International Journal of Control and Automation
Vol.9, No.2 (2016), pp.103-118
http://dx.doi.org/10.14257/ijca.2016.9.2.11

Probability Threshold Scheme in Fast Model-set Adaptive IMM

Hua Wu[1,*], Jianping Xing[2] and Liu Yang[1]

[1]School of information science and electrical engineering, Shandong Jiaotong University 250357 Jinan, China

[2] School of information science and engineering, Shandong University 250100 Jinan, China

**wuhua1982111@126.com*

Abstract

This study is devoted to overcome the underflow problem and poorly cost-effective limitation of model-set adaptive IMM algorithm. Cause of underflow problem in Novel-IMM is addressed firstly, based on which an underflow prevented selection probabilities (UPSP) algorithm is presented to solve this problem. This paper then presents a fast model-set adaptive (FAIMM) IMM algorithm based on steady state Kalman filters that decrease the computational burden greatly while keeping acceptable tracking accuracy. Finally, the threshold choosing strategy of UPSP algorithm is presented, which could make the FAIMM algorithm achieves ideal performance. Simulation results demonstrate that the FAIMM algorithm can be an effective estimator in real-time application.

Keywords*: Model-set adaptive IMM, Steady state Kalman filters, Underflow prevented selection probabilities (UPSP) algorithm, FAIMM*

1. Introduction

For track-while-scan problems, Multiple-model (MM) methods have been generally regarded as the mainstream approach to maneuvering target tracking under motion-mode uncertainty condition. The first generation Multiple Model approach was proposed by Magill [1], which is effective for problems involving infrequent mode transitions. However, this MM estimator ignores the interaction of different models, which is not well suited to situations with frequent modal changes [2, 3].

To solve this problem, Blom and Bar-Shalom proposed a novel algorithm called Interacting Multiple Model (IMM) estimator with Markovian swiching coefficients [4]. This is the second generation MM estimator. The IMM estimator is one of the most cost-effective estimators estimating the state of a dynamic system with several behavior modes that can “switch” from one to another, which makes the MM approach able to track maneuvering target. Many literatures have put great effort into improving the IMM estimator both in tracking accuracy and computational efficiency [5-8]. However, these methods are not effective in handling systems with large numbers of models, since performances of these estimators will deteriorate if too many models are used due to the excessive competition from the unnecessary models [14].

To overcome the dilemma between model increment and performance degradation, Xiao-Rong Li and Bar-Shalom presented a variable structure MM (VSMM) estimator based on partitioning the range of the mapping defined by the state dependency of system mode set [10], which is called the third generation MM estimator that opened up a new area in MM estimation. The VSMM estimator not only inherits the first generation’s superior rule for estimate fusion and the second generation’s effective cooperation mechanisms, but also adapts to the outside world by producing new elemental filters if the existing ones are not good enough and by eliminating those elemental filters that are harmful [3].

ISSN: 2005-4297 IJCA

Specializing in different problems, all the three-generation MM estimators have their own advantages [2,3]. To realize model set adaptation without auxiliary information, Qu proposed a Novel-IMM estimator that employs independent parallel model set without using the model set adaptation scheme of VSMM, which combines advantages of the second and third generation MM estimators [10]. In Novel-IMM algorithm, there are *N* independent IMM filters operating in parallel, and the model set adaptation process is achieved by MSPT (model set probability test) algorithm without any auxiliary information. However, two challenges block the application of Novel-IMM. The most harmful one is that a numerical underflow problem may occur when calculating the model sets selection probabilities for Novel-IMM under some scenarios, which would lead to unacceptable deterioration in estimates. Besides, the Novel-IMM algorithm occupies too much computational resources, since several IMM filters are used in this algorithm. Thus, it is rather difficult for Novel-IMM to be put into real-time application unless the above two problems are solved.

Aiming at keeping availabilities of all model sets and decreasing the computational burden for Novel-IMM, firstly, this paper proposes an underflow prevented selection probabilities (UPSP) algorithm, which avoids underflow problem by setting appropriate probability threshold scheme. Then, the fast model-set adaptive IMM (FAIMM) algorithm is proposed to decrease executing time by introducing steady state Kalman filters. Finally, the threshold choosing strategy of UPSP algorithm is presented.

The remaining paper is organized as follows: Section 2 analyzes the mechanism of underflow problem in standard Novel-IMM and proposes the UPSP algorithm to solve this problem. Section 3 describes the steady state Kalman filters to derive the FAIMM algorithms. Section 4 demonstrates the effectiveness of proposed algorithms by simulation examples of target tracking. Finally, the conclusion is presented in Section 5.

2. Novel-IMM and Underflow Problem

2.1. Standard Novel-IMM Algorithm

The standard Novel-IMM consists of several independent IMM filters operating in parallel, and these IMM filters do not interact with each other, which could avoid competitions from unsuitable model sets. To get optimal estimate from potential candidates, the Novel-IMM algorithm proposed a model set probability test (MSPT) algorithm to realize model set adaptation without auxiliary information under certain conditions. Supposing the Novel-IMM is consist of *N* independent IMM filters, the procedures of standard Novel-IMM can be expressed as follows:

Step 1: Process *of each independent IMM*. These *N* IMM filters operate separately to produce their outputs in this step.

Step 2: Calculating *likelihood function value of each model set (or IMM)*:

$$\Lambda_j(k+1) = \sum_i^r \Lambda_j^i(k+1)\mu_j^i(k),\ j = 1, 2, \ldots N \tag{1}$$

where $\Lambda_j(k+1)$ is the *j*th model-set likelihood function value at discrete time *k+1*, while $\mu_j^i(k)$ and $\Lambda_j^i(k+1)$ are weighting probability and likelihood function value of the *i*th model in the *j*th model set, respectively.

