM4. The IMM4 includes 4 models one CV and three CT with \( \omega \) [0.05,-0.05,0.02].

\[
P_t = \begin{bmatrix} 0.97 & 0.01 & 0.01 & 0.01 \\ 0.01 & 0.97 & 0.01 & 0.01 \\ 0.01 & 0.01 & 0.97 & 0.01 \\ 0.01 & 0.01 & 0.01 & 0.97 \end{bmatrix} \text{ and } \mu = \begin{bmatrix} 0.25 \\ 0.25 \\ 0.25 \\ 0.25 \end{bmatrix}
\]

The \( Q \) is the same and \( R=\text{diagonal} (90) \).

The fourth IMM is IMM10. The IMM10 includes 10 models one CV and nine CT with the turning angle \( \omega \) [0.01, -0.01, 0.02, -0.02, 0.03, -0.03, 0.04, -0.05, 0.05], and same \( Q \) and \( R=\text{diagonal} (1000) \).
The transition matrix

\[
P_r = \begin{bmatrix}
0.82 & 0.02 & 0.02 & 0.02 & 0.02 & 0.02 & 0.02 & 0.02 \\
0.02 & 0.82 & 0.02 & 0.02 & 0.02 & 0.02 & 0.02 & 0.02 \\
0.02 & 0.02 & 0.82 & 0.02 & 0.02 & 0.02 & 0.02 & 0.02 \\
0.02 & 0.02 & 0.02 & 0.82 & 0.02 & 0.02 & 0.02 & 0.02 \\
0.02 & 0.02 & 0.02 & 0.02 & 0.82 & 0.02 & 0.02 & 0.02 \\
0.02 & 0.02 & 0.02 & 0.02 & 0.02 & 0.82 & 0.02 & 0.02 \\
0.02 & 0.02 & 0.02 & 0.02 & 0.02 & 0.02 & 0.82 & 0.02 \\
0.02 & 0.02 & 0.02 & 0.02 & 0.02 & 0.02 & 0.02 & 0.82
\end{bmatrix}
\]

The minimum likelihood functions \( \xi = 0.02 \).

**Second model set design:**

For a range of operation \( L (5 \rightarrow 5) \), change the number of models of IMM, first IMM2 includes two models CV and CT with the turning angle \( \omega = 2 \) radian/sec. The other parameters remains as is model set 1. Second IMM3 includes three models one CV and two CT turning angle \( \omega = 2, -2 \) radian/sec respectively. The third IMM is IMM4. The IMM4 includes 4 models one CV and three CT with \( \omega [5, -5, 2] \) The fourth IMM is IMM10. The IMM10 includes 10 models one CV and nine CT with the turning angle \( \omega [1, -1, 2, -2, 3, -3, 4, -4, 5] \). The other parameters remains as is model set 1. The minimum likelihood functions for both sets \( \xi = 0.02 \).

Two maneuver scenarios were investigated in the simulation as introduced in figure 2 and table 1. We tested our model using Matlab 7.0.4 on Pentium (R)4 CPU 2.66Ghz and 256 MB RAM., under windows Xp environment. The results are the average of 200 run. The time interval is 1 sec. All values is in units of sensor beam length. To evaluate the performance of the proposed method; we calculate Root Mean Square error (RMS) for the positions \( x \) and \( y \) also for the velocity \( V_x \), and \( V_y \):

\[
RMS\_error(k) = \sqrt{\frac{1}{M} \sum_{j=1}^{M} (x(k) - x^j(k))(x(k) - x^j(k))^T}
\]

(28)

![Figure 2: Ground truth Trajectories](image-url)
<table>
<thead>
<tr>
<th>Scenario1</th>
<th>Scenario 2</th>
<th>Scenario of model set 2</th>
</tr>
</thead>
<tbody>
<tr>
<td>K 0-59</td>
<td>Ω 0</td>
<td>ω 0.03</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.01</td>
<td></td>
<td></td>
</tr>
<tr>
<td>90-119</td>
<td>-0.05</td>
<td>-0.01</td>
</tr>
<tr>
<td>120-129</td>
<td>-0.03</td>
<td>0.04</td>
</tr>
<tr>
<td>130-150</td>
<td>0.01</td>
<td>0.02</td>
</tr>
</tbody>
</table>

