Proceedings of the 17th World Congress
The International Federation of Automatic Control
Seoul, Korea, July 6-11, 2008

Input-Adaptive Models Based Multiple-Model Algorithm for Maneuvering Target Tracking

Jian Lan. Chundi Mu

Tsinghua National Laboratory for Information Science and Technology
Department of Automation, Tsinghua University, Beijing, P. R. China,
(e-mail: lan-j04@mails.tsinghua.edu.cn, muchd@tsinghua.edu.cn).

Abstract: The dynamic models with multilevel inputs are adopted in a kind of multiple model estimator for highly maneuvering target tracking. While the target maneuvers with the continuous time-varying accelerations, the estimator increases the levels to improve the percentage of coverage, which induces two problems: the increase of calculation burden and the decrease of the estimation precision due to the competition between the models. A multilevel input-adaptive multiple model (IAMM) algorithm is proposed, in which the inputs are adjusted according to the prior value and the on-line estimated maneuver parameters by introducing a dualistic distribution. The adaptabilities of the inputs can depict the actual maneuver process better compared with the static multilevel inputs. The simulation proves the effectiveness of IAMM algorithm compared with the IMM (Interacting Multiple Model) estimator with models containing multilevel static inputs.

1. INTRODUCTION

Multiple-model methods are vastly adopted in highly maneuvering target tracking. Successful application of any multiple-model algorithm depends on the proper corresponding designs, which mainly includes parametric designs, structural designs and hybrid designs etc. (Li, 2005). The typically designed parameters include acceleration inputs, process noise levels and turn rates. The possible acceleration is quantized into multiple known levels as the acceleration inputs of several dynamic models used in the multiple-model methods. The probabilities of every model are updated according to the measurements and the switches among the models. A typical model set of this kind includes a CV model and several CA/Singer models with known constant acceleration inputs(Averbuch, 1991). The switches among the models are described as a time homogeneous Markov chain with a known transition probability matrix. Because the maneuvers of the target are stimulated by uncertain accelerating inputs, this design can reflect the actual maneuvering process to a certain extent. This kind of design has been firstly proposed in (Moose, 1975) and further researched and extended in (Averbuch, 1991, Li, 1999, 2000a, b). Under the assumptions of (Moose, 1975), the inputs are summed into a single combined input with probabilities as their weights based on GPB1 (generalized pseudo-Bayesian estimator of first order) algorithm. Based on IMM(interacting multiple model) estimator, (Averbuch, 1991) applies 1 CV model and 12 CA models with acceleration inputs symmetric distributed around zero to track a maneuvering target in a 2D region. To improve the performance and reduce the computation burden, (Li, 1999, 2000a, b) proposed one kind of VSMM(variable-structure multiple model) algorithms to dynamically adjust the model set. In present researches, models with discretization designs of acceleration and turn rates are widely used in VSMM algorithms.

The multiple-model algorithms based on this kind of acceleration design are faced with two problems: improving the quantization levels to cover more possible acceleration values would increase calculation complexity and may decrease the estimation performance for excessive competitions among all the models. Prior quantization of possible acceleration is a trade-off between calculation complexity and accuracy without considering the practical dynamic property of the target. Because the actual accelerating process is always continuous, thus the current series of the estimated states would contain the prediction for maneuver property during the next sampling period. This paper proposes a dynamic model to fuse the prior acceleration quantization design with on-line maneuver information, which makes every level of acceleration input adaptive without increasing the quantizing level. The corresponding multiple-model algorithm to the model is derived based on the IMM algorithm. Compared with the standard IMM algorithm using constant quantized multilevel inputs models, the proposed algorithm with adaptive models has a better performance when tracking highly maneuvering target with continuous accelerations, which decreases the influence caused by the prior quantization.