Step 3: Calculating *selection probability of each model set*.

$$\eta_j(k+1) = \frac{\Lambda_j(k+1)\eta_j(k)}{\sum_{j=1}^N \Lambda_j(k+1)\eta_j(k)},\ j = 1, 2, \ldots N \tag{2}$$

in which $\eta_j(k+1)$ is the selection probability of the *j*th model set.

Step 4: *Choosing output estimate and covariance error matrix.*

$$\eta_m(k+1) = \max_{j=1,\dots N}\{\eta_j(k+1)\}$$

(3)

where m is index of the biggest selection probability, then, we get the final output estimate and covariance error matrix:

$$\hat{x}(k+1|k+1) = \hat{x}_m(k+1|k+1) \quad (4)$$

$$\mathrm{P}(k+1|k+1) = \mathrm{P}_m(k+1|k+1) \quad (5)$$

As each IMM filter operates independently, this algorithm avoids deterioration in performance caused by excessive competition from harmful model sets.

Further details about Novel-IMM algorithm can refer to [10].

2.2. Underflow Problem In Novel-IMM

To cover all possible modes of the true system, some model sets in this algorithm are necessarily quite different from the current system mode, which will lead to a selection probability underflow problem [11].

In Equation 1, $\Lambda_j^i(k)$ is computed by the following equation:

$$\Lambda_j^i(k) = \frac{1}{\sqrt{2\pi|\mathrm{A}_j^i(k)|}} \bullet \exp\{-\frac{1}{2}[x_0(k) - \hat{x}_j^i(k|k-1)]^{\mathrm{H}}[\mathrm{A}_j^i(k)]^{-1}[x_0(k) - \hat{x}_j^i(k|k-1)]\}$$

$$i=1,2,\dots n;\ j=1,2,\dots N \quad (6)$$

in which $x_0(k)$ is the measured state vector, $\hat{x}_j^i(k|k-1)$ and $\mathrm{A}_j^i(k)$ are predicted state vector and covariance matrix of innovation process derived from the ith model of jth IMM, respectively. When the ith model of jth IMM mismatch the current system mode greatly, $x_0(k) - \hat{x}_j^i(k|k-1)$ is likely to be very large while $\mathrm{A}_j^i(k)$ would not as large as it should be since it is calculated under assumption that model i is the correct model. Thus, the likelihood function value $\Lambda_j^i(k)$ will be extremely small because the normalized measurement residual squared $[x_0(k) - \hat{x}_j^i(k|k-1)]^{\mathrm{H}}[\mathrm{A}_j^i(k)]^{-1}[x_0(k) - \hat{x}_j^i(k|k-1)]$ is very large. As a result, when one of the model sets in Novel-IMM is very different with the current system mode, which can be assumed as model set j, the likelihood function value $\Lambda_j(k+1)$ is likely to be very small. Thus, its corresponding selection probability $\eta_j(k+1)$ will be extremely small. If this situation lasts in the next M moments, $\eta_j(k+1+M)$ will tend to be infinitely small or zero finally. From the definition of Equation 2, it is obviously that η_j will stay at zero even if the jth model set turns to be the most suitable one with true system mode in the following process. That is, the underflow problem occurred when calculating the model sets selection probabilities, which is a vital drawback for the application of standard Novel-IMM.

2.3. Solutions For Underflow Problem

To solve the above problem, we propose an underflow prevented selection probabilities (UPSP) algorithm by setting appropriate threshold for every model set in Novel-IMM. Differently with thresholds used in single model set IMM filter, the UPSP algorithm proposed here will not impact on performance of the chosen IMM, since there is no interaction or fusion between several parallel IMM filters used in Novel-IMM algorithm.

Firstly, in Equation 1, the standard Novel-IMM derives the *j*th IMM's likelihood function value $\Lambda_j(k+1)$ through weighting probabilities at the discrete time *k*, which would lead to mistakes because the mismatch between model likelihood function values and their weighting probabilities, especially when the system mode maneuvers. On this condition, the model-set likelihood function value can't reflect the real status of each model set exactly. Thus, we redefine the model-set likelihood function value as follows:

$$\Lambda_j(k) = \sum_i^r \Lambda_j^i(k)\mu_j^i(k)\ ,\ j = 1, 2, \ldots N \tag{7}$$

Then, following Equation 2, supposing there are *s* model sets come across the underflow problem in the *N* model sets, and the underflow threshold is denoted as T_L, the UPSP algorithm can be defined as follows:

$$\eta_{l_u}(k) = T_L\ ,\ u = 1,2\ldots s \tag{8}$$

$$\eta_{l_n}(k) = \frac{\eta_{l_n}(k)}{\sum_{n=s+1}^{N} \eta_{l_n}(k) + s \times T_L}\quad ,\ n = s+1,\ldots N \tag{9}$$

where l_u and l_n are indexes of model sets with and without underflow problem, respectively.

The total selection probabilities meet the following formula:

$$\sum_{j=1}^{N} \eta_j(k) = 1\ ,\ j = 1,2\ldots N \tag{10}$$

Figure 1 gives the flow chart of UPSP algorithm.

