Maneuvering Star-Convex Extended Target Tracking Based on Modified Expected-Mode Augmentation Algorithm

Jinjin Zhang[1] , Lifan Sun[1,2],* , Dan Gao[1]

1.Henan University of Science and Technology – School of Information Engineering – Luoyang/Honā – China. **2.**Longmen Laboratory – Luoyang – China.

*Correspondence author: lifan.sun@gmail.com

ABSTRACT

In utilizing a variable-structure multiple-model (VSMM) algorithm for kinematic state estimation, the core step is the model set design. This study aims to refine the existing expected-mode augmentation (EMA) algorithm, a method of model set design. First, the OTSU algorithm is employed to determine an adaptive threshold, which in turn allows for a reasonable partition of the basic model set. Next, a subset of possible models is preserved, reactivating models adjacent to the one with the highest prediction probability, eliminating improbable models, and yielding an augmented expected mode. Additionally, the study leverages the translation properties of radial functions and inverse trigonometric function formulas to derive a maneuvering model for star-convex extended targets under uniformly accelerated conditions. In order to assess the effectiveness of the proposed algorithm and the validity of the established maneuvering model, simulation experiments were carried out in both fixed and random scenarios. The proposed algorithm demonstrates improved performance when compared to the interactive multiple-model algorithm and the unmodified EMA algorithm.

Keywords: Extended target tracking; Variable-structure multiple-model; OTSU.

INTRODUCTION

This study addresses the tracking problem of extended targets in the context of radar sensors. Extended target tracking can be characterized as the joint estimation of kinematic and extension states. The foundation of any tracking algorithm lies in the tracking models, making the selection of an appropriate model a crucial starting point for subsequent tracking. This paper primarily concentrates on the development of motion models.

Existing single motion models encompass constant acceleration (CA), constant velocity (CV) (Yunita *et al.* 2020), constant turn (CT) (Amrouche *et al.* 2018), Singer model (Zhu *et al.* 2019), Jerk model (Deng *et al.* 2019), and others. Nonetheless, a single model is often insufficient to describe the complex and variable kinematic behavior of targets in maneuvering scenarios. When a sudden, intense maneuver occurs, the tracking accuracy may significantly diminish or even result in tracking failure. Subsequently, the interactive multiple-model (IMM) algorithm was proposed for the first time (Blom 1984). This method

employs several fixed models for parallel estimation at each instance, while concurrently introducing interaction among these models (Qu *et al.* 2009). Ultimately, the estimation results of each model are fused based on updated model probabilities. However, this approach can only achieve optimal estimation when the model set corresponds to the true pattern. In an attempt to maximize the approximation to the target's true motion mode, it is common to increase the number of models. Nevertheless, this not only escalates the computational burden but may also lead to reduced tracking accuracy due to competition between models (Shen-tu *et al.* 2018).

To overcome this predicament, there are two potential solutions: either develop a superior model set or employ a variable model set. Li and Bar-Shalom (1996) initially introduced a variable-structure multiple-model (VSMM) algorithm. In comparison to the fixed-structure multi-model (FSMM) algorithm, the VSMM algorithm includes model set design, which can also be perceived as a model-set adaptation (MSA) module. This module can determine which models from the entire set will be chosen to carry out subsequent multiple-model (MM) estimation operations at each instance. As a result, a smaller number of models can achieve a higher degree of approximation to the true system mode. The one-time recursive process of the VSMM algorithm accomplishes two primary tasks, which can be summarized as follows:

- MSA: Tasked with utilizing prior and posterior information to determine the model set at each instance, primarily focusing on activating new models and terminating a subset of the current model set (Lan and Li 2013). The objective of activating a new model is to identify a "superior" model, which is incorporated into the existing model set and then utilized collectively for state estimation. In this context, "superior" implies that the newly activated model can more accurately describe the true system mode. Conversely, the goal of terminating a model is to eliminate models that significantly deviate from the system mode; thus, reducing computational demands while avoiding unnecessary competition between models.

- VSIMM (Shen *et al.* 2021): Responsible for estimating the target state using the IMM approach under the condition of a given model set.

The distinction between various VSMM algorithms lies in the differing model-set adaptation strategies. The expected-mode augmentation (EMA) algorithm is classified as a model activation method (Li *et al.* 2005). While maintaining the basic model set, it extends an expected mode, which is the probability-weighted sum of the basic model. Figure 1 presents valid model topology diagrams for 7, 9, and 13 basic models, with different coordinate positions corresponding to varying levels of maneuver. The tracking performance of the EMA algorithm is significantly superior to that of the IMM algorithm. However, its basic model set is predefined and unchangeable, leading to limited adaptability of the augmented expected mode to the basic model set. Ideally, the model should adapt in real time based on the potential kinematic behavior of the target.
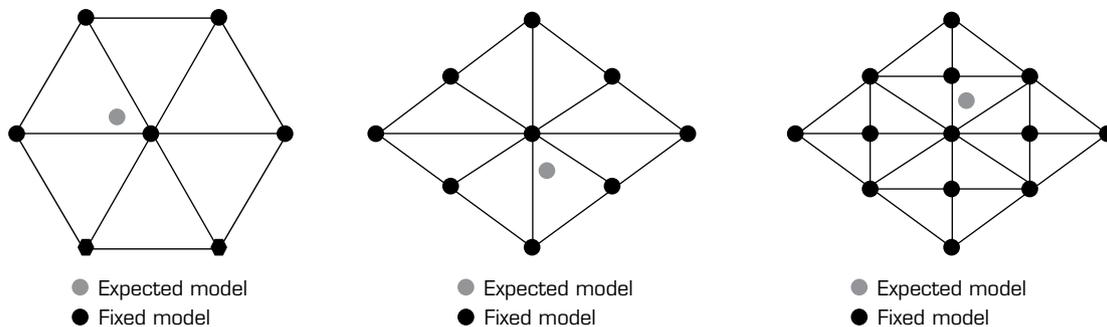
Source: Adapted from Li *et al.* (2005).