2. ADAPTIVE MULTILEVEL ACCELERATION INPUTS MODEL

Consider the following model:

$$x_i(k+1) = \Phi_i(k)x_i(k) + B_i(k)u_i(k) + \Gamma_i(k)\omega_i(k) \tag{1}$$

$$z(k+1) = H_i(k+1)x_i(k+1) + \varepsilon_i(k+1), \quad k = 1,2,... \tag{2}$$

Where $i = 1,\cdots,r$ is the index of model, r is the amounts of models, $x_i(k+1) \in R^n$ is the system state vector, $u_i(k)$ is the input, $z(k+1) \in R^m$ is the measurement vector, $\Phi_i(k)$ is the state transition matrix, $B_i(k)$ is input gain, $\Gamma_i(k)$ is the gain of process noise, $H_i(k+1)$ is measurement matrix, $\omega_i(k)$ and $\varepsilon_i(k+1)$ are, respectively, the discrete-time process noise and measurement noise, assumed to be mutually independently with covariances, respectively, $Q(k)$ and $R(k+1)$. In the following parts, the next symbols will be used: mode $M_i(k)$ indicates the model i matches the actual model at time k, $\hat{x}_i(k) \triangleq E[x(k) \mid M_i(k), Z^k]$, $P(k) \triangleq E\{[\hat{x}(k) - x(k)][\cdot]^T \mid Z^k\}$, $\hat{x}_i(k+1 \mid k) \triangleq E[x(k+1) \mid M_i(k+1), Z^k]$, $\mu_i(k) \triangleq P\{M_i(k) \mid Z^k\}$ $\hat{x}(k) \triangleq E[x(k) \mid Z^k]$, $P_i(k) \triangleq E\{[\hat{x}_i(k) - x(k)][\cdot]^T \mid M_i(k), Z^k\}$, $Z^k \triangleq \{z(1), z(2), \cdots, z(k)\}$, $P_i(k+1 \mid k) \triangleq E\{[\hat{x}_i(k+1 \mid k) - x(k+1)][\cdot]^T \mid M_i(k+1), Z^k\}$. And the switches among the models are assumed as a time homogeneous Markov chain with a known transition probability matrix $P \in R^{r \times r}$, the elements of which are denoted as: p_{ij}, $i, j = 1,\cdots,r$.

For the purpose of tracking highly maneuvering target and under the assumption that the maneuvers are caused by acceleration inputs, CA models with multilevel acceleration inputs are considered. Then in (1), the system state is $x_i(k+1) = \begin{bmatrix} p_i(k+1) & v_i(k+1) & a_i(k+1) \end{bmatrix}$, the corresponding matrixes are:

$$\Phi_i(k) = \begin{bmatrix} 1 & T & T^2/2 \\ 0 & 1 & T \\ 0 & 0 & 1 \end{bmatrix}, \quad B_i(k) = \Gamma_i(k) = \begin{bmatrix} T^3/6 \\ T^2/2 \\ T \end{bmatrix} \tag{3}$$

where T is the sampling period. The input $u_i(k)$ is a quantized value to make acceleration within its possible value span. Different models correspond to different inputs. For the structure fixed multiple model algorithms, the input levels are predetermined. In these algorithms, the estimation performance depends on the quantization steps or amounts of the levels. Increasing the amounts would improve the performance to a certain extent, but that would also increase the computation burden. Large amounts of models should be considered while the practical acceleration alters continuously. In a further aspect, too many models will decrease the estimation accuracy and thus impair the efficiency because of the unnecessary competition among the models. In the third generation of multiple model algorithms, the VSMM algorithms are designed to adjust the model set to keep most necessary ones, and many research fruits have been presented. For the algorithms of this kind, the task of producing new filters systematically in a general setting needs some new supporting theories. And the main drawback of the most algorithms of this kind is their sophistication (Li, 2005).

In the following subsections, a multiple model algorithms is designed to make every acceleration input adaptive with the on-line maneuver information, which makes the algorithm adaptive to reflect the dynamic properties of the maneuvering target, while keeping the scale of the model set unchanged.

