electronics

MDPI

Article

A Variable Structure Multiple-Model Estimation Algorithm Aided by Center Scaling

Qiang Wang, Guowei Li, Weitong Jin, Shurui Zhang * and Weixing Sheng

School of Electronic and Optical Engineering, Nanjing University of Science and Technology, Nanjing 210094, China

* Correspondence: shuruizhang@njust.edu.cn

Abstract: The accuracy for target tracking using a conventional interacting multiple-model algorithm (IMM) is limited. In this paper, a new variable structure of interacting multiple-model (VSIMM) algorithm aided by center scaling (VSIMM-CS) is proposed to solve this problem. The novel VSIMM-CS has two main steps. Firstly, we estimate the approximate location of the true model. This is aided by the expected-mode augmentation algorithm (EMA), and a new method—namely, the expected model optimization method—is proposed to further enhance the accuracy of EMA. Secondly, we change the original model set to ensure the current true model as the symmetry center of the current model set, and the model set is scaled down by a certain percentage. Considering the symmetry and linearity of the system, the errors produced by symmetrical models can be well offset. Furthermore, narrowing the distance between the true model and the default model is another effective method to reduce the error. The second step is based on two theories: symmetric model set optimization method and proportional reduction optimization method. All proposed theories aim to minimize errors as much as possible, and simulation results highlight the correctness and effectiveness of the proposed methods.

Keywords: variable structure of interacting multiple-model; symmetric model set optimization method; proportional reduction optimization method; expected model optimization method

check for updates

Citation: Wang, Q.; Li, G.; Jin, W.; Zhang, S.; Sheng, W. A Variable Structure Multiple-Model Estimation Algorithm Aided by Center Scaling. *Electronics* **2023**, *12*, 2257. https://doi.org/10.3390/electronics12102257

Academic Editor: Dimitris Apostolou

Received: 27 March 2023
Revised: 11 May 2023
Accepted: 12 May 2023
Published: 16 May 2023

1. Introduction

Multiple-model (MM) is an advanced method to solve many problems, especially the target tracking problem [1,2]. Compared with traditional algorithms combined with radar systems [3,4], MM's power of MM comes from the teamwork of multi parallel estimators [5], not only a single estimator. The MM approach has been presented for more than fifty years, and it was first proposed in [6,7]. It has a mature framework system [8,9], and the parallel structure using Bayesian filters proves its great performance. Usually, the model set designed in advance or generated in real-time is used to cover the possible true models. Then, the system dynamics can be described as hybrid systems [10,11] with discrete modes and continuous states. The model set used during the target tracking process has a big influence on estimation results [12], and a better model set often leads to more precise tracking results. The overall estimation is the combination of all estimations from the parallel-running Bayesian filters [13,14]. In recent decades, the MM methods have achieved rapid development [15]. The MM has been used widely because of its completeness of method and easiness of implementation. In addition, it goes through three stages altogether [16]: static MM (SMM), interacting MM (IMM) and variable-structure interacting of MM (VSIMM).

Compared with SMM, the biggest advantage of IMM is considering the jumps between models [6,7], and this drawback was fixed by Blom and Bar-Shalom [17]. Many advanced methods of IMM have been proven that improving tracking accuracy and not increasing computation burden can be conducted at the same time [18–20]. A reweighted interacting multiple model algorithm [21], which is a recursive implementation of a maximum a posteriori (MAP) state sequence estimator, is a competitive alternative to the popular IMM algorithm and GPB methods. Considering the non-Gaussian white noise, the interacting multiple model based on maximum correntropy Kalman filter (IMM-MCKF) [22] is presented to combine interacting multiple model with MCKF to deal with the impulsive noise [23]. Furthermore, to overcome the small kernel bandwidth, changing the kernel is also presented in it. Not only that, some emerging technologies, such as neural network [24] combined with MM also achieves rapid development. For example, the multiple model tracking algorithm of the multiple processing switch is a reliable method to improve the tracking precision [25]. However, the inherent defect, fixed model structure, of IMM limits its development to a large extent. In many real-world scenarios, the true model space is not discrete [26], but continues and uncountable. It is hard to design a model set to cover all possible models. Therefore, it seems natural to design a model set including huge models to fit the true model space perfectly. However, too many models leads to competition between models, even worse than few models [5]. In addition, the surging computation burden is another severe problem. Loosely speaking, the major objective of MM is to find a method using as few models as possible to achieve better performance of target tracking.