Figure 1. The model topological structure diagram of EMA algorithm.

The likely-model set (LMS) algorithm (Li and Zhang 2000) divides the original model set into unlikely model sets M_{k-1}^L, general model sets M_{k-1}^M, and important model sets M_{k-1}^H by setting threshold values based on the model probability when determining the model set M_k at k. Subsequently, it discards the unlikely models with low probability while retaining the general and crucial models. Besides, if there are important models $\left(m^{(i)} \mid m^{(i)} \in M_{k-1}^H\right)$, activate the model set $A_{m^{(i)}}$ adjacent to them to obtain the changed model group M_k. The adaptive strategy of LMS can be represented by the Eq. 1:

$$M_k = \left(M_{k-1} - M_{k-1}^L \right) \bigcup \left(\bigcup_{m^{(i)} \in M_{k-1}^H} A_{m^{(i)}} \right) \tag{1}$$

where M_{k-1} represents the valid set of models at k-1.

The LMS algorithm can maintain estimation accuracy while reducing a certain amount of computation. However, it is unable to activate models outside the comprehensive model set, leading to higher peak errors during target maneuvers. Moreover, fixed thresholds are employed when determining different subsets of models.

The fixed threshold method demonstrates effective clustering for model sets with substantial probability differences among various models, facilitating the division of each model subset. Conversely, when probabilities between models are closely related, assessing their relative importance becomes challenging. The OTSU algorithm incorporates clustering concepts, enabling the identification of an appropriate numerical level through variance calculations to bifurcate the elements within the entire numerical set, maximizing the overall numerical value difference between the two parts and minimizing the difference within each part (Otsu 1979; Qian *et al.* 2021; Xiao *et al.* 2022; Zhao 2022). Consequently, the OTSU algorithm can be utilized for clustering numeric sets.

Inspired by the design idea of the LMS algorithm, this study introduces a novel model set design method based on EMA, ingeniously employing OTSU. Initially, the OTSU algorithm is utilized to automatically select a threshold value at any given time, enabling the binarization of the model prediction probability set. Then, the minimum value in the large category is taken as the threshold value for "retaining models" and "eliminating models" to obtain two different subsets of models (likely subset $M_l = \bigcup_{i=1}^{p} \left(m_{k-1}^i \mid \mu_{k|k-1}^i \geq \tau \right)$ and unlikely subset $M_u = \bigcup_{i=1}^{q} \left(m_{k-1}^i \mid \mu_{k|k-1}^i < \tau \right)$). Meanwhile, the model adjacent to the most likely model $m_{k-1}^s \mid \mu_{k|k-1}^s = \max \left(\mu_{k|k-1} \right)$ is reactivated. Thus, the basic model set at k can be determined.

Finally, an augmented model is acquired based on the prediction probability and parameters of the preserved basic model. Additionally, a maneuvering model for star-convex extended targets under uniformly accelerated scenarios is derived, enabling the joint estimation of kinematic and extension states. In order to validate the effectiveness of the proposed algorithm, simulation experiments were performed in both deterministic and random scenarios. When compared to the fixed IMM and original EMA algorithm, the proposed algorithm demonstrates a notable enhancement in tracking speed and accuracy.

PROBLEM FORMULATION

State equation

The transition process between states can be modeled as a simple stochastic hybrid system, i.e., a linear time-varying system (Eq. 2).

$$\begin{bmatrix} x_k^m \\ x_k^e \end{bmatrix} = \begin{bmatrix} F_k^m & 0 \\ 0 & F_k^e \end{bmatrix} \begin{bmatrix} x_{k-1}^m \\ x_{k-1}^e \end{bmatrix} + \begin{bmatrix} w_k^m \\ w_k^e \end{bmatrix}, k \in \mathbb{N} \tag{2}$$

where x_k^m and x_k^e denote kinematic and morphological parameter vectors, F_k^m and F_k^e are corresponding state transition matrices. w_k^m and w_k^e are uncorrelated process noise sequences.

Measurement equation

The measurement generation process of star-convex extended targets includes the establishment of measurement source model and sensor model, as shown in Fig. 2. For an extended target, since measurement data sources may also be distributed inside the boundary, scaling the boundary can be selected to cover its internal situation (Zhang *et al.* 2022). For an unknown measurement source data $y_k^{(l)}$, if a matching scaling factor $\hat{S}_k$ ($\hat{S}_k \in [0,1]$) is known, its location can be defined by Eq. 3:

$$y_k^{(l)} \in x_k^c + \hat{s}_k(S(x_k) - x_k^c) \tag{3}$$

where x_k^c denotes the centroid position, and $S(x_k)$ represents the contour point.
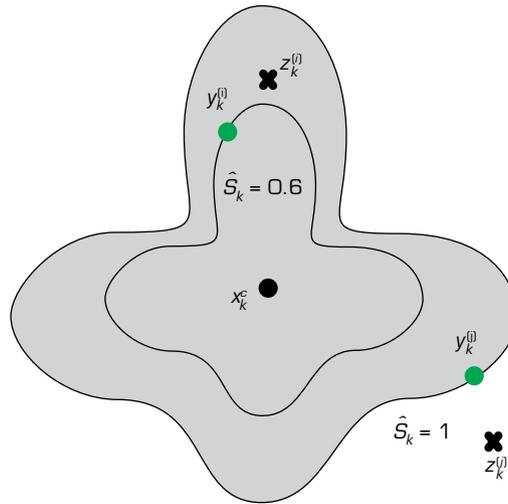
Source: Elaborated by the authors.

Figure 2. Measurement model for extended targets: Independent generation of measurements.