2.1 Multiple model estimation with on-line parameters

Consider multiple model estimation algorithms, according to the total probability theorem:

$$\begin{aligned} &p[x(k+1) \mid Z^{k+1}] \\ &= \sum_{j=1}^{r} p[x(k+1) \mid M_j(k+1), Z^{k+1}] P\{M_j(k+1) \mid Z^{k+1}\} \\ &= \sum_{j=1}^{r} p[x(k+1) \mid M_j(k+1), z(k+1), Z^k] \mu_j(k+1) \end{aligned} \tag{4}$$

where:

$$\begin{aligned} &p[x(k+1) \mid M_j(k+1), z(k+1), Z^k] \\ &= \frac{p[z(k+1) \mid M_j(k+1), x(k+1)]}{p[z(k+1) \mid M_j(k+1), Z^k]} p[x(k+1) \mid M_j(k+1), Z^k] \end{aligned} \tag{5}$$

Expand the second item in the right side of the last equation:

$$\begin{aligned} &p[x(k+1) \mid M_j(k+1), Z^k] \\ &= \sum_{i=1}^{r} p[x(k+1) \mid M_j(k+1), M_i(k), Z^k] P\{M_i(k) \mid M_j(k+1), Z^k\} \\ &\approx \sum_{i=1}^{r} p[x(k+1) \mid M_j(k+1), M_i(k), \{\hat{x}^l(k), P^l(k)\}_{l=1}^{r}, \\ &\qquad \{\hat{x}(s), P(s)\}_{s=1}^{k-1}] \mu_{i|j}(k) \\ &= \sum_{i=1}^{r} p[x(k+1) \mid M_j(k+1), \hat{x}^i(k), P^i(k), \{\hat{x}(s), P(s)\}_{s=1}^{k-1}] \mu_{i|j}(k) \end{aligned} \tag{6}$$

where $\mu_{i|j}(k) \triangleq P\{M_i(k) \mid M_j(k+1), Z^k\}$.

In the standard IMM algorithm, Z^k are approximated with $\{\hat{x}^l(k), P^l(k)\}_{l=1}^{r}$. To reserve more information about the maneuver, we approximate Z^k with $\{\hat{x}^l(k), P^l(k)\}_{l=1}^{r}$ and the estimated states series: $\{\hat{x}(s), P(s)\}_{s=1}^{k-1}$. Because the estimated states series reflects the moving process of the target, including it into the algorithm may provide more maneuver information to modify the parameter of multilevel inputs.

For the model (1), the following equations exist:

$$\begin{aligned} &E[a_i(k+1)] = E[a_i(k)] + T \bullet E[u_i(k)] \\ &\Rightarrow E[u_i(k)] = (E[a_i(k+1)] - E[a_i(k)])/T \end{aligned} \tag{7}$$

Considering the continuity of the practical altering acceleration, we assume that the change rate of the acceleration at time k is correlated with the one at time $k-1$. For all the estimated states, $\left\{\hat{x}^i(k), P^i(k), \{\hat{x}(s), P(s)\}_{s=1}^{k-1}\right\}$, are included into the pdf(probability density function) of the state vector's one-step prediction, as shown in (6), the approximate posterior estimate of the actual change rate of the acceleration $u_i(k-1)$ can be derived:

$$\hat{u}_i(k-1) \approx \left(\hat{a}_i(k) - \hat{a}(k-1)\right)/T \tag{8}$$

where $\hat{a}_i(k)$ and $\hat{a}(k-1)$ are respectively the acceleration member variable of the estimated state vectors $\hat{x}^i(k)$ and $\hat{x}(k-1)$. Reserving the $\hat{u}_i(k-1)$ and neglecting other history of the estimated states make (6):

$$\begin{aligned} & p[x(k+1) \mid M_j(k+1), Z^k] \\ & \approx \sum_{i=1}^{r} p[x(k+1) \mid M_j(k+1), \hat{x}^i(k), P^i(k), \hat{u}_i(k-1)] \mu_{i|j}(k) \end{aligned} \tag{9}$$

Here $\hat{u}_i(k-1)$ is defined as on-line maneuver parameter, thus (9) is the pdf of the state vector's one-step prediction at time $k+1$.

2.2 Dynamic models with multilevel adaptive acceleration inputs

To fuse the on-line maneuver parameter with the prior defined input of every level, a kind of explanation on the relationship between every model and its corresponding input is adopted here. Assume the actual acceleration $a(k) \in [-A_{\max}, A_{\max}]$, then the acceleration rate $u(k) \in [-2A_{\max}/T, 2A_{\max}/T]$. The state space of $u(k)$ can be partitioned into r parts as:

$$u(k) \in \bigcup_{i=1}^{r} S_i, \qquad S_i \cap S_j = \varnothing (i \neq j) \tag{10}$$

where S_i is a state subspace of u.