FIMM and SMM both use fixed model set at all time, without considering the properties of true model space. VSIMM [27–29] is a successful method to consider. On the one hand, it uses a limited number of models to reduce computation burden. On the other hand, the model set generated on the current is closer than the original model set. Generally, the VSIMM adapts the continuous mode space. It seems that tracking precision and acceptable computation burden are both solved perfectly. The key problem of VSIMM is how to design a highly cost-effective model set adaptive (MSA) mechanism. The model-group switching algorithm (MGS) has been used widely in a large class of problems with hybrid (continuous and discrete) uncertainties [30]. Compared with FIMM, the main advantage of MGS is reducing computation significantly, but it has limited performance improvements. The expected-mode augmentation covers a large continuous mode space by a relatively small number of models at a given accuracy level [31]. Though the expected model is closer than any other model, the results may even deteriorate because it is not precise enough. Similarly, the model set is augmented by a variable model intended to best match the unknown true model, and the algorithm is named the equivalent-model augmentation algorithm (EqMA) [32]. Compared with IMM and EMA, the EqMA has stronger timeliness. The likely-model set algorithm is also an adaptive method for VSIMM, and its cost-effectiveness is more valid than many other methods [33]. However, those three algorithms mentioned in [33] are more difficult to implement. A method using hypersphere-symmetric model-subset and axis-symmetric model-subset is presented as the fundamental model-subset for multiple models estimation with fixed structure, variable structure, and moving bank [34]. Different kinds of VSIMM algorithms provide reasonable methods to achieve model sets, and MSA is still an open topic to study.

With the gradual upgrading of radar detection systems, the tracking of maneuvering targets requires better accurate results. VSIMM-CS not only meets the requirements well in terms of tracking accuracy but also hardly adds any computation. In this paper, a variable structure of multiple-model algorithm aided by center scaling (VSIMM-CS) is proposed for providing a rational method to generate model sets in real-time. Considering the properties of the linear system [35] and the effect of distance between the model set and the true model on final error, we provide a symmetric model set optimization method and proportional reduction optimization method. For the Kalman filter [36], it is a linear system, and if the true model is at the geometric center of the model set, any two symmetric models produce opposite errors. Therefore, if the model set has an even number of models, the overall error converges to 0. From another point of view, the error of a single filter is related to the distance of the model from the actual model; thus, it is reasonable to scale down the model set by a certain percentage. Those two theories are based on the current true model. Therefore, to find a model that is closer to the true model, the expected model optimization method is proposed. The modified model is closer to the real model than the original model.

For many existing methods, such as FIMM, VSIMM-CS has excellent cost performance ratio without more computation in the design of the initial model set. Compared with many VSIMM algorithms, just like LMS proposed in [33], VSIMM has better implementability with the same computation. In general, VSIMM-CS shows high precision and universality, and it is also easy to implement.

The remaining parts of the paper are organized as follows: Section 2 introduces the processes of the IMM and VSIMM. Section 3 provides three optimization methods: symmetric model set optimization method, proportional reduction optimization method, and expected model optimization method to prove the feasibility. Section 4 presents the process of VSIMM-CS. Finally, Section 5 provides the conclusion.

2. Multiple-Model Algorithm

In this section, the processing of VSIMM and FIMM are briefly introduced.

2.1. The Process of FIMM

If the model set $M = \{m^{(1)}, m^{(2)}, ..., m^{(N)}\}$ is determined in advance, the model probability transition matrix Π is also determined. The transition probability from i-th model ($m^{(i)}$) to j-th model ($m^{(j)}$) is π_{ij}. If a multiple-model system has M models, each of them can be denoted as

$$\begin{aligned} &x_k^{(i)} = F_{k-1}^{(i)} x_{k-1}^{(i)} + G_{k-1}^{(i)} (a_{k-1} + w_{k-1}^{(i)}) \\ &z_k^{(i)} = H_k^{(i)} x_k^{(i)} + v_k^{(i)} \\ &i = 1, ..., N \end{aligned} \tag{1}$$

where $x = (x, \dot{x}, y, \dot{y})'$ is the target state; $a = (a_x, a_y)'$ is the acceleration; the process noise is $w_k \sim N[0, Q]$; the measurement value is z and its random measure error is $v \sim N[0, R]$; F represents the state transition matrix, G represents the acceleration input matrix, and H is the observation matrix.