Furthermore, for a given single measurement data source $y_k^{(l)}$, the sensor model defines the measurement location by Eq. 4:

$$z_k^{(l)} = y_k^{(l)} + v_k \tag{4}$$

where $z_k^{(l)}$ is the l_{th} measurement data, and v_k represents Gaussian white noise with a zero-mean value.

The measurement generation process of the extended target is shown in Fig. 3, which can be summarized as an extent source model and a sensor model.
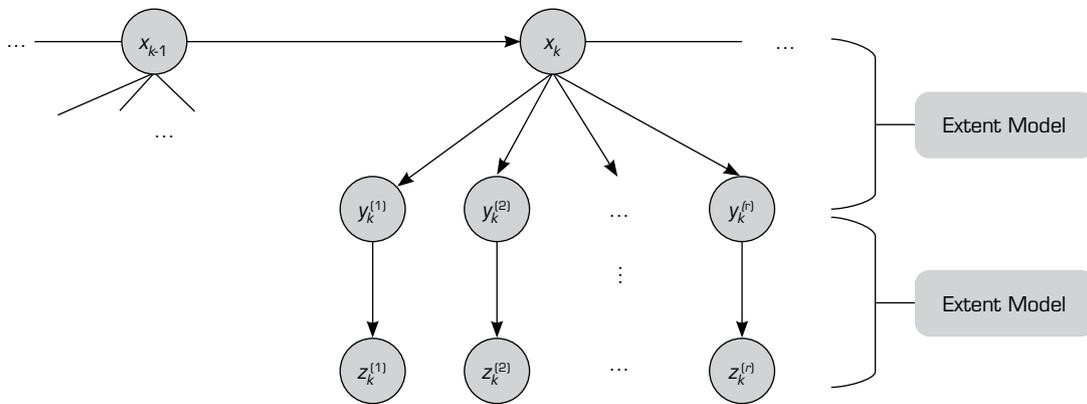
Source: Retrieved from Baum (2013).

Figure 3. Measurement generation process for extended targets (visualization).

Model jump equation

The jump process between models can be considered as a first order Markov chain, and the transition probability can be defined by Eq. 5:

$$P\{s_k = m_j | s_{k-1} = m_i\} = \pi_{ij}, \forall m_i, m_j \in M \tag{5}$$

where the set M contains all the possible motion models of the system, π_{ij} represents the transition probability from state m_i to m_j. Figure 4 shows the structural diagram of the transition between states in a Markov chain.
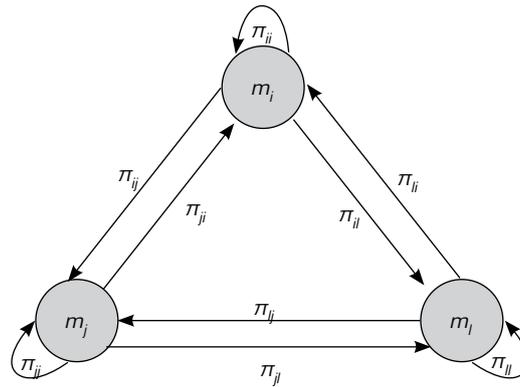
Source: Elaborated by the authors.

Figure 4. State transition diagram of Markov chain.

THE PROPOSED MEMA ALGORITHM

S1. Model set partitioning: Divide the basic model set M_{k-1}^b at k-1 into likely model subset M_l ($M_l = \bigcup_{i=1}^{p} (m_{k-1}^i \mid \mu_{k|k-1}^i \geq \tau)$) and unlikely model subset M_u ($M_u = \bigcup_{i=1}^{q} (m_{k-1}^i \mid \mu_{k|k-1}^i < \tau)$). Here, the threshold value τ can be adaptive according to the OTSU algorithm, which uses variance calculations to find an appropriate numerical level τ to divide the elements of the entire set (M_{k-1}^b; m_1, m_2, ..., m_r) into two subsets M_l and M_u. The partitioning process maximizes the difference between the overall values (i.e., model prediction probability $\mu_{k|k-1}^i$) in the two subsets, and minimizes the difference between the values in each subset. The specific division process is as the following:

Assuming that the number of models in the basic model set M_{k-1}^b is r, the proportion of the number of models in M_l to the total number is λ_1, and its numerical average value is ρ_1; the proportion of the number of models in M_u to the total number is λ_0, and its numerical average is ρ_0; t he inter class variance is defined as g. The relationship among λ_1, λ_2, ρ_1, ρ_0 and g can be jointly defined by Eqs. 6, 7 and 8:

$$\lambda_1 = \frac{p}{r}, \quad \lambda_o = \frac{q}{r}, \quad \lambda_1 + \lambda_0 = 1 \tag{6}$$

$$\lambda_1 \rho_1 + \lambda_0 \rho_0 = 1 \tag{7}$$

$$g = \lambda_1 \lambda_0 * (\lambda_1 - \lambda_0)^2 \tag{8}$$

Then, using the traversal method to obtain a threshold value τ that maximizes the inter class variance.

The reasons why using OTSU algorithm for model set partitioning can be summarized as follows:

- The calculation is simple, and the model set can also be effectively divided when the numerical values of the model probabilities in the set do not differ significantly;
- Compared to the method of using a fixed threshold for partitioning, it can achieve adaptation based on the distribution of elements in the set, making the partitioning process more reasonable.