Defining:

$$M_j(k+1) \triangleq \{u(k) \in S_j\} \tag{11}$$

Substitution of (11) into (4) yields:

$$\begin{aligned} p[x(k+1) \mid Z^{k+1}] = \sum_{j=1}^{r} & p[x(k+1) \mid \{u(k) \in S_j\}, Z^{k+1}] \\ & \bullet P\{\{u(k) \in S_j\} \mid Z^{k+1}\} \end{aligned} \tag{12}$$

As to the pdf $p[x(k+1) \mid \{u(k) \in S_j\}, Z^{k+1}]$ in (12), $M_j(k+1) \triangleq \{u(k) \in S_j\}$ is an abstract description of the input $u(k)$. To simplify the expansion of (12) and make the estimation realizable, the pdf is approximated as:

$$\begin{aligned} & p[x(k+1) \mid \{u(k) \in S_j\}, Z^{k+1}] \\ & \approx p[x(k+1) \mid \{E[u(k)] = u'_j(k), \mathrm{cov}(u(k)) = \sigma(k)_j^2\}, Z^{k+1}] \end{aligned} \tag{13}$$

where $\{u(k) \in S_j\}$ is replaced with a Gaussian distributed variable with $u_j(k)$ and $\sigma(k)_j^2$ respectively as its mean and covariance. Off-line design of the input models of this kind is reviewed in (Li, 2002).

While all the other assumptions about the properties of the multiple model, including the Markov assumption etc., left unchanged, (12) and (13) provide an physical explanation of every level of the acceleration inputs $u_j(k)$, i.e., $u_j(k)$ and the process noise $\omega_i(k)$ in (1) can be treated as an approximation of the condition $\{u(k) \in S_j\}$, and all the conditions together compose the possible state space of the acceleration rate $u(k)$. Thus, comparing (1) (3) with (13) yields:

$$u_i(k) = u'_j(k), \qquad Q_i(k) = \sigma_j^2(k), \qquad i = j \tag{14}$$

In other words, for the multilevel acceleration input model (1), every input $u_i(k)$ have an subspace of effect: S_j.

As defined in (9), $\hat{u}_i(k-1)$ is the on-line maneuver parameter, which reflects the actual acceleration rate during $k-1$ th sampling period. To fuse $\hat{u}_i(k-1)$ with the predefined acceleration input $u_i(k)$, referring to the ternary-uniform mixture distribution adopted in (Singer, 1970), we propose that the acceleration input of model i during the k th sampling period obeys to the following binary distribution: when $\hat{u}_i(k-1) \in S_i$, the probability of predefined input $u_i(k)$ is P_i, the probability of $\hat{u}_i(k-1)$ is $1-P_i$; otherwise, when $\hat{u}_i(k-1) \notin S_i$, the probability of predefined input $u_i(k)$ is 1, the probability of $\hat{u}_i(k-1)$ is 0. With this distribution assumption, the mean and the covariance of model i's input during the k th sampling period are:

$$\begin{cases} \begin{cases} \bar{u}_i(k) = P_i u_i(k) + (1-P_i)\hat{u}_i(k-1) \\ Q_i^u(k) = P_i(1-P_i)\left(u_i(k) - \hat{u}_i(k-1)\right)^2 \end{cases}, & \hat{u}_i(k-1) \in S_i \\ \begin{cases} \bar{u}_i(k) = u_i(k) \\ Q_i^u(k) = 0 \end{cases}, & \hat{u}_i(k-1) \notin S_i \end{cases} \tag{15}$$

To apply standard Kalman filter to every modified model, Gaussian variable with the same first order and second order as in (15) is used to replace the modified input. Thus the modified model can be formulated as:

$$x_i(k+1) = \Phi_i(k)x_i(k) + B_i(k)\bar{u}_i(k) + B_i(k)\omega_i^u(k) + \Gamma_i(k)\omega_i(k) \tag{16}$$

where $\bar{u}_i(k)$ is the same as in (15), $\omega_i^u(k)$ and $\omega_i(k)$ are uncorrelated Gaussian noises with variances $Q_i^u(k)$ and $Q_i(k)$ respectively. It should be noticed that this replacement is an approximation to simplify the calculation.