### Setting of EKF2

The nonlinear model has system dynamics [equation (18)]:

\[
f(k, x(k)) = \begin{bmatrix}
    x_1 + \sin(x_3 \Delta T)x_2 - \cos(x_3 \Delta T)x_4 \\
    \cos(x_3 \Delta T)x_2 - \sin(x_3 \Delta T)x_4 \\
    1 - \cos(x_3 \Delta T) \\
    \sin(x_3 \Delta T)x_2 + \cos(x_3 \Delta T)x_4 \\
\end{bmatrix} \\
\begin{bmatrix}
    x_1 + \sin(x_3 \Delta T)x_2 - \cos(x_3 \Delta T)x_4 \\
    \cos(x_3 \Delta T)x_2 - \sin(x_3 \Delta T)x_4 \\
    1 - \cos(x_3 \Delta T) \\
    \sin(x_3 \Delta T)x_2 + \cos(x_3 \Delta T)x_4 \\
\end{bmatrix}
\]

**G(k) =** \[
\begin{bmatrix}
    \frac{1}{2} \Delta T^2 & 0 & 0 \\
    \Delta T & 0 & 0 \\
    0 & \frac{1}{2} \Delta T^2 & 0 \\
    0 & \Delta T & 0 \\
    0 & 0 & 1
\end{bmatrix}
\]

Measuring equation

\[
z(k) = \begin{bmatrix}
    1 & 0 & 0 & 0 & 0 \\
    0 & 0 & 1 & 0 & 0
\end{bmatrix} x(k-1) + w(k-1),
\]

The EKF2 is initialized by \(Q_m\) diagonal 0.01² and \(R\) diagonal (20), \(w\) white noise with zero mean and \(\sigma_{x_1} = \sigma_{x_2} = \sigma_{x_4} = 0.5\). \(Q_m=10^{-7}\). These values are changed as the time step and duration is changed. The EKF2 is very sensitive to any change of noise and need tuning as the motion of the target change. Adding a constant value over the state 5 reduces the effect of the noise change since it’s very sensitive to noise. The change of \(\Delta T\) can drive the filter to instability (due to nonlinearity). Calculating the \(R\), and \(Q\) also need to change to be sure that the filter still stable.

### 5. The results and analysis

In Table 2 we compare between the minimum number of models of IMM (IMM2), high number of models of IMM (IMM10) and EKF2. Taking minimum noise (\(\sigma_{x_1}=0.5, \sigma_{x_2}=0.5\)) and small interval of \(\omega\) with small model distance.

The EKF2 introduced minimum errors. While maximum number of model (IMM10) introduces higher errors. The second model set is compared between IMM2 and IMM10. EKM2 need new adjustment to follow the model sets and will be unstable for varying noise. As the separation between models increase the errors introduced is minimum. IMM10 introduce minimum errors. So as the varying range of \(\omega\) is small and also information noise is small the EKM2 is optimal and stable. As the range of variation of \(\omega\) increase it becomes sensitive to variation of noise and unstable. In case of changing time step, at low \(\Delta T\) for another type of motion as in virtual reality [4] we get better results. Validate our model of EKF by comparing its results with [4] as \(\Delta T\) values but we change \(Q_m\) to \(10^{-4}\), \(R\) to (diagonal 25) and \(Q\) to (diagonal 0.5²) the RMSE was of order \(10^{-2}\) for positions and less than 0.5 for velocities. Using EKF needs a good routine to tune its parameters as the noise change.

9/12
In figures 3 and 4 the range of \( \omega \) is small. At minimum noise there is no benefit from increasing the number of models but it increases execution time. At higher noise IMM3 for first scenario and IMM2 for second scenario introduce minimum errors. This means small number of models has better performance than higher number of models at higher noise. In figure 5 the separation between models is increased. At minimum noise as the number of models increases the errors are reduced while as the noise increases the change of number of models doesn’t improve the accuracy.

From these results there are two cases for noise and two cases for models set design.

First consider the cases of noise. In case of low noise (the information noise is negligible): The small value of \( \Delta \) increase both hit probability to the right estimate model and computation complexity as number of model increase. On the other hand there will be some models introduce low innovation which increase their likelihood probability. If these models are \( \varepsilon \) then the error of the estimated states will increase by a percentage of

\[
\sum_{i=1}^{\xi} \hat{x}^i(k | k) \mu_i(k).
\] (30)

In case of high noise, small number of models and large number introduce bad accuracy. But a medium number introduces acceptable accuracy.