Assume that the best target estimation, the state estimation covariance matrix and the model probability of $m^{(i)}$ are $\hat{x}_{k|k}^{(i)}$, $p_{k|k}^{(i)}$ and $u_k^{(i)}$ at the time k, respectively. Then, the overall state estimation and state estimation covariance are

$$\hat{x}_{k|k} = \sum\nolimits_i x_{k|k}^{(i)} u_k^{(i)} \tag{2}$$

$$p_{k|k} = \sum\nolimits_i u_k^{(i)} [p_{k|k}^{(i)} + (\hat{x}_{k|k} - \hat{x}_{k|k}^{(i)})(\hat{x}_{k|k} - \hat{x}_{k|k}^{(i)})'] \tag{3}$$

2.2. The Process of VSIMM

The biggest difference between FIMM and VSIMM is whether the model set changes through the target tracking process. For a maneuvering target, its mode space of acceleration S is very large and even uncountable. In most cases, the real model does not fall on the model set. Thus, using a limited model set M to approach S is unreasonable. Simply increasing the number of models does not improve the results, and it has a high probability to degrade the performance, even worse than very few models. The significant advantages of VSIMM are high precision, less computation, and strong adaptability.

If we obtain the current model set M_k through a specific method, and $M_k = \{m_k^{(j)}$, $j = 1, ..., n_k\}$. The model set M_k at any time is included in the total model set M. Then, the model transition probability is $\pi_{ij} = p\{m_k^{(j)} | m_{k-1}^{(i)}\} = p\{m_k^{(j)} | m_{k-1}^{(i)}, s_k \in M\}$. The overall state estimation and state estimation covariance based on $\hat{x}_{k|k}^{(i)|M_k}$, $p_{k|k}^{(i)|M_k}$ and $u_k^{(i)|M_k}$ are, respectively,

$$\hat{x}_{k|k} = \sum_{m^{(i)} \in M_k} \hat{x}_{k|k}^{(i)|M_k} u_k^{(i)|M_k} \tag{4}$$

$$p_{k|k} = \sum_{m^{(i)} \in M_k} [p_{k|k}^{(i)|M_k} + (\hat{x}_{k|k}^{(i)|M_k} - \hat{x}_{k|k}) \\ (\hat{x}_{k|k}^{(i)|M_k} - \hat{x}_{k|k})']u_{k|k}^{(i)|M_k} \tag{5}$$

3. Model Optimization Method

In this section, we introduce three methods to optimize the model sets, including symmetric model set optimization method, proportional reduction optimization method and expected model optimization method.

3.1. Symmetric Model Set Optimization Method

A linear system can be described as shown in Figure 1. In addition, X is input and output is $AX + B$. If input is $-X$, the output is $-Ax + B$. It is clear that the output of the sum of $-X$ and X is $2B$. If AX is the error produced by the system, two opposite inputs eliminate errors well.

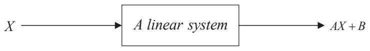


Figure 1. A linear system.

It is assumed that two model sets $M^{(1)}$ and $M^{(2)}$, where $M^{(1)} = \{m_1^{(1)}, m_2^{(1)}, ..., m_N^{(1)}\}$ and $M^{(2)} = \{m_1^{(2)}, m_2^{(2)}, ..., m_N^{(2)}\}$. The current motion mode is s_k. $M^{(1)}$ and $M^{(2)}$ are centrosymmetric and axially symmetric, respectively. However, the centers of symmetries are different. $M^{(1)}$'s center of symmetry is (0,0) and $M^{(2)}$ is s_k. Obviously, the connection between $M^{(1)}$ and $M^{(2)}$, as shown in Figure 2, is

$$m_i^{(1)} + s_k = m_i^{(2)}, i = 1, ..., N \tag{6}$$

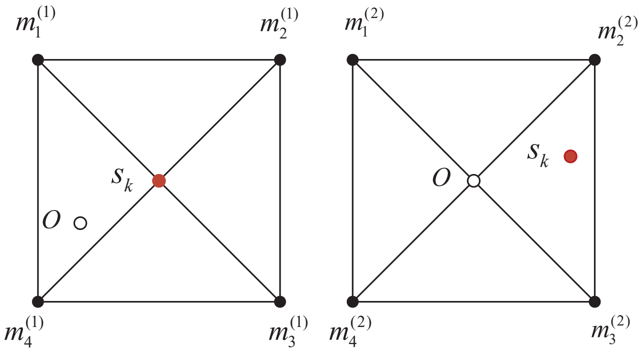


Figure 2. Example of $M^{(1)}$ and $M^{(2)}$.