S2. Model-set adaptation: Calculate M_k according to $M_k = M_k^b \cup E_k$, where M_k defines the set of all possible models at k, M_k^b contains models in the basic model set, and E_k is the augmented expected mode. Specifically, it contains the following four steps:

- Retention of likely models: Retain all models in the subset M_l.
- Activation of a new model: Activates the model set M_a adjacent to the most likely model ($m_{k-1}^s \mid \mu_{k|k-1}^s = \max (\mu_{k|k-1})$).
- Elimination of unlikely models: Delete all models that are not activated in the unlikely model set M_u, then the basic model set at k is $M_k^b = M_{k-1}^b - (M_u - M_u \cap M_a)$.
- Augmentation of expected mode: Use the newly generated basic model set M_k^b and its prediction probability $\mu_{k|k-1}^i$ to obtain an augmented expected mode $E_k = E(M_k^b; m_1, m_2, ... m_e)$. where e is the number of models in the basic model set M_k^b.

The entire process of the proposed MEMA algorithm can be briefly described in Table 1.

Table 1. The proposed MEMA algorithm.

Step	Title	Description
1	Initialization	a) Design total model set (M_{k-1}); b) Design basic model set (M_{k-1}^b);
2	Model set partitioning	a) Calculate threshold τ using OTSU; b) Likely models, $M_l = \cup_{i=1}^{p} \{m_{k-1}^i \| \mu_{k\|k-1}^i \geq \tau\}$; c) Unlikely models, $M_u = \cup_{i=1}^{q} \{m_{k-1}^i \| \mu_{k\|k-1}^i < \tau\}$;
3	Model-set adaption	a) Retain all the models in M_l; b) Activate the models M_a adjacent to the most likely model; c) Delete the inactive model in M_u; d) Obtain the basic model set according to $M_k^b = M_{k-1}^b - (M_u - M_u \cap M_a)$ and calculate E_k; e) Determine the current total model set according to $M_k = M_k^b \cup E_k$;
4	Model-set sequence conditioned estimation	Run VSIMM;
5	Send output	$\{\hat{x}_i(k\|k), P_i(k\|k), \mu_i(k)\}\ m_i \in M_k$

Source: Elaborated by the authors.

A MANEUVERING MODEL FOR STAR-CONVEX EXTENDED TARGET UNDER UNIFORM ACCELERATION SCENARIO

The state of the extended target includes dynamic and morphological variables, and there is a coupling relationship between them (Baum 2013; Liu *et al.* 2022). For example, when a target generates maneuvers such as constant acceleration, it is necessary to ensure that the orientation of shape is always consistent with the direction of movement. Therefore, how to establish a concise maneuver model to simultaneously describe the dynamic evolution process of two variables is a key issue to be solved in extended target tracking. In this paper, using the translation characteristics of radial function and the inverse trigonometric function formula, the morphological evolution equation of a star-convex maneuvering extended target in a uniform acceleration scenario is derived. Furthermore, combined with the dynamic equation of kinematic state, a complete maneuvering model is obtained. The content of this chapter is the preparatory work for the subsequent simulation.

As shown in Fig. 5, the contour of a star-convex extended target can be described using a radial function in the two-dimensional coordinate system.
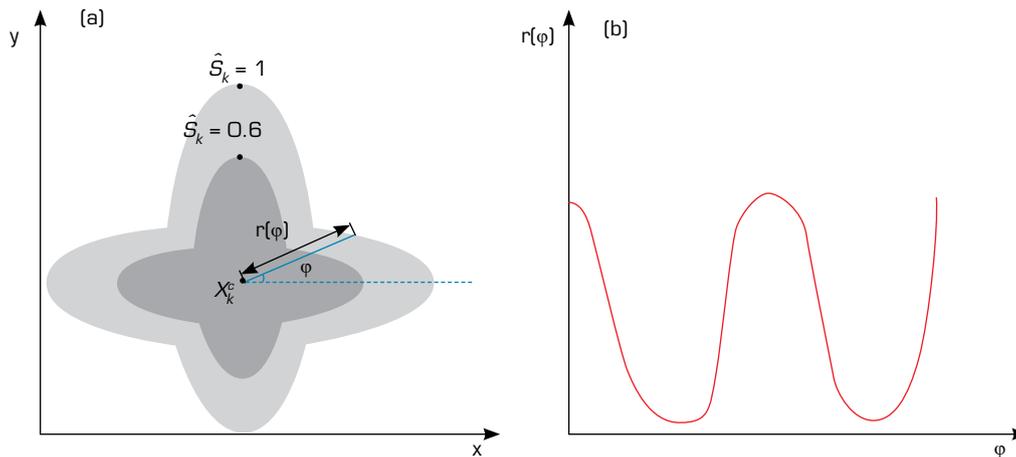
Source: Elaborated by the authors.

Figure 5. Representing a star-convex shape with the radial function. (a) Star-convex target; (b) Radial function.

The horizontal axis ranges from 0 to 2π, and the size of the vertical axis defines the distance between each contour point and the centroid x_k^c. The contour $\bar{S}(x_k)$ after zooming can be described as follows (Eqs. 9 and 10) (Baum and Hanebeck 2014):

$$\bar{S}(x_k) = \{\hat{s}_k r(\varphi)e(\varphi) + x_k^c | \varphi \in [0,2\pi), \hat{s}_k \in [0,1]\} \tag{9}$$

$$e(\varphi) = \begin{bmatrix} \cos(\varphi) \\ \sin(\varphi) \end{bmatrix} \tag{10}$$

Perform a linearization process on the radial function, i.e., Fourier series expansion (Eqs. 11–13),

$$r(B_k, \varphi) = a_k^{(0)} + \sum_{j=1,\cdots,N^F} \left(a_k^{(j)} \cos(j\varphi) + b_k^{(j)} \sin(j\varphi) \right) = R(\varphi) B_k \tag{11}$$

$$R(\varphi) = \left[1, \cos(\varphi), \sin(\varphi), \dots, \cos(N^F\varphi), \sin(N^F\varphi) \right] \tag{12}$$

$$B_k = \left[a_k^{(0)}, a_k^{(1)}, b_k^{(1)}, \dots, a_k^{(N^F)}, b_k^{(N^F)} \right]^T \tag{13}$$

where N^F represents the order of Fourier expansion. The higher the order, the more detailed morphological information can be reflected by its coefficients. The extension parameter vector x_k^e can be represented by B_k, Eq. 14:

$$x_k^e = \left[a_k^{(0)}, a_k^{(1)}, b_k^{(1)}, \dots, a_k^{(N^F)}, b_k^{(N^F)} \right]^T \tag{14}$$