Let $P_i = 1$ or $\hat{u}_i(k) = u_i(k)$, then $\bar{u}_i(k) = u_i(k)$ and $Q_i^u = 0$, (16) is transformed into the standard model with multilevel inputs. In other words, dynamic model (1) is a special form of (16). Because P_i is defined offline, the adaptability of model (16) depends directly on the maneuver parameter $\hat{u}_i(k-1)$, which is obtained in (8).

3. MULTIPLE-MODEL ALGORITHM BASED ON MODELS WITH ADAPTIVE INPUTS

Approximate the last term of (9) with a single Gaussian probability function:

$$
\begin{aligned}
& p[x(k+1) \mid M_j(k+1), Z^k] \\
& = \sum_{i=1}^{r} N[x(k+1); E[x(k+1) \mid M_j(k+1), \hat{x}^i(k), \hat{u}_i(k-1)] \\
& \qquad , \mathrm{cov}[\bullet]] \cdot \mu_{i|j}(k) \\
& \approx N[x(k+1); \sum_{i=1}^{r} E[x(k+1) \mid M_j(k+1), \hat{x}^i(k), \hat{u}_i(k-1)] \\
& \qquad \cdot \mu_{i|j}(k), \mathrm{cov}[\bullet]]
\end{aligned} \tag{17}
$$

If the one-step prediction function of the Kalman filter is linear, (17) can be further simplified to the IMM algorithm with mixed initial condition for the filter matched to $M_j(k+1)$ using $\{\hat{x}^i(k), P^i(k)\}_{i=1}^r$ at time $k+1$. Nevertheless, for the dynamic model (16), the variance $Q_i^u(k)$ is a nonlinear equation of $u_i(k)$ and $\hat{u}_i(k-1)$ as shown in (15). Thus, the interacting process can only be executed to the one-step predictions of the Kalman filters matched to $M_j(k+1)$ using every estimated results acquired at time k, which includes $\{\hat{x}^i(k), P^i(k)\}_{i=1}^r$ and $\{\hat{u}_i(k-1)\}_{i=1}^r$, as shown in the last term of (17).

Based on the above equations, the multi-level input adaptive multiple-model algorithm can be synthesized as the following:

1. Construction of the adaptive models with multi-level inputs.

Firstly the on-line maneuver parameter is estimated based on (8), and the mean and variance of the acceleration rate input is calculated using (15). After this, the dynamic equation (16) of model j is generated.

2. One-step prediction

Based on (16), for model $i = 1, \cdots, r$:

$$\hat{x}_i(k+1 \mid k) = \Phi_i(k)\hat{x}_i(k) + B_i(k)\overline{u}_i(k) \tag{18}$$

$$
\begin{aligned}
P_i(k+1 \mid k) = & \Phi_i(k)P_i(k)\Phi_i(k)^T + B_i(k)Q_i^u(k)B_i(k)^T \\
& + \Gamma_i(k)Q_i(k)\Gamma_i(k)^T
\end{aligned} \tag{19}
$$

3. Interacting and mixing the one-step predictions

For model $i, j = 1, \cdots, r$:

$$\hat{x}_{0j}(k+1 \mid k) = \sum_{i=1}^{r} \hat{x}_i(k+1 \mid k)\mu_{i|j}(k) \tag{20}$$

$$
\begin{aligned}
P_{0j}(k+1 \mid k) = & \sum_{i=1}^{r} \mu_{i|j}(k)\{P_j(k+1 \mid k) \\
& + [\hat{x}_i(k+1 \mid k) - \hat{x}_i(k+1 \mid k)][\cdot]^T\}
\end{aligned} \tag{21}
$$

Same as in IMM algorithm, the probability $\mu_{i|j}(k)$ is:

$$\mu_{i|j}(k) = p_{ij}\mu_i(k)\left(\sum_{i=1}^{r} p_{ij}\mu_i(k)\right)^{-1} \tag{22}$$