For $M^{(1)}$, the distance between s_k and $m_i^{(1)}$ is different, and the magnitude of the distance is

$$|m_{q^1}^{(1)} - s_k| \leq |m_{q^2}^{(1)} - s_k| \leq \cdots \leq |m_{q^N}^{(1)} - s_k| \tag{7}$$

For IMM algorithm, the connection between the model probability $u^{(i)}$ and the distance $|m^{(i)} - s_k|$ is

$$|m^{(i)} - s_k| \propto \frac{1}{u^{(i)}} \tag{8}$$

Then, the following relation can be determined as

$$u_{q^1}^{(1)} \leq u_{q^2}^{(1)} \leq ... \leq u_{q^N}^{(1)} \tag{9}$$

For each model $m_i^{(1)}$, they have different errors $\hat{x}_{k|k}^{(i)|M^{(1)}} - s_k = \delta_i^{(1)}$; thus, the overall estimation error is

$$ERROR^{(1)} = \sum_i u_i^{(1)} \delta_i^{(1)} \tag{10}$$

Clearly, according to (8), the overall estimation error has been reduced to a certain extent. However, since the system is linear and it is asymmetrical with respect to s_k, the error of each $m_i^{(1)}$ is not eliminated well.

Since $M^{(2)}$ holds symmetric properties, the relationships can be obtained as

$$\begin{cases} m^{(2)}_{p_1^{(1)}} - s_k = \ldots = m^{(2)}_{p_1^{(2/N_1)}} - s_k \\ = s_k - m^{(2)}_{p_1^{(2/N_1+1)}} \ldots = s_k - m^{(2)}_{p_1^{(N_1)}} = \delta^{(2)}_{p_1} \\ \vdots \\ m^{(2)}_{p_i^{(1)}} - s_k = \ldots = m^{(2)}_{p_i^{(2/N_i)}} - s_k \\ = s_k - m^{(2)}_{p_i^{(2/N_i+1)}} \ldots = s_k - m^{(2)}_{p_i^{(N_i)}} = \delta^{(2)}_{p_i} \\ \vdots \\ m^{(2)}_{p_n^{(1)}} - s_k = \ldots = m^{(2)}_{p_n^{(2/Nn)}} - s_k \\ = s_k - m^{(2)}_{p_n^{(2/Nn+1)}} \ldots = s_k - m^{(2)}_{p_n^{(Nn)}} = \delta^{(2)}_{p_n} \end{cases} \tag{11}$$

where $N = \sum_i N_i$, and N_i is even number. In this linear system, $m^{(2)}_{p_j^{(i)}}$ and $m^{(2)}_{p_j^{(i+N_j/2)}}$ produce two opposite errors $\varepsilon^{(2)}_{p_j}$ and $-\varepsilon^{(2)}_{p_j}$, and

$$\begin{aligned} \varepsilon^{(2)}_{p_j} &= x_{k|k}^{(p_j)|M^{(2)}} - s_k \\ -\varepsilon^{(2)}_{p_j} &= x_{k|k}^{(p_{j+N_j/2})|M^{(2)}} - s_k \end{aligned} \tag{12}$$

Theoretically, if $M^{(2)}$ is strictly symmetric with respecting to s_k, the overall errors are equal to 0, without considering the system noise.

3.2. Proportional Reduction Optimization Method

Suppose two model sets $M^{(1)} = \{m_1^{(1)}, m_2^{(1)}, ..., m_N^{(1)}\}$ and $M^{(2)} = \{m_1^{(2)}, m_2^{(2)}, ..., m_N^{(2)}\}$, and the current model is s_k. The relationship between $M^{(1)}$ and $M^{(2)}$ is

$$\frac{m_i^{(1)} - s_k}{m_i^{(2)} - s_k} = \alpha > 1 \tag{13}$$

Then, the model sets obeying the equation above can be called relative position invariant model sets. The most important feature is that the position of the model sets relative of s_k does not change, as shown in Figure 3.