The radial function has a translational characteristic. When the target rotates angle θ, the radial function also shifts the corresponding angle in the horizontal direction. According to this property, the radial function after rotation can be obtained through translation calculation based on the radial function at the previous moment, Eq. 15:

$$r(x_k^e, \varphi) = r(x_{k-1}^e, \varphi - \theta) + w_{k-1}^e \tag{15}$$

Literature (Sun *et al*. 2021) has derived the maneuvering model of a star-convex extended target in a turning scenario, Eqs. 16–19:

$$\begin{bmatrix} x_k^m \\ x_k^e \end{bmatrix} = \begin{bmatrix} F_k^m(\theta) & 0 \\ 0 & F_k^e(\theta) \end{bmatrix} \begin{bmatrix} x_{k-1}^m \\ x_{k-1}^e \end{bmatrix} + \begin{bmatrix} w_k^m \\ w_k^e \end{bmatrix}, k \in \mathrm{N} \tag{16}$$

$$F_k^m(\theta) = \begin{bmatrix} 1 & \frac{\sin\theta}{\omega} & 0 & \frac{\cos\theta}{w} \\ 0 & \cos\theta & 0 & -\sin\theta \\ 0 & \frac{1-\cos\theta}{\omega} & 1 & \frac{\sin\theta}{\omega} \\ 0 & \sin\theta & 0 & \cos\theta \end{bmatrix} \tag{17}$$

$$F_k^e(\theta) = diag\left(1, F^{e,1}, \cdots, F^{e,N_F}\right) \tag{18}$$

$$F^{e,j} = \begin{bmatrix} \cos(j\theta) & -\sin(j\theta) \\ \sin(j\theta) & \cos(j\theta) \end{bmatrix}, j = 1, \cdots N_F \tag{19}$$

If a target undergoes a motion of constant acceleration, its kinematic state can be defined as $x_k^m = (x_k, v_{k(x)}, a_{k(x)}, y_k, v_{k(y)}, a_{k(y)})^T$, where $(a_{k(x)}, a_{k(y)})^T$ represents the acceleration vector, $a_{k(x)}$ and $a_{k(y)}$ define the acceleration on the horizontal and vertical axes, respectively. Combined with the inverse trigonometric function formula, it can be deduced that (Eq. 20):

$$\theta = \arccos\left(\frac{v_{k(x)}}{\sqrt{\left(v_{k(x)}\right)^2 + \left(v_{k(y)}\right)^2}}\right) \tag{20}$$

Substituting the above into Eqs. 18 and 19 to obtain the transition matrix of extension parameter, i.e., $F_k^e(a_{k(x)}, a_{k(y)})$. The transition matrix of kinematic parameter is (Eq. 21):

$$F_k^m\left(a_{k(x)}, a_{k(y)}\right) = \begin{bmatrix} 1 & T & \frac{T^2}{2} & 0 & 0 & 0 \\ 0 & 1 & T & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & T & \frac{T^2}{2} \\ 0 & 0 & 0 & 0 & 1 & T \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix} \tag{21}$$

By substituting $F_k^e(a_{k(x)}, a_{k(y)})$ and $F_k^m(a_{k(x)}, a_{k(y)})$ into Eq. 2, the dynamic equation of a star-convex extended target in uniform acceleration motion can be obtained.

SIMULATION RESULTS AND DISCUSSIONS

To test the effectiveness of the proposed MEMA algorithm and validate the established maneuver model, the experiment will be conducted through Monte Carlo simulation under a constant acceleration scenario. The tracking object is a single maneuvering extended target, which does not involve clutter and missed detection issues. Besides, in order to reflect fairness, a deterministic scenario (DS) and a random scenario (RS) are set respectively.

Deterministic scenario (DS)

The simulation was conducted in a constant accelerated scenario, and the algorithms involved in the comparison were the IMM and EMA algorithms. Root-mean-square-error (RMSE) and Hausdorff distance (Marošević 2018) are used to evaluate the estimation accuracy of motion state and shape, respectively. The smaller the value, the higher the accuracy.

The total number of models in the IMM algorithm is 7, and the number of basic model sets both in EMA and MEMA is also uniformly set to 7. Specific parameters can be defined by Eq. 22.

$$\left\{a_1 = (0,0)^T \ a_2 = (20,0)^T \ a_3 = \left(10,10\sqrt{3}\right)^T \ a_4 = \left(-10,10\sqrt{3}\right)^T \ a_5 = (-20,0)^T \ a_6 = \left(-10,-10\sqrt{3}\right)^T \ a_7 = \left(10,-10\sqrt{3}\right)^T\right\} \tag{22}$$

The initial kinematic state of the target is $x_o^m = (1000,0, -10,5000, -200, -10)^T$. Table 2 shows the detailed maneuvering process. The program performs 100 simulation steps each time, during which the acceleration (m/s^2) of the target changes a total of 4 times.

Table 2. Maneuvering process in DS.

Step	[0,20]	[20,40]	[40,60]	[60,80]	[80,100]
Acceleration	(3,3)	(-2,-2)	(3,3)	(-3,-3)	(3,3)

Source: Elaborated by the authors.