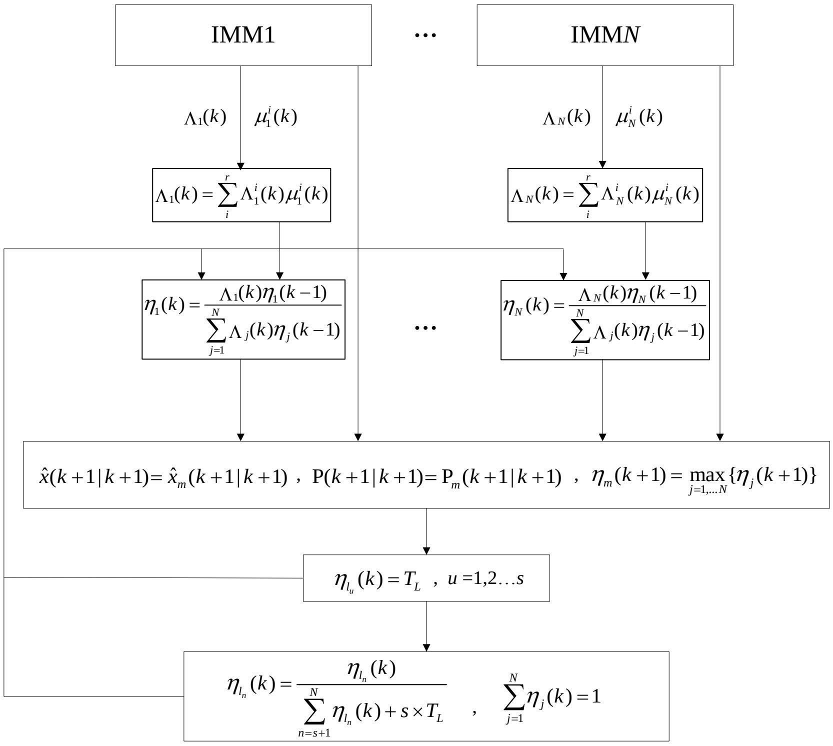

Figure 1. UPSP Algorithm

Note: An undesired situation would like to happen when one of the $\eta_{l_n}(k)$ is just a little larger than threshold T_L that $\eta_{l_n}(k)$ will turn out to be a little smaller than T_L after the process of Equation 9. However, it will not lead to an underflow problem because $\eta_{l_n}(k)$ will be very close to T_L and keep the corresponding model set "alive". More importantly, it will not prevent the proposed algorithms from choosing correct model set.

We will discuss performances of proposed algorithms based on UPSP algorithm with different T_L values, and present the principle to choose appropriate T_L in Section 4.

3. Cost-Effective Implementation of Model-Set Adaptive IMM

Another challenge that standard Novel-IMM confronts is the poor cost-effectiveness, which makes the algorithm difficult to be available in real-time system. Therefore, it is significant to decrease computational burden for standard Novel-IMM.

3.1. Steady State Kalman Filters

The complexity of Novel-IMM is determined by the structure of MSPT algorithm, which chooses the most suitable model set based on selection probabilities that calculated by their model-set likelihood function values. So it's a knotty problem and great challenge to improve the cost-effectiveness by transforming the structure of MSPT algorithm. Thus, decreasing the computational burden of sub-filters in each model set is the most practical and feasible method. Based on this consideration, we would like to introduce the steady state Kalman filters into the proposed algorithms.

α-β filter is the steady state of the second order Kalman filter. According to [12, 13], the α-β filter is defined as follows:

$$\hat{x}(k|k-1) = F\hat{x}(k-1|k-1) \tag{11}$$

$$\hat{x}(k|k) = \hat{x}(k|k-1) + K(x_0(k) - H\hat{x}(k|k-1)) \tag{12}$$

where $\hat{x}(k|k-1)$, $\hat{x}(k|k)$, $x_0(k)$ denote the predicted state vector, estimated state vector and measured state vector at the discrete time k, respectively; F, H, Q, K denote the state transfer matrix, measure matrix, process noise covariance matrix and filter gain matrix. And the filter gain matrix can be expressed as:

$$K = \begin{bmatrix} \alpha & 0 & \beta/T & 0 \\ 0 & \alpha & 0 & \beta/T \end{bmatrix}^T \tag{13}$$

where T is the sample interval, and α β are filter gain factors that can be expressed by one parameter that called tracking index:

$$\lambda = \frac{\sigma_\xi T^2}{\sigma_w} \tag{14}$$

in which σ_w and σ_ξ are standard deviations of the measurement noise and process noise, respectively. And α and β can be calculated by the following equations [7]:

$$\alpha = -(\lambda^2 + 8\lambda - (\lambda+4)\sqrt{\lambda^2+8\lambda})/8, \ \beta = (\lambda^2 + 4\lambda - \lambda\sqrt{\lambda^2+6\lambda})/4 \tag{15}$$

Similarly, α-β-γ filter is the steady state of the third order Kalman filter. According to [13], the α-β-γ filter is designed similarly with Equation 11 and 12, while the filter gain matrix is defined as:

$$K = \begin{bmatrix} \alpha & 0 & \beta / T & 0 & \gamma / 2T^2 & 0 \\ 0 & \alpha & 0 & \beta / T & 0 & \gamma / 2T^2 \end{bmatrix}^T \quad (16)$$

where the α, β and γ can be derived by the following equations:

$$\alpha = 1 - s^2 ,\ \beta = 2(1-s)^2 ,\ \gamma = \beta^2 / (2\alpha) \quad (17)$$

$$s = z - (\frac{p}{3z}) - \frac{b}{3} \quad (18)$$

$$z = -\sqrt[3]{-q - \sqrt{(q^2 + \frac{4p^3}{27}) / 2}} \quad (19)$$

in which

$$b = \frac{\lambda}{2} - 3 ,\quad p = c - \frac{b^2}{3} \quad (20)$$

with

$$c = \frac{\lambda}{2} + 3 ,\quad q = \frac{2b^3}{27} - \frac{bc}{3} - 1 \quad (21)$$

3.2. The Fast Model-Set Adaptive IMM (FAIMM) Algorithm

Aiming at resolving the underflow problem and improving cost-effectiveness of Novel-IMM, we propose the fast model-set adaptive IMM (FAIMM) algorithm, which not only adopts the proposed UPSP algorithm to keep every model set "alive" but also employs steady state Kalman filters to decrease the computational burden while keeping acceptable accuracy.