4. One-step prediction

Updating the state with the measurement at time $k+1$:

$$\hat{e}_j(k+1) = z(k+1) - H_j\hat{x}_{0j}(k+1 \mid k) \tag{23}$$

$$S_j(k+1) = H_j(k+1)P_{0j}(k+1 \mid k)H_j(k+1)^T + R_j(k+1) \tag{24}$$

$$\hat{x}_j(k+1) = \hat{x}_{0j}(k+1 \mid k) + K_j(k+1)\hat{e}_j(k+1) \tag{25}$$

$$P_j(k+1) = (I - K_j(k+1)H_j(k+1))P_{0j}(k+1 \mid k) \tag{26}$$

$$K_j(k+1) = P_{0j}(k+1 \mid k)S_j(k+1)^{-1} \tag{27}$$

5. Mode probability update and estimate combination

The estimate results are combined to output the final estimation of state and its covariance according to (4):

$$\hat{x}(k+1) = \sum_{j=1}^{r} \hat{x}_j(k+1)\mu_j(k+1) \tag{28}$$

$$P(k+1) = \sum_{j=1}^{r} \mu_j(k+1)\{P_j(k+1) + [\hat{x}_j(k+1) - \hat{x}(k+1)][\cdot]^T\} \tag{29}$$

where:

$$
\begin{aligned}
& \mu_j(k+1) \\
& \triangleq P\{M_j(k+1) \mid Z^{k+1}\} \\
& = \frac{1}{c} p[z(k+1) \mid M_j(k+1), Z^k]P\{M_j(k+1) \mid Z^k\} \\
& = \frac{1}{c}\Lambda_j(k+1)\sum_{i=1}^{r} P\{M_j(k+1) \mid M_i(k), Z^k\}P\{M_i(k) \mid Z^k\} \\
& = \frac{1}{c}\Lambda_j(k+1)\sum_{i=1}^{r} p_{ij}\mu_i(k)
\end{aligned} \tag{30}
$$

where $c = \sum_{j=1}^{r} \Lambda_j(k+1)\overline{c}_j$, $\overline{c}_j = \sum_{i=1}^{r} p_{ij}\mu_i(k)$, p_{ij} is the known mode transition probabilities in the transition matrix P.

The likelihood functions

$$
\begin{aligned}
\Lambda_j(k+1) & = N[\hat{v}_j(k+1); 0, S_j(k+1)] \\
& = [(2\pi)^m \mid S_j \mid]^{-1/2} \exp\left(-\frac{1}{2}\hat{v}_j^T S_j^{-1}\hat{v}_j\right)
\end{aligned} \tag{31}
$$

4. SIMULATION RESULTS

Because the proposed adaptive model is coordinate-uncoupled, without losing generality, target tracking of a maneuvering object moving at one direction is considered during the simulation. The measurements of the target are assumed to be the positions, and CA models are adopted, thus the measurement matrix is $H_i(k+1) = \begin{bmatrix} 1 & 0 & 0 \end{bmatrix}$. The sampling period is assumed to be one second: $T = 1.0s$. Initially, the target was at the position [2000m], with velocity 20 m/s flew at constant velocity for the first 5 seconds. Then the target maneuvered with the constant acceleration 30 m/s^2 from 5s to 15s and returned to constant velocity motion from 15s to 25s. The target then performed a maneuver during 25s to 45s with acceleration $a = r\omega^2 \sin\omega(t-25)$, where r =3000 m, ω =0.1 rad/s. The results are the root mean square(RMS) values of position, velocity and acceleration estimation errors with 1000 simulation experiments.

The simulation compares the performance of IAMM and IMM algorithms, the following assumptions are made:

1. The CA models with three input level are adopted. The corresponding parameters of CA model are the same as in (3). The values of levels are respectively $\{u_i, i=1,2,3\}=\{-u_{\max}, 0, u_{\max}\}$, where $u_{\max}$ =8m/s^3. The known transition matrix of the modes is $P=\begin{bmatrix}0.90 & 0.05 & 0.05\\ 0.05 & 0.90 & 0.05\\ 0.05 & 0.05 & 0.90\end{bmatrix}$. It needs to be noticed that the simulation just compare the performances of maneuvering target tracking, no CV model is included. In practice, CV can be included to improve the tracking performance while no maneuver happens. The subspaces corresponding to the levels u_i are assumed respectively as: $\{S_i, i=1,2,3\}$ = {[-60 m/s^3, -1 m/s^3), [-1 m/s^3, 1 m/s^3], (1 m/s^3, 60 m/s^3)}.