Figure 6, Tables 3 and 4 show the overall performance evaluation results of the three algorithms. Compared to the fixed IMM algorithm, EMA and MEMA perform better in both kinematic and extension parameters estimation. This can be attributed to a

time-varying set of models based on the possible motion patterns of the target. In other words, the augmented expected mode can better match the true system mode to some extent. Therefore, even if the target maneuvers, there will not be a significant peak error. Furthermore, compared to EMA, MEMA exhibits higher estimation accuracy and less computational complexity. The reasons may be summarized as follows: On the one hand, the MEMA algorithm sets the basic model set to be variable, eliminating mismatched models at every moment, thus reducing unnecessary competition between models. On the other hand, using variable thresholds for model set partitioning enhances the rationality and fairness of the partitioning process.
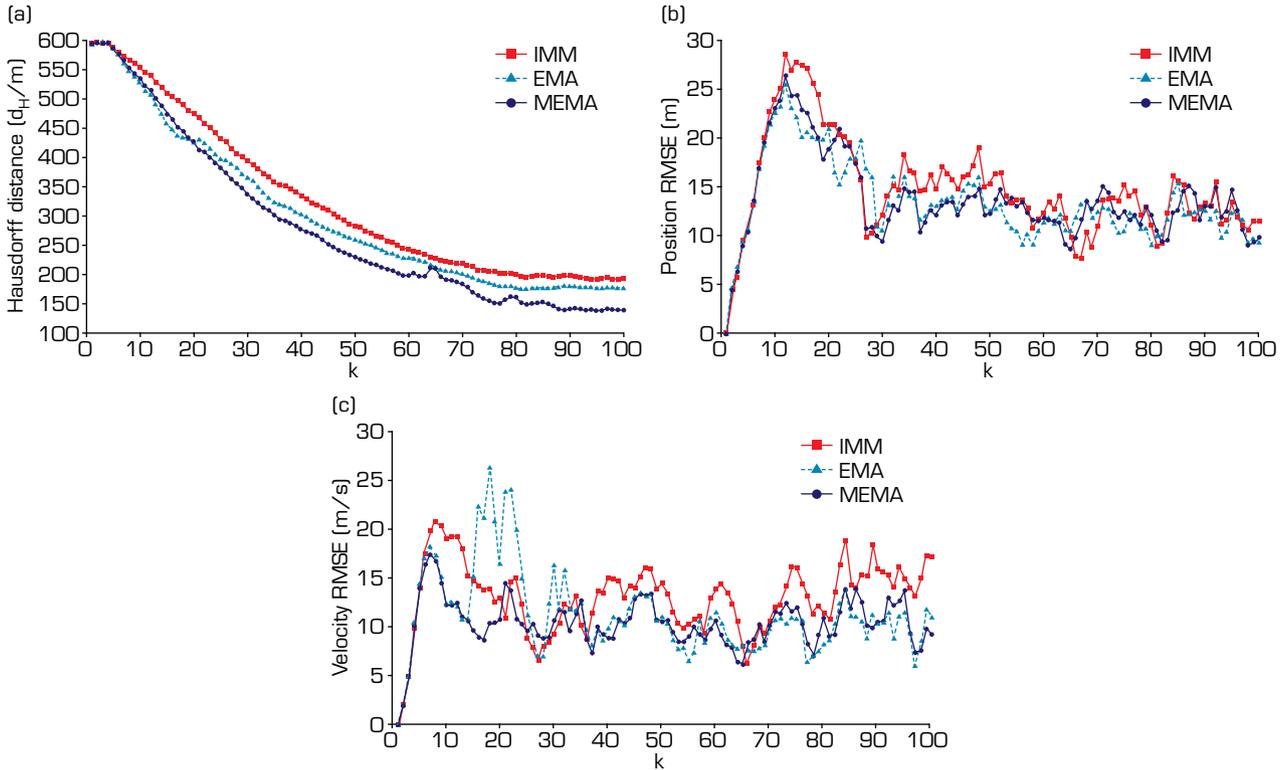
Source: Elaborated by the authors.

Figure 6. Performance comparisons of the three algorithms in DS. (a) Hausdorff distance; (b) Position RMSE; (c) Velocity RMSE.

Table 3. Performance comparisons of the three algorithms in DS.

	Hausdorff distance (d_H/m)	Position RMSE (m)	Velocity RMSE (m/s)
IMM	327.172	14.912	13.040
EMA	303.045	13.570	11.324
MEMA	283.183	13.748	10.375

Source: Elaborated by the authors.

Table 4. Single run time of three algorithms in DS.

Algorithm	IMM	EMA	MEMA
Time (s)	3.139	3.677	3.361

Source: Elaborated by the authors.

Random scenario (RS)

In order to provide a relatively fair testing environment, another simulation scenario is set up. In this scenario, the acceleration of the target in both the horizontal and vertical directions is random, and the time τ_T spent at a certain acceleration is also random. Relevant parameters are set as follows:

The acceleration in the horizontal direction is defined as $a_x = a_{max} * cos\beta$, and that in the numerical direction is $a_y = a_{max} * sin\beta$. where $a_{max} = 10\sqrt{2}$, $\beta \in (0°, 135°)$, five β within the range of this angle during a single program execution will be randomly generated. The simulation program executes 50 steps each time, and the residence time τ_t corresponding to a certain acceleration (a_x, a_y) is a random number that satisfies the condition $\sum_{t=1}^{5} \tau_t = 50$.