It is obvious that $M^{(2)}$ is more likely to have better performance than $M^{(1)}$. However, the precondition is that the topology formed by $M^{(1)}$ includes all possible real models, as shown in Figure 4. If the topology is tangent to the true mode space S, this situation can be called the critical point, namely, $\alpha = \alpha_0$.

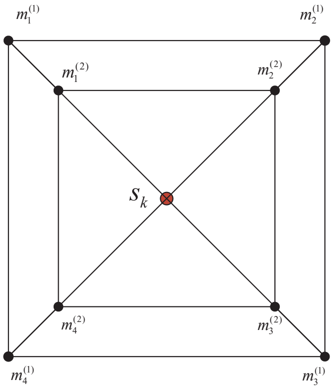


Figure 3. Topology structure of $M^{(1)}$ and $M^{(2)}$.

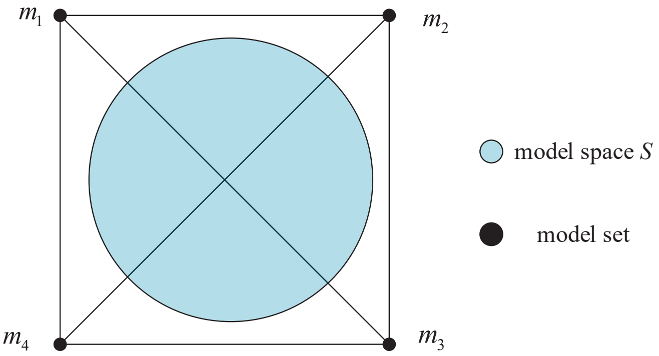


Figure 4. Topology of a model set including all model space S.

Thus, if some noise could be tolerated, the following relationships are determined as

$$\begin{cases} u_1^{(1)} = u_1^{(2)} \\ u_2^{(1)} = u_2^{(2)} \\ \quad\vdots \\ u_N^{(1)} = u_N^{(2)} \end{cases} \tag{14}$$

For IMM, the distance between m_i and s_k directly affects the final error

$$|m_i - s_k| \propto |\hat{x}_{k|k}^{(i)} - s_k| \tag{15}$$

From (13), if the value of α is suitable, it is easy to deduce as

$$|\hat{x}_{k|k}^{(i)|M^{(2)}} - s_k| < |\hat{x}_{k|k}^{(i)|M^{(1)}} - s_k| \tag{16}$$

The connection between overall estimation error of $M^{(1)}$ and $M^{(2)}$ is

$$|ERROR^{(2)}| < |ERROR^{(1)}| \tag{17}$$

Generally, if the model set is scaled down, its performance improves. However, the scale should not be too small, or even exceed the critical point; otherwise, it may make the covariance matrix irreversible.

3.3. Expected Model Optimization Method

If the theories referenced in Sections 3.1 and 3.2 is feasible, the common condition is: the current model s_k is known or a model is found to approach s_k as close as possible. Then, an effective method to approach the current model s_k, called expected-mode augmentation, is given in [31]. The current expected model m_e is the weighted sum of all the models at the current time

$$m_e = \sum_i u_i m_i \tag{18}$$

and it is closer than any model to s_k, namely, $|m_e - s_k| < \min\{|m_1 - s_k|, |m_2 - s_k|, \ldots, |m_N - s_k|\}$. However, the value of $|m_e - s_k|$ is not small enough, as shown in Figure 5. Expected model m_e may appear in anywhere in expected model space. If $m_e + \gamma = s_k$, where γ is the error, it is necessary to find a method to reduce γ to a certain extent.