Following analyses and introductions above, we suppose the FAIMM algorithm is composed of N parallel IMM filters, then the FAIMM algorithm can be derived as follows:

Step 1: *Parallel cost-effective IMM*. Executing independent IMM filters in which the second-order and third-order Kalman filters are replaced by α-β and α-β-γ filters, respectively.

Step 2: *Calculating* likelihood *function value of each model set*. Just as introduced in Equation 7, we have redefined the model-set likelihood function value as follows:

$$\Lambda_j(k) = \sum_i^r \Lambda_j^i(k) \mu_j^i(k) ,\ j = 1, 2, \ldots N \quad (22)$$

Step 3: Calculating *selection probability of each model set*.

$$\eta_j(k) = \frac{\Lambda_j(k)\eta_j(k-1)}{\sum_{j=1}^N \Lambda_j(k)\eta_j(k-1)} ,\ j = 1, 2, \ldots N \quad (23)$$

Step 4: *Estimate output.*

$$\eta_m(k) = \max_{j=1,\ldots N} \{\eta_j(k)\} \quad (24)$$

where *m* is the index of the biggest selection probability, and we can get the final output estimate:

$$\hat{x}(k|k) = \hat{x}_m(k|k) \quad (25)$$

Step 5: *The* UPSP *algorithm.* After completing the estimate output process, we check and handle the underflow problem with UPSP algorithm: if $\eta_{l_u}(k) < T_L$, u=1,2…s, then carry out the following operations:

$$\eta_{l_u}(k) = T_L \text{ , } u=1,2\ldots s \quad (26)$$

$$\eta_{l_n}(k) = \frac{\eta_{l_n}(k)}{\sum_{n=s+1}^{N} \eta_{l_n}(k) + s \times T_L}, \quad n=s+1,\ldots N \quad (27)$$

where l_u and l_n are indexes of model sets with and without underflow problem, respectively. In addition, the above selection probabilities meet following formula:

$$\sum_{j=1}^{N} \eta_j(k) = 1 \text{ , } j=1,2\ldots N \quad (28)$$

4. Results and Analyses

4.1. Simulation Settings

Simulations are provided to verify the tests for FAIMM algorithm proposed in this paper. Similarly with literature [10], this section also uses two examples of maneuvering target tracking. Performance measures are position RMSE (root mean square error) and executing time based on 100 Monte Carlo runs.

Starting from $x = 10000m, y = 40000m$, the first trajectory of target is generated by different motions for five periods: 1) the target moves in a CV motion with velocity $v_x = 300m/s, v_y = 0m/s$ from 1s to 30s; 2) a CA motion with acceleration $a_x = -10m/s^2, a_y = -10m/s^2$ from 31s to 60s; 3) a CV motion from 61s to 90s; 4) a CT motion with turn rate $\omega = -0.19634375$ rad/s from 91s to 98s. Finally, the target ends the trajectory with a CV motion from 99s to 128s.

To test performances of the FAIMM algorithm more sufficiently, the second trajectory is generated by following motions with initial position $x = 10000m, y = 40000m$: 1) a CV motion with velocity $v_x = 0m/s, v_y = -300m/s$ from 1s to 50s; 2) a CA motion from 51s to 65s with acceleration $a_x = 20m/s^2, a_y = 20m/s^2$; 3) a CV motion form 66s to 115s; 4) a CA motion from 116s to 125s with acceleration $a_x = -30m/s^2, a_y = -30m/s^2$; 5) a CV motion from 126s to 185s; 6) a CT motion from 186s to 216s with turn rate $\omega = -0.10133871$ rad/s; 7) a CV motion from 217s to 266s; 8) a CT motion from 267s to 297s with turn rate $\omega = 0.10133871$ rad/s; 9) a CV motion from 298s to 347s.

In this section, the proposed FAIMM algorithm employs two model sets that consist of three models, respectively. The model set M_1 is consist of a CV model with process noise $\sigma_{\xi_{CV}} = 0.5m/s^2$ and two CA models with process noise $\sigma_{\xi_{CA1}} = 0.5m/s^2$ and $\sigma_{\xi_{CA2}} = 2m/s^2$. Model set M_2 is consist of a CV model with process noise $\sigma_{\xi_{CV}} = 0.5m/s^2$ and two CT models with opposite turn rates. Besides, the sample interval is T=1s, and the standard deviation of measurement noise and process noise in every single dimension are chosen to be $\sigma_w = 20m$ and $\sigma_\xi = 1m/s^2$. The model transition probability matrix used in each model set of standard Novel-IMM, UPSP based Novel-IMM and FAIMM algorithm is defined as:

$$p_{ij} = \begin{bmatrix} 0.95 & 0.025 & 0.025 \\ 0.025 & 0.95 & 0.025 \\ 0.025 & 0.025 & 0.95 \end{bmatrix}$$

the transition probability matrix used in standard IMM algorithm is:

$$p_{ij} = \begin{bmatrix} 0.95 & 0.05 \\ 0.05 & 0.95 \end{bmatrix}$$

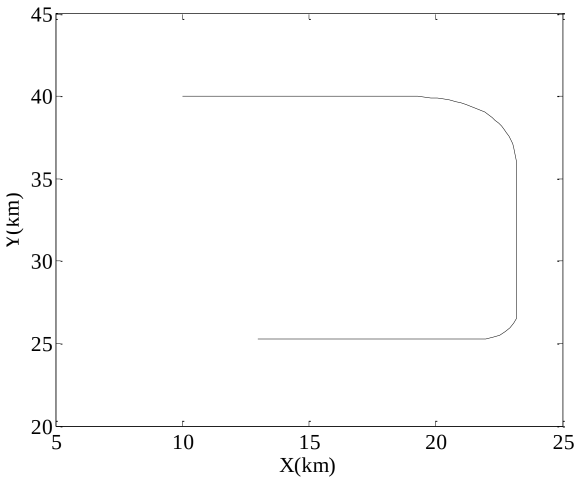

Figure 2. Trajectory a

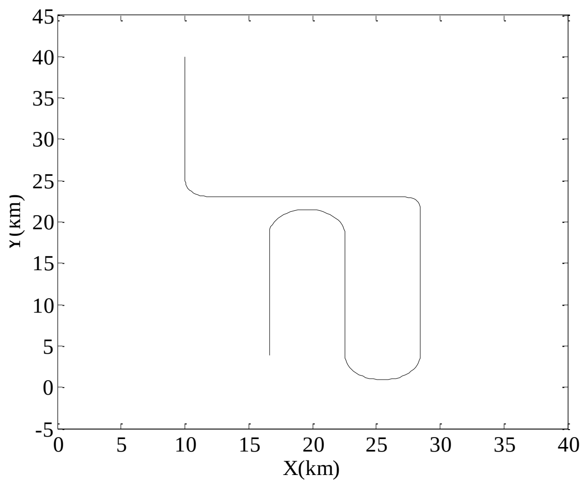

Figure 3. Trajectory b

The trajectory a and trajectory b of the target are shown in Figure 2 and Figure 3, respectively. Then, performances of IMM, standard Novel-IMM, UPSP based Novel-IMM and FAIMM will be discussed in the following sections.

4.2. The Effectiveness of UPSP Algorithm

Figure 4 and Figure 5 are position RMSE of IMM, UPSP based Novel-IMM and standard Novel-IMM based on trajectory a and trajectory b, respectively, where the position RMSE is derived by following equation:

$$\mathrm{RMSE}_k = \sqrt{\frac{1}{\mathrm{M}}\sum_{i=1}^{M}[(\hat{x}^i(k)-x^i(k))^2+(\hat{y}^i(k)-y^i(k))^2]} \qquad (29)$$

in which $\hat{x}^i(k)$ and $\hat{y}^i(k)$ are estimated positions in x and y dimensions at the discrete time k in the ith Monte Carlo run, respectively, while $x^i(k)$ and $y^i(k)$ denote true positions in x and y dimensions, and M is the total number of independent Monte Carlo runs.

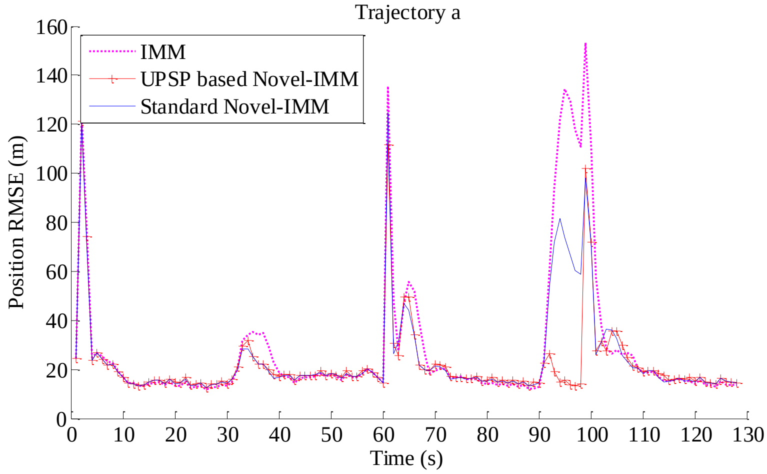

Figure 4. Position RMSE based on Trajectory a

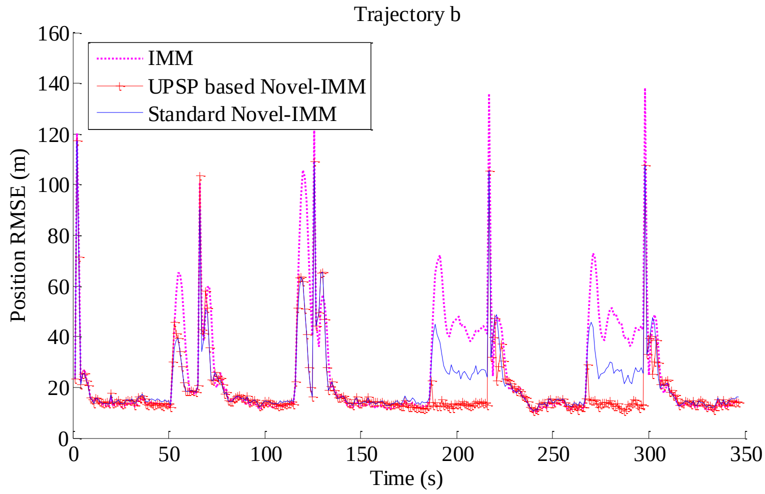

Figure 5. Position RMSE based on Trajectory b

As we can see from Figure 4, the performances of standard Novel-IMM and UPSP based Novel-IMM are similar from 1s to 90s when the target moves in a CV or CA motion. However, when the target maneuvers to turn with a nearly constant turn rate, the standard Novel-IMM can't track the target correctly and an saltation occurs from 91s to 99s, since this algorithm fails to shift from current model set to correct model set, while the UPSP based Novel-IMM doesn't encounter this problem. Then, both standard Novel-IMM and UPSP based Novel-IMM fluctuate sharply in position RMSE at 100s when the target restarts a CV motion, and the two algorithms converge towards the true target state gradually. Besides, the standard Novel-IMM performs better than IMM when target motion changes because model set M_1 could cover more situations compared with model set used in IMM. The above analyses could also explain performances of the three algorithms based on trajectory b in Figure 5.