Figure 7, Tables 5 and 6 show the performance comparison results of the three algorithms. Hausdorff distance is an indicator used to measure the performance of morphological estimation during extended target tracking. The smaller the value, the closer the estimated result of the corresponding algorithm is to the true shape. Velocity and position RMSE can judge the accuracy of tracking algorithms in estimating motion behavior (including position, velocity, etc.). Similarly, the smaller the value, the better the performance. In this simulation scenario, the maneuvering behavior of the target at each moment is random, which poses a high challenge to the tracking algorithm. Compared to the EMA algorithm, the proposed MEMA algorithm improves tracking accuracy while also reducing a certain amount of computation. Moreover, the performance of the MEMA algorithm in the entire tracking process has been relatively stable compared to the other two algorithms, indicating that the robustness of MEMA is better.
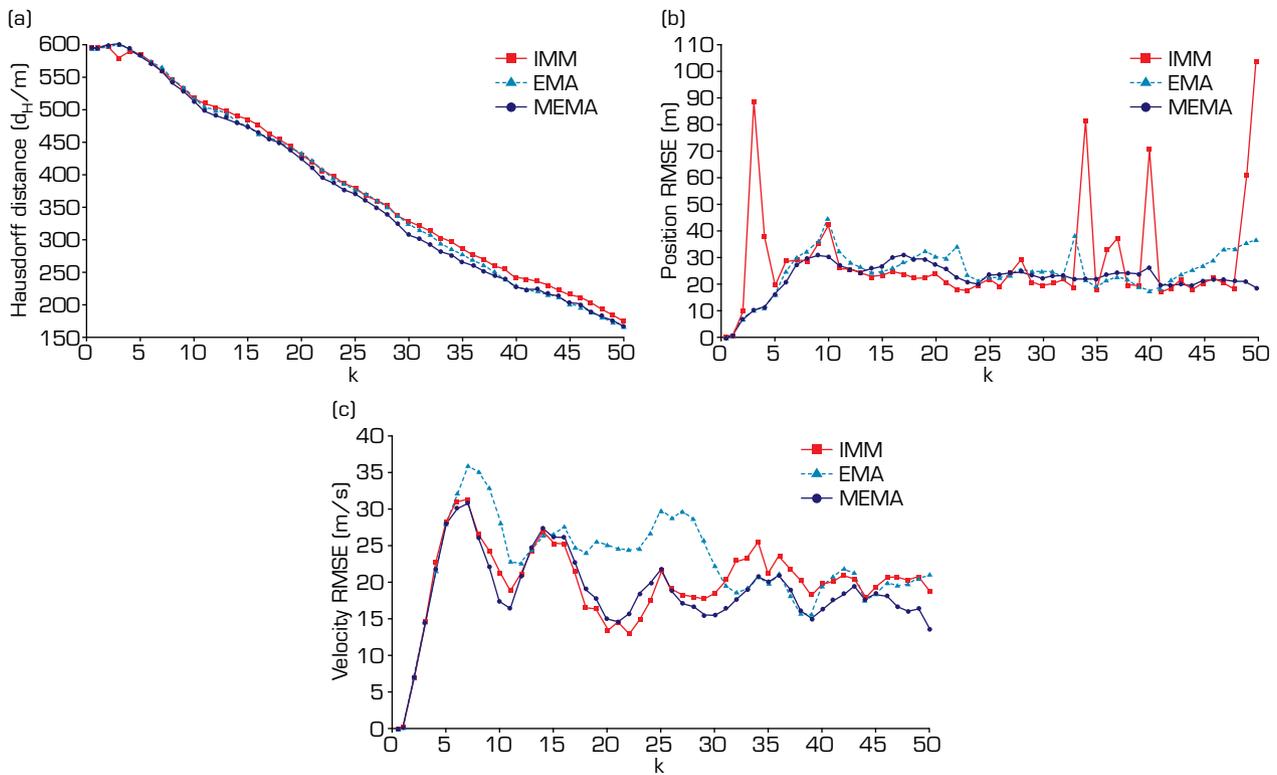
Source: Elaborated by the authors.

Figure 7. Performance comparisons of the three algorithms in RS.
(a) Hausdorff distance; (b) Position RMSE; (c) Velocity RMSE.

Table 5. Performance comparisons of the three algorithms in RS.

	Hausdorff distance (d_H/m)	Position RMSE (m)	Velocity RMSE (m/s)
IMM	381.515	28.999	21.254
EMA	376.107	24.817	23.058
MEMA	371.506	23.965	20.181

Source: Elaborated by the authors.

Table 6. Single run time of three algorithms in RS.

Algorithm	IMM	EMA	MEMA
Time [s]	1.729	1.964	1.931

Source: Elaborated by the authors.

DISCUSSION

In the two simulation scenarios established in this study, the overall performance of the MEMA algorithm surpasses that of the IMM and EMA, attributable to three inherent characteristics of the algorithm:

- Utilizing the OTSU algorithm, a variable threshold value is computed based on the model's prediction probability, enabling a reasonable partitioning of the existing basic model set;
- Activation and termination strategies for the model are provided, eliminating some implausible models and retaining only potential matching models, resulting in a relatively reduced computational burden;
- By anticipating the possible movement trends of the target and reactivating more suitable models, the algorithm can better adapt to state changes.

CONCLUSION

This paper investigates the existing VSMM framework and introduces a novel model-set design method based on the EMA algorithm. On the one hand, through model set partitioning, some implausible models in the basic model set are removed, thus, reducing unnecessary competition between models. On the other hand, by utilizing the OTSU algorithm to obtain variable thresholds during model set partitioning, the irrationality and limitations of previously fixed thresholds are effectively circumvented. Moreover, a maneuvering model for star-convex extended targets in a uniform acceleration scenario is constructed, employing the translational invariance of radial functions and the inverse cosine function formula. Simulation experiments in fixed and random scenarios further substantiate that the proposed algorithm can efficiently enhance the tracking performance of targets in maneuvering scenarios.

CONFLICT OF INTEREST

Nothing to declare.

AUTHOR CONTRIBUTIONS

Conceptualization: Zhang J; **Data curation:** Zhang J and Sun L; **Formal analysis:** Sun L; **Acquisition of funding:** Sun L; **Research:** Zhang J and Sun L; **Methodology:** Zhang J; **Project administration:** Sun L and Gao D; **Resources:** Sun L and Gao D; **Software:** Zhang J and Sun L; **Supervision:** Sun L and Gao D; **Validation:** Zhang J and Sun L; **Writing - Preparation of original draft:** Zhang J and Sun L; **Writing - Proofreading and editing:** Zhang J and Sun L.