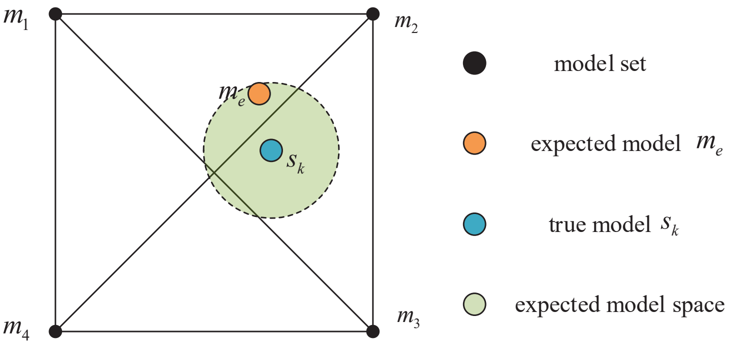


Figure 5. Example of the mentioned hypothesis.

If m_e is scaled a little bit, let m_e becomes λm_e, where λ is scaling factor. Thus, (18) is rewritten as

$$s_k - \lambda m_e = \gamma - (\lambda - 1)m_e \tag{19}$$

and the conditions are

$$\begin{cases} \lambda > 1, \gamma < 0 \\ 0 < \lambda < 1, \gamma > 0 \end{cases} \tag{20}$$

Obviously, the error γ reduces to $\gamma - (\lambda - 1)m_e$. If λ is chosen reasonably, the error becomes small enough and λm_e becomes close enough to s_k, namely, $\frac{\lambda m_e - s_k}{\lambda m_e + s_k} \approx 0$. What is worth noting is increasing or decreasing λ blindly contributes to serious mistakes, including even worse performance.

4. Variable Structure of Interacting Multiple-Model Algorithm Aided by Center Scaling

In this section, we introduce a new VSIMM algorithm: VSIMM-CS. The Sections 3.1–3.3 provide three model optimization theories. The main idea of VSIMM-CS is finding the λm_e to approach the current real model s_k as close as possible, and moving the original model set M and scaling it to obtain the current model set $M^{(k)}$. The current model set $M^{(k)}$ could achieve better performance than the original model set M. The main function of the original model set M is locating the expected model m_e.

The VSIMM algorithm has a clear framework of inputs and outputs. The process of Section 2.2 can be denoted as

$$\begin{aligned} &VSIMM[M^{(k-1)}, M^{(k)}]: \\ &\{\hat{x}_{k|k}^{i|M^{(k)}}, p_{k|k}^{i|M^{(k)}}, u_{k|k}^{i|M^{(k)}}\} \\ &= VSIMM(\hat{x}_{k-1|k-1}^{i|M^{(k-1)}}, p_{k-1|k-1}^{i|M^{(k-1)}}, u_{k-1|k-1}^{i|M^{(k-1)}}) \end{aligned} \tag{21}$$

Obviously, if $M^{(k)} = M^{(k-1)}$, the processing becomes to FIMM. Thus, the FIMM is a special case of VSIMM.

$$\begin{aligned} & FIMM[M]: \\ & \{\hat{x}_{k|k}^{i|M}, p_{k|k}^{i|M}, u_{k|k}^{i|M}\} \\ & = FIMM(\hat{x}_{k-1|k-1}^{i|M}, p_{k-1|k-1}^{i|M}, u_{k-1|k-1}^{i|M}) \end{aligned} \tag{22}$$

The novel VSIMM-CS always has an original model set M, which is taken part in the whole processing of target tracking, and $M^{(k)}$ is always generated by M. $M^{(k)}$ and $M^{(k-1)}$ are probably completely different. The steps of VSIMM-CS are shown in Algorithm 1.

Algorithm 1 VSIMM-CS Process

S1: Increase the time counter k by 1.

S2: Run the $FIMM[M]$ cycle, and obtain the outputs: $\hat{x}_{k+1|k+1}^{i}$,$u_{k+1|k+1}^{i}$ and $p_{k+1|k+1}^{i}$, based on the model set M.