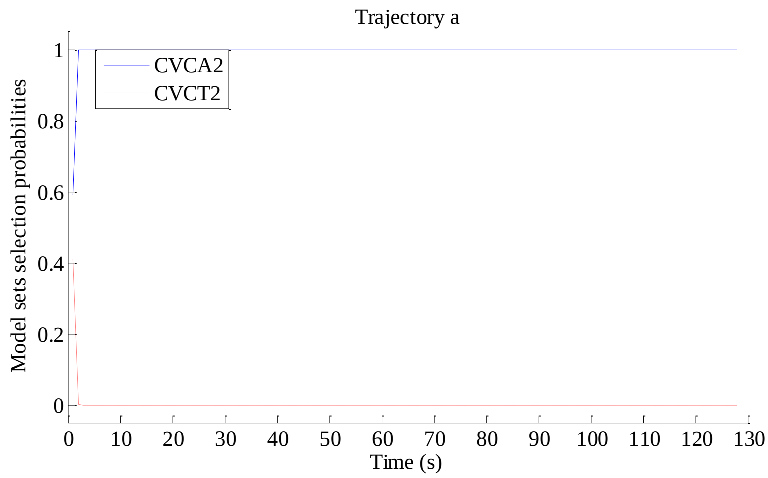

Figure 6.1. Standard Novel-IMM

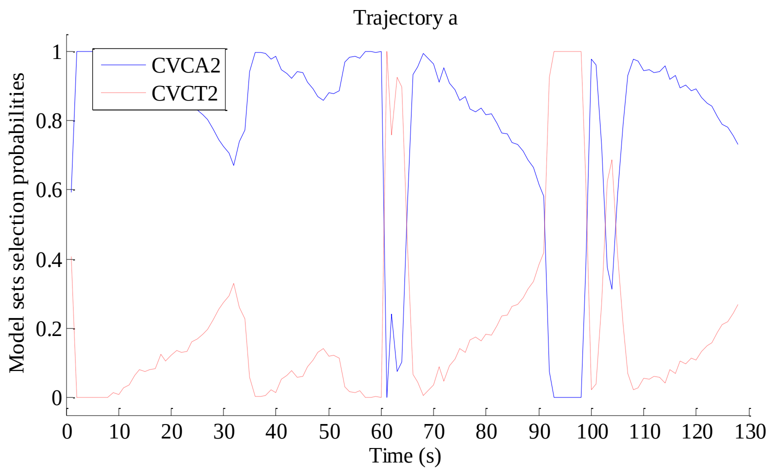

Figure 6.2. UPSP based Novel-IMM

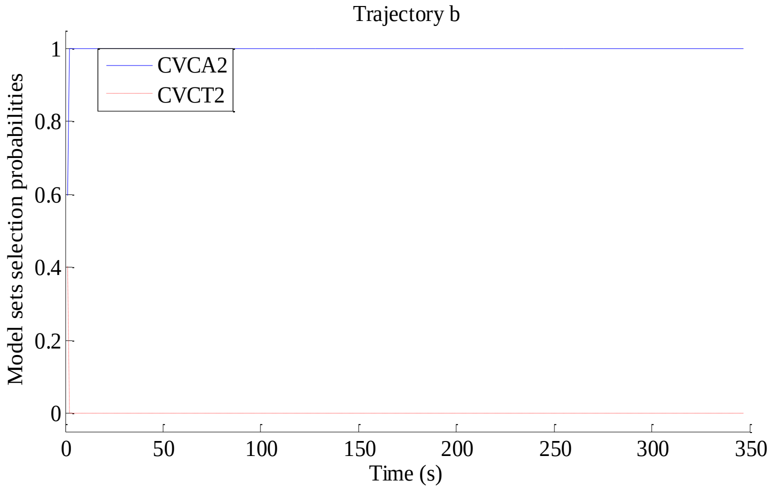

Figure 7.1. Standard Novel-IMM

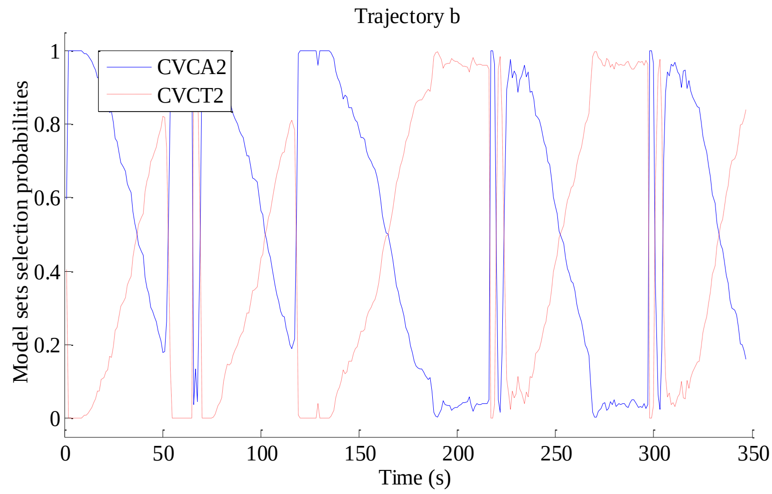

Figure 7.2. UPSP based Novel-IMM

Figure 6.1 and Figure 6.2 reveal the model sets selection probabilities of standard Novel-IMM and UPSP based Novel-IMM based on trajectory a, while Figure 7.1 and Figure 7.2 are selection probabilities based on trajectory b.

From Figure 6.1 we can see that in standard Novel-IMM, the model set M_2 "dead" at the beginning of the target motion, because the target starts the trajectory with a CV motion, while model set M_1 is consist of a CV model and two CA model that describes the target motion more accurately, so the likelihood function value of model set M_2 is extremely small compared with model set M_1, which leads to an underflow problem as analyzed in Section 2.2. Thus, as we can see, the selection probability value of model set M_1 equals to 1 in the rest moments, and this is why a saltation occurred from 91s to 99s when this model set doesn't match the target motion in Figure 4.