FUNDING

ACKNOWLEDGMENTS

REFERENCES

Amrouche N, Khenchaf A, Berkani D (2018) Detection and Tracking Low Maneuvering Target in a High Noise Environments. Paper presented at 2018 International Conference on Radar (RADAR). IEEE; Brisbane, Australia. https://doi.org/10.1109/RADAR.2018.8557244

Baum M (2013) Simultaneous Tracking and Shape Estimation of Extended Objects (doctoral dissertation). Karlsruhe: Karlsruhe Institute of Technology.

Baum M, Hanebeck UD (2014) Extended Object Tracking with Random Hypersurface Models. IEEE Trans Aerosp Electron Syst 50(1):149-159. https://doi.org/10.1109/TAES.2013.120107

Blom HAP (1984) An Efficient Filter for Abruptly Changing Systems. Paper presented at The 23rd IEEE Conference on Decision and Control. IEEE; Las Vegas United States. https://doi.org/10.1109/CDC.1984.272089

Deng Q, Chen G, Lu H (2019) Adaptive Sample-Size Unscented Particle Filter with Partitioned Sampling for Three-Dimensional High-Maneuvering Target Tracking. Appl Sci 9(20):4278. https://doi.org/10.3390/app9204278

Lan J, Li X-R (2013) Equivalent-model Augmentation for Variable-structure Multiple-model Estimation. IEEE Trans Aerosp Electron Syst 49(4):2615-2630. https://doi.org/10.1109/TAES.2013.6621840

Li X-R, Bar-Shalom Y (1996) Multiple-model estimation with variable structure. IEEE Trans Autom Control 41(4):478-493. https://doi.org/10.1109/9.489270

Li X-R, Zhang Y (2000) Multiple-Model Estimation with Variable Structure Part V: Likely-Model Set Algorithm. IEEE Trans Aerosp Electron Syst 36(2):448-466. https://doi.org/10.1109/7.845222

Li X-R, Jilkov VP, Ru J (2005) Multiple-model Estimation with Variable Structure-part VI: Expected-mode Augmentation. IEEE Trans Aerosp Electron Syst 41(3):853-867. https://doi.org/10.1109/TAES.2005.1541435

Liu S, Xue M, Qiu Y, Zhou X, Zhao Q (2022) Design of the Missile Attitude Controller Based on the Active Disturbance Rejection Control. J Aerosp Technol Manag 14:e1322. https://doi.org/10.1590/jatm.v14.1255

Marošević T (2018) The Hausdorff distance between some sets of points. Math Commun 23(2):247-257. https://hrcak.srce.hr/198692

Otsu N (1979) A Threshold Selection Method from Gray-level Histograms. IEEE Trans Syst Man Cybern 9(1):62-66. https://doi.org/10.1109/TSMC.1979.4310076

Qian F, Lei H (2021) A pupil location algorithm based on cascaded Haar feature and OTSU dynamic threshold segmentation. Paper presented at AOPC 2021: Optical Sensing and Imaging Technology (Vol. 12065, pp. 796-801). SPIE; Beijing, China. https://doi.org/10.1117/12.2607002

Qu HQ, Pang LP, Li SH (2009) A novel interacting multiple model algorithm. Signal Processing 89(11):2171-2177. https://doi.org/10.1016/j.sigpro.2009.04.033

Shen N, Chen L, Lu X, Hu H, Pan Y, Gao Z, Liu X, Liu Z, Chen R (2021) Online Displacement Extraction and Vibration Detection Based on Interactive Multiple Model Algorithm. Mech Syst Signal Process 155:107581. https://doi.org/10.1016/j.ymssp.2020.107581

Shen-tu H, Luo J-a, Xue A, Peng D, Guo Y (2018) A Novel Variable Structure Multi-Model Tracking Algorithm Based on Error-Ambiguity Decomposition. Paper presented at 2018 21st International Conference on Information Fusion (FUSION). IEEE; Cambridge, United Kingdom. https://doi.org/10.23919/ICIF.2018.8455416

Sun L, Yu H, Fu Z, He Z, Zou J (2021) Modeling and Tracking of Maneuvering Extended Object with Random Hypersurface. IEEE Sens J 21(18):20552-20562. https://doi.org/10.1109/JSEN.2021.3097967

Xiao J, Li H, Jin H (2022) Transtrack: Online Meta-Transfer Learning and OTSU Segmentation Enabled Wireless Gesture Tracking. Pattern Recognit 121:108157. https://doi.org/10.1016/j.patcog.2021.108157

Yunita M, Suryana J, Izzuddin A (2020) Error Performance Analysis of IMM-Kalman Filter for Maneuvering Target Tracking Application. Paper presented at 2020 6th International Conference on Wireless and Telematics (ICWT). IEEE; Yogyakarta, Indonesia. https://doi.org/10.1109/ICWT50448.2020.9243662

Zhang X, Yan Z, Chen Y, Yuan Y (2022) A novel particle filter for extended target tracking with random hypersurface model. Appl Math Comput 425:127081. https://doi.org/10.1016/j.amc.2022.127081

Zhao L (2022) Motion Track Enhancement Method of Sports Video Image Based on OTSU Algorithm. Wirel Commun Mob Comput 2022:8354075. https://doi.org/10.1155/2022/8354075

Zhu M, Yu F, Xiao S (2019) An Unconventional Multiple Low-Cost IMU and GPS-Integrated Kinematic Positioning and Navigation Method Based on Singer Model. Sensors 19(19):4274. https://doi.org/10.3390/s19194274