2. Variances of process noises are $Q_i(k)$ =0.001 m^2/s^6, i =1,2,3. Variance of the measurement noise: $R=100m^2$.

3. In IAMM algorithm, the prior probability of $u_i(k)$ in every model is P_i =0.8, i =1,2,3. Other parameters used in IAMM algorithm are the same as in IMM algorithm.

Figure 1, figure2 and figure 3 compare the root-mean-square(RMS) error of position, velocity and acceleration estimations of the two algorithms. Table 1 shows the root mean square of the accumulated estimation error. According to the simulation, the two algorithms have similar performance when no maneuver happens. When target moves at a uniform acceleration, IAMM and IMM algorithms perform comparably(from 5s to 15s). Because the IAMM can only adapt with altering acceleration, when accelerating uniformly, the acceleration parameter adopted in the IAMM algorithm keeps steady, which makes the IAMM algorithm reduced into standard IMM algorithm with static inputs.

When target maneuvers with time-varying accelerations, IAMM performance better than IMM algorithm on either position, velocity and acceleration estimation, which can be found in the RMS error of estimation during 25s and 45s shown in the figures. When target acceleration altering continuously, the maneuver parameter $\hat{u}_i(k-1)$ can reflect the on-line maneuver characteristics. IAMM algorithm modifies the mean and the variance of the multilevel inputs according to the real-time maneuver status, which makes the considered models contain acceleration rates closer to the real ones. This modification makes the IAMM performs better when accelerating process happens as shown in the figures and table 1.

The probability P_i used in (15) actually indicates the degree of confidence of the predefined input. $P_i=1$ means that the prior input parameter totally coincides with the practical parameters. Because the probability of the on-line maneuver parameter $\hat{u}_i(k-1)$ is $1-P_i$, decrease P_i can include more on-line maneuver information, which means the models process more adaptabilities. Whereas, because the estimation error of the $\hat{u}_i(k-1)$, more uncertainties will also be included into the state estimation. Simulations show that proper compromise can be acquired when P_i between 0.8 and 0.98.