On the contrary, as is shown in Figure 6.2, the UPSP based Novel-IMM avoids this predicament. At the beginning, model set M_1 also has a large selection probability nearly equals to 1, but the UPSP algorithm prevents the selection probability of model set M_2 from smaller than 0.05, which keeps model set M_2 "alive". Besides, since there is a CV model in model set M_2, the likelihood function value of model set M_2 becomes larger and larger as time goes by. Consequently, the selection probability of model set M_2 becomes lager and lager gradually. Then, the target moves with a CA motion from 31s to 60s, and the selection probability of model set M_2 turns to small immediately. When the target maneuvers to turn with a nearly constant turn rate from 91s to 99s, M_2 is the correct model set to describe the target's motion, this is why the selection probability of model set M_2 is much larger than that of model set M_1 in Figure 6.2. The same conclusions can be derive from Figure 7.1 and Figure 7.2 based on trajectory b too.

This section demonstrates the effectiveness of UPSP algorithm proposed in Section 2.3.

4.3. Performances of FAIMM Algorithm

In this section, we compare performances of UPSP based Novel-IMM and FAIMM algorithms, while they both employ the UPSP algorithm with an underflow threshold $T_L = 0.05$.

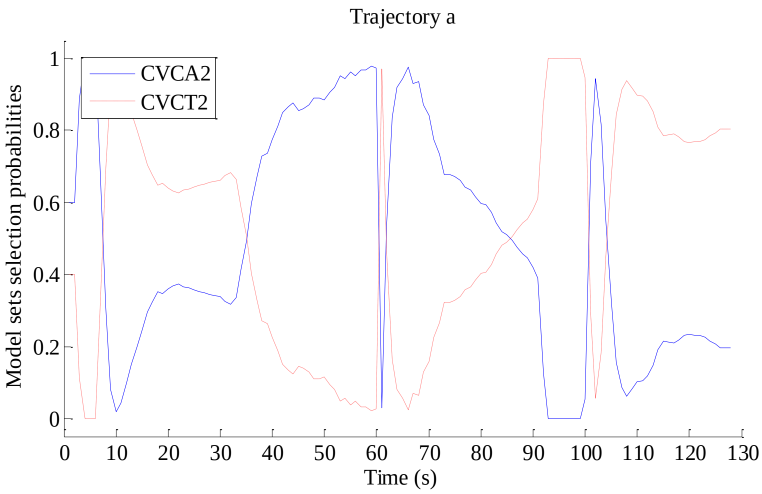

Figure 8.1. FAIMM

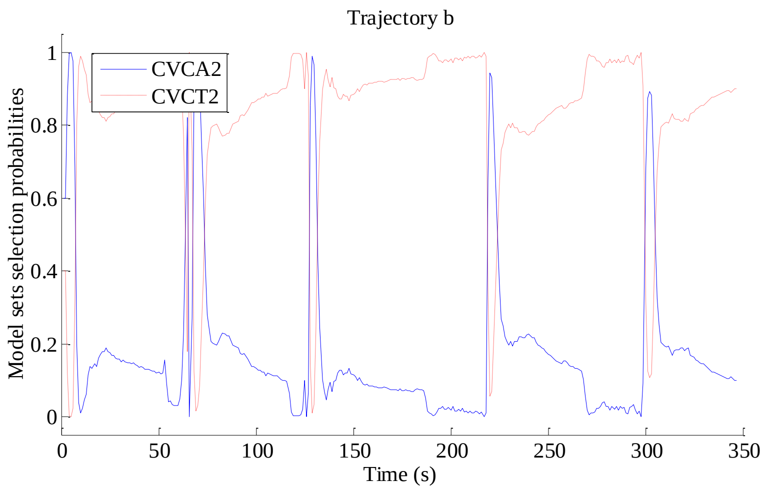

Figure 8.2. FAIMM

The selection probabilities of FAIMM algorithm shown in Figure 8.1 and Figure 8.2 also reflect the effectiveness of UPSP algorithm used in them, which makes every model set “alive” in these algorithms.

Table 1 reveals the position RMSE and executing time of the four algorithms.

Table 1. Comparison on Average Position RMSE and Executing Time

	Average Position RMSE (m)		Executing Time (s)	
	Trajectory a	Trajectory b	Trajectory a	Trajectory b
IMM	28.9936	27.2022	4.1651	10.7420
Standard Novel-IMM	25.0263	22.3945	11. 1347	29.4908
UPSP based Novel-IMM	22.3625	19.4591	11. 2162	29.5806
FAIMM	23.6229	20.2737	6.4631	16.9703

It can be seen from Table 1 that all of the four algorithms in trajectory a outperform those in trajectory b. Because the target doesn’t stay in any motion more than 30 second in trajectory a, however, it takes at least more than 10 second for these five algorithms to converge into the steady state, which is very harmful to average position RMSE. This problem is especially obvious to FAIMM algorithm, since it need more time to reach the steady state. Furthermore, as we can see from Figure 8, although the FAIMM algorithm spends more time to converge, it has higher tracking accuracy when the motion of target changes. Besides, the FAIMM algorithm even perform slightly better than UPSP based Novel-IMM when the target remains in a motion for enough time, while it decreases the computational burden about 40%.

Note: The IMM algorithm shown in Table 1 consists of two models, which lead to great superiority in executing speed. When the IMM algorithm consists of three models, it has similar executing time compared with FAIMM algorithm.

4.4. Threshold Choosing Principle of UPSP Algorithm

Table 2 shows the average position RMSE of UPSP based Novel-IMMand FAIMM algorithms based on trajectory a with different threshold value T_L.