Second consider the model set design

In case of separation between models doesn’t allow low innovation to the other incorrect models. The accurate estimate of states values increase but for the other incorrect models, its mode probability will reach zero that cause slow transition between models. To overcome this by putting a limit to the minimum value of likelihood functions

\[ 0 < \Lambda_j(k) < \xi, \quad j = 0, \ldots, r \] (31)

In other case if separation between models is high but noise is high this case if the right model introduce minimum innovation the accuracy increase if not there will be bad accuracy more than the low number of models. The time step changing its value affects the accuracy of the estimated state. As \( \xi \) increase for the same running time the number of calculation decrease and the system error accumulation from incorrect models increase so accuracy decrease. That for all used IMM and also EKF but IMM is more robust than EKF and can remain without adjustment if we increase the covariance \( R \).

6. The conclusion

Tracking maneuvering targets in sensor network requires quick tracking at minimum power consumption with high accuracy. Modeling the targets motion with single nonlinear model with 2EKF or using multiple linear models using IMM are two methods which are considered in this paper. The nonlinear state estimate problem was solved by EKF and recently by UKF that overcome the linearization assumption of EKF. In this paper, we consider the nonlinear model of coordinated turn rate if \( \omega \) changing with time. Two ranges of operation are considered. According to these ranges we have two models sets for IMM. Equal-Distance Model-Set Design is considered to evaluate these sets. It is clear that for small range of variation of nonlinear parameter, and low noise the EKF2 introduced quick and accurate tracking. As the range of nonlinearity increase and separation between models of IMM increase at low noise high numbers of models of IMM introduced best accuracy. As the noise increase the large numbers introduced good accuracy until it becomes as bad as small number of IMM. At high noise optimized number of models and separation between model sets, IMM introduces better accuracy as IMM3 in scenario 1 and IMM2 in scenario 2.
Table 2

<table>
<thead>
<tr>
<th>IMM 2 Scenario1 Model set1</th>
<th>IMM 2 Scenario2 Model set1</th>
<th>IMM 2 Model set2</th>
<th>IMM 10 Scenario1 Model set1</th>
<th>IMM 10 Scenario2 Model set1</th>
<th>EKF2 Scenario1</th>
<th>EKF2 Scenario2</th>
</tr>
</thead>
<tbody>
<tr>
<td>RMSE&lt;sub&gt;x&lt;/sub&gt;</td>
<td>1.3182</td>
<td>1.4871</td>
<td>3.5665</td>
<td>0.2265</td>
<td>0.2417</td>
<td>0.789</td>
</tr>
<tr>
<td>RMSE&lt;sub&gt;y&lt;/sub&gt;</td>
<td>1.2987</td>
<td>0.6039</td>
<td>0.5504</td>
<td>0.2116</td>
<td>0.1914</td>
<td>0.1586</td>
</tr>
<tr>
<td>RMSE&lt;sub&gt;vx&lt;/sub&gt;</td>
<td>0.0857</td>
<td>0.2728</td>
<td>0.1235</td>
<td>0.0741</td>
<td>0.2314</td>
<td>0.0519</td>
</tr>
<tr>
<td>RMSE&lt;sub&gt;vy&lt;/sub&gt;</td>
<td>0.0683</td>
<td>0.28</td>
<td>0.2203</td>
<td>0.0973</td>
<td>0.2419</td>
<td>0.1168</td>
</tr>
<tr>
<td>Execution Time</td>
<td>0.00071496</td>
<td>0.0089</td>
<td>0.00092153</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

\( \sigma \) 0.5 0.5 0.5

Figure 3 the RMSE for the four state of models for IMM2, IMM3, IMM4 and IMM10 for both \( \sigma_x, \sigma_y \) (sigma axis) changing from 0.5 to 10 at time step 1 sec. For Scenario1

Figure 4 the RMSE for the four state of models for IMM2, IMM3, IMM4 and IMM10 for both \( \sigma_x, \sigma_y \) (sigma axis) changing from 0.5 to 10 at time step 1 sec. For Scenario 2

Figure 5 the RMSE for the four state of models for IMM2, IMM3, IMM4 and IMM10 for both \( \sigma_x, \sigma_y \) (sigma axis) changing from 0.5 to 10 at time step 1 sec. For second model design
7. References

[1] Ludmila Mihaylova 1 Emil Semerdjiev1 X. Rong Li, "Detection and Localization of Faults in System Dynamics by IMM Estimator" 1999


[19] X. Rong Li, Zhanlue Zhao, and Xiao-Bai Li, ’’General Model-Set Design Methods for Multiple-Model Approach”, IEEE TRANSACTIONS ON AUTOMATIC CONTROL, VOL. 50, NO. 9, SEPTEMBER 2005

[20] Hongshe Dang, Chongzhao Han Xi’an, Shaan and Dominique GRUYER “Combining of IMM filtering and DS data association for multi-target tracking” 2004 china