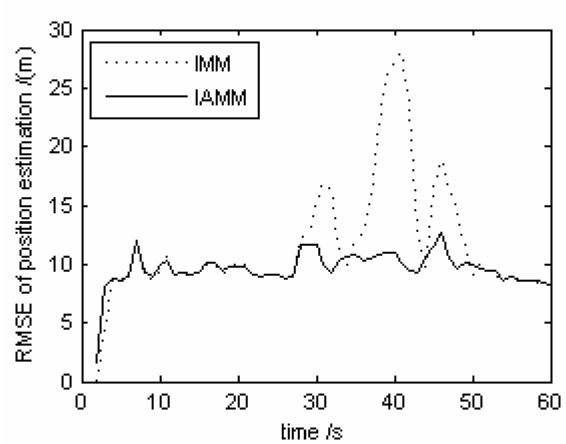

Fig.1. RMS error of position estimation

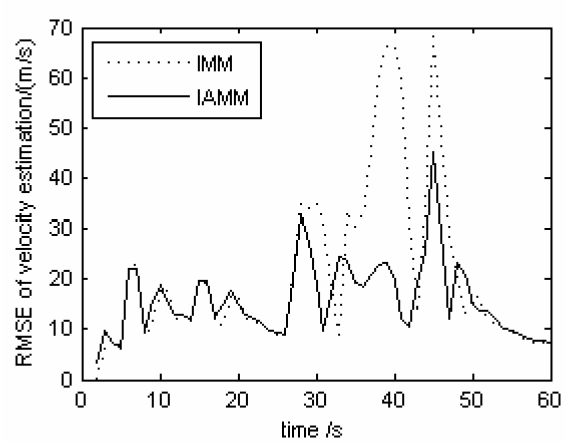

Fig.2. RMS error of velocity estimation

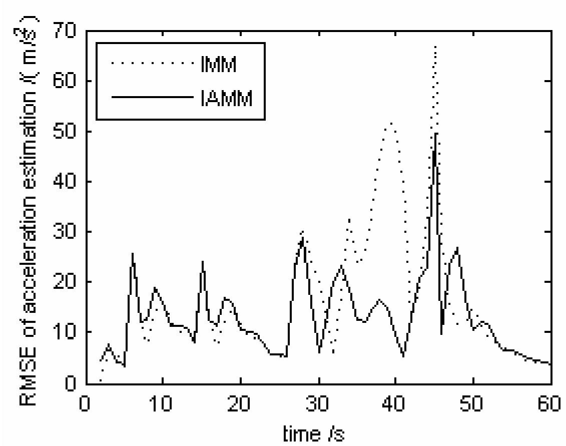

Fig.3. RMS error of position acceleration

Table 1. Accumulated Estimation Error(Root Mean Square)

	IMM	IAMM
Acceleration(m/s^2)	21.3101	15.3215
Velocity(m/s)	26.9158	17.4052
Position(m)	12.6842	9.6692

5. CONCLUSIONS

In this paper, an IAMM algorithm is presented for highly maneuvering target tracking. Based on the multiple model algorithms with multilevel-input models, the IAMM algorithm introduces an on-line estimated maneuver parameter to modify the predefined input parameters, and the estimation structure is re-derived according to the modification. The modification makes the multiple models adaptive. Compared with the IMM algorithm with same amounts of models, IAMM is proved effective when tracking highly maneuvering target with continuously altering accelerations by the simulation without increasing models. IAMM algorithm needs a probability to reflect the confidence of the predefined multilevel inputs. Proper value of the probability can acquire a compromise between the precision of the steady state estimation and the maneuvering tracking performance. A value span of this probability is presented by simulation and recommended as a general reference for the application of the IAMM algorithm, and its precise value needs further discussions.

REFERENCES

Averbuch A., Itzikowitz S., and Kapon T. (1991). Parallel Implementation of Multiple Model Tracking Algorithms. *IEEE Transactions on Parallel and Distributed Systems*, 1991, **2,** 242-252.

Li X. R., Zhang Y. M., and Zhi X. R. (1999). Multiple-Model Estimation with Variable Structure-Part IV: Design and Evaluation of Model-Group Switching Algorithm. *IEEE Transactions on Aerospace and Electronic Systems*, **35,** 242-254.

Li X. R. and Zhang Y. M. (2000). Multiple-Model Estimation with Variable Structure-Part V: Likely-Model Set Algorithm. *IEEE Transactions on Aerospace and Electronic Systems*, **36,** 448-466.

Li X. R. and Jilkov V. P. (2000). Survey of maneuvering target tracking: dynamic models. *Proceedings of SPIE: Signal and Data Processing of Small Targets 2000,* **4048,** 212-235.

Li X. R. (2002). Model-Set Design for Multiple-Model Estimation-Part I. Proceedings of International Conference on Information Fusion 2002, **1**, 26–33.

Li X. R. and Jilkov V. P. (2005). Survey of maneuvering target tracking. Part V. Multiple-model methods. *IEEE Transactions on Aerospace and Electronic Systems*, **41,** 1255-1321.

LIANG Yan, PAN Quan, JIA Yugang, ZHOU donghua, Strong-Tracking Multiple Model Estimator , Acta Electronica Sinica, 2002, 30(1): 34-37.（in Chinese）

ZANG Rong-chun; CUI Ping-yuan. Research on Adaptive Markov Parameter IFIMM Algorithm. Acta Electronica Sinica, 2006, 34 (3) : 521-524. （in Chinese）

Moose R. L. (1975). An Adaptive State Estimation Solution to the Maneuvering Target Problem. *IEEE Transactions on Automatic Control*, **20,** 359-362.

Singer R. A. (1970). Estimating optimal tracking filter performance for manned maneuvering targets. *IEEE Transactions on Aerospace and Electronic Systems,* **6,** 473-483.

Zhou H. and Kumar K. S. P. (1984). A "Current" Statistical Model and Adaptive Algorithm for Estimating Maneuvering Targets. *AIAA Journal of Guidance,* **7,** 596-602.