Table 2. Average Position RMSE with Different Threshold Values

	$T_L = 0.001$	$T_L = 0.05$	$T_L = 0.1$	$T_L = 0.15$	$T_L = 0.2$
UPSP based Novel-IMM	22.6270	22.3625	22.4461	22.4924	22.4938
FAIMM	24.6791	23.6229	23.6775	23.7353	23.7940

As we can see from Table 2, the two algorithms perform better when T_L ranges from 0.05 to 0.1, based on which we get following principles to choose most appropriate underflow threshold T_L in UPSP algorithm: On the one hand, T_L should be large enough to activate corresponding model set as the correct one instantly when this model set match the system well. On the other hand, T_L should be small enough so that the selection probabilities of unsuitable model sets would not be close to or even surpass the selection probability of the correct model set subsequently. Thus, T_L should be in an appropriate range [$T_{L\min}$, $T_{L\max}$], so that the proposed algorithm could make the most intelligent choice. Although not flexible, the value of $T_{L\min}$ and $T_{L\max}$ can be derived by experiments or simulations based on specific applications, and any theoretical derivation in mathematics to get optimal value of $T_{L\min}$ and $T_{L\max}$ or achieve better performance in solving model sets selection probabilities underflow problem will be a breakthrough on this methodology.

5. Conclusion

In this paper, a UPSP algorithm is proposed to solve the underflow problem in standard Novel-IMM, and we discussed the principle to set appropriate threshold for UPSP algorithm in order to get ideal output. Based on UPSP algorithm, the FAIMM algorithm that employs steady state Kalman filters is presented, which decreases approximately 40% executing time while keeping similar tracking accuracy compared with UPSP based Novel-IMM. So the proposed FAIMM algorithm is a considerable alternatives in real-time application compared with standard Novel-IMM.

Acknowledgment

This work was supported by the Research Fund of Shandong Jiaotong University (No. Z 201306, Z201419), Science and Technology Plan Project of Shandong Province (2013YD 01042,2014GGX101015) and Natural Science Foundation of Shandong Province (ZR2014 FL006).

References

[1] D. Magill, "Optimal Adaptive Estimation of Sampled Stochastic Processes", IEEE Transactions on Automatic Control, vol. 10, **(1965)**, pp. 434-439.
[2] X.R. Li, "Engineer's guide to variable-structure multiple-model estimation for tracking", Multitarget-multisensor tracking: Applications and advances. vol. 3, **(2000)**, pp. 499-567.
[3] X.R. Li and V.P. Jilkov, "Survey of maneuvering target tracking—Part V Multiple-model methods", IEEE Transactions on Aerospace and Electronic Systems, vol. 41, **(2005)**, pp.1255-1321**.**
[4] H.A.P. Blom, and Y. Bar-Shalom, "The interacting multiple model algorithm for systems with Markovian swiching coefficients", IEEE Transactions on Automatic Control, vol. 33, no. 8, **(1988)**, pp. 780-783.
[5] D. Svensson and L. Svensson, "A new multiple model filter with switch time conditions", IEEE Transactions on Signal Processing, vol. 58, no. 1, **(2010)**, pp.11-25.
[6] X. Fu, Y. Jia, J. Du and F. Yu, "New interacting multiple model algorithm for the tracking of the manoeuvring target", IET Control Theory Applications, vol. 4, no. 10, **(2009)**, pp. 2184-2194.
[7] C.E. Seah and I. Hwang, "Algorithm for performance analysis of the IMM algorithm", IEEE Transactions on Aerospace and Electronic Systems, vol. 47, no. 2, **(2011)**, pp.1114-1124**.**
[8] C. E. Seah and I. Hwang, "State estimation for stochastic linear hybrid system with continuous-state-dependent transitions: An IMM approach", IEEE Transactions on Aerospace and Electronic Systems, vol. 45, no. 1, **(2009)**, pp.376-392.
[9] X.R. Li, Y. Bar-Shalom, "Multiple-model estimation with variable structure", IEEE Transactions on Automatic Control, vol. 41, no. 4, **(1996)**, pp. 478-493.
[10] H.Q. Qu, L.P. Pang and S.H. Li, "A novel interacting multiple model algorithm", Signal Processing, vol. 89, no. 11, **(2009)**, pp. 2171–2177**.**
[11] X.R. Li and Y.M. Zhang, "Numerically robust implementation of multiple-model algorithms", IEEE Transactions on Aerospace and Electronic Systems, vol. 36, no. 1, **(2000)**, pp. 266–278**.**
[12] B. Ekstrand, "Poles and zeros of α-β and α-β-γ tracking filters". IEE Proceedings: Control Theory and Applications, vol. 148, no. 5, **(2001)**, pp. 370-376.
[13] M. Dahmani, M. Keche, A. Ouamri and A. Meche, "A new IMM algorithm using fixed coefficients filters (FastIMM)". AEU - International Journal of Electronics and Communications, vol. 64, no. 12, **(2010)**, pp.1123-1127.
[14] X.R. Li, "Multiple-model estimation with variable structure some theoretical considerations", Proceedings of the 33rd IEEE Conference on Decision and Control, vol. 2, no. 8, **(1994)**, pp. 1199-1204，December 14-16 ;Lake Buena Vista, FL, America.

Authors

Hua Wu, male, lecturer, was born in 1982. He received the Master degree from Shandong University in 2008. He is currently working as a lecturer of Shandong Jiaotong University. His research interests lie in satellite navigation and marine communication, wuhua1982111@126.com.

JianPing Xing (Corresponding author), male, professor, was born in 1969. He received the Ph.D. degree from Beihang University in 2009. He is currently in charge of electric and electronic experiment center as vice dean, Shandong University. His research interests lie in signal processing, wireless sensor network, ITS network information transmission and satellite navigation.

Yang Liu, male, born in March 1987, received his ME degree in Communication & Information System from Shandong University in 2012. He is currently a teaching assistant in Shandong Jiaotong University. His research interests include wireless sensor networks, localization algorithms and intelligent transportation systems.