S3: Obtain the expected model m_e by using EMA algorithm, which is the weighted summation of all models.

$$m_e = \sum_i u_{k+1|k+1}^{i} m_i$$

S4: Select the suitable values of α and λ to generate the current model set $M^{(k+1)}$. $M^{(k+1)} = \alpha\{m_i + \lambda m_e, i = 1, 2, ..., N\}$

S5: Run the $VSIMM[M^{(k)}, M^{(k+1)}]$ cycle to obtain the final results.

$$\hat{x}_{k+1|k+1} = \sum_{m^{(i)} \in M^{(k+1)}} \hat{x}_{k+1|k+1}^{(i)|M^{(k+1)}} u_{k}^{(i)|M^{(k+1)}}$$

$$\begin{aligned} p_{k+1|k+1} = \sum_{m^{(i)} \in M^{(k+1)})} & [p_{k+1|k+1}^{(i)|M^{(k+1)}} + (\hat{x}_{k+1|k+1}^{(i)|M^{(k+1)}} - \hat{x}_{k+1|k+1}) \\ & (\hat{x}_{k+1|k+1}^{(i)|M_{(k+1)}} - \hat{x}_{k+1|k+1})']u_{k+1|k+1}^{(i)|M^{(k+1)}} \end{aligned}$$

S6: Go to **S1**.

The algorithm complexity is $T(n) = 2n^2 + 16n$, and n equals to the number of models.

The most important part of the process is model set generation. The values of λ and α have a huge impact on the performance of the model set. Therefore, it is unreasonable to choose λ and α blindly. The biggest advantage of VSIMM-CS is that the model set is generated in real-time without increasing huge computational complexity. In addition, if the original model set is designed rational and the number of models is few, the proposed VSIMM-CS can achieve rewarding results and keep the calculation volume within a reasonable range at the same time. The rational combination of λ and α achieve great performance in terms of precision.

5. Simulation Results

In this section, a reasonable simulation process is given: firstly, the target state, the measurement equations and their specific parameters are presented. Secondly, the original model set is given, including the probability transition matrix. Thirdly, the target motion state and performance criterion are presented. Finally, different simulation results are analyzed.

The target state and the measurement equations are, respectively, given as

$$x_{k+1} = F_k^{(j)} x_k + G_k^{(j)} a_k^{(j)} + w_k^{(j)} \tag{23}$$

$$z_k = H_k^{(j)} x_k + v_k^{(j)} \tag{24}$$

where $w_k^{(j)} \sim N[0, 0.01]$; $v_k^{(j)} \sim N[0, 1250]$; $a_k^{(i)}$ is target acceleration input; $F = \begin{bmatrix} 1 & T & 0 & 0 \\ 0 & 1 & T & 0 \\ 0 & 0 & 1 & T \\ 0 & 0 & 0 & 1 \end{bmatrix}$ and $G = \begin{bmatrix} 0.5 & 1 & 0 & 0 \\ 0 & 0 & 0.5 & 1 \end{bmatrix}'$. T is time interval.

The original model set M included 4 models as

$$\begin{Bmatrix} m_1 = [-10, 10]' & m_2 = [10, 10]' \\ m_4 = [-10, -10]' & m_3 = [10, -10]' \end{Bmatrix} \tag{25}$$

and its topology structure is shown in Figure 6

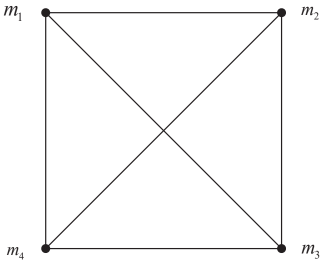


Figure 6. Topology structure of M.

Its probability transition matrix Π is $\begin{bmatrix} 0.85 & 0.05 & 0.05 & 0.05 \\ 0.05 & 0.85 & 0.05 & 0.05 \\ 0.05 & 0.05 & 0.85 & 0.05 \\ 0.05 & 0.05 & 0.05 & 0.85 \end{bmatrix}$. The Table 1 illustrates an ensemble of maneuver trajectories for compare the existing and proposed algorithm.

Table 1. Parameters of deterministic scenarios.

Time k (s)	a_x **(m/s^2)**	a_y **(m/s^2)**
1–50	0	0
50–100	5	5
100–150	3	−7
150–200	7	−2
200–250	4	1
250–300	−4	−2
300–350	0	0

If $\bar{e}$ represents the average error of the whole processing, it can be denoted as

$$\bar{e} = \sum_{k=k_1}^{k_N} \sqrt{\left((\hat{x}_k - x_k)' \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} \right) * \left(\begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} (\hat{x}_k - x_k) \right)} \Big/ (k_N - k_1) \tag{26}$$

where $\hat{x}_k$ is the best estimation at time k, x_k is the target state; and $N = 300$. Therefore, different λ and α produce different $\bar{e}$, as shown in Table 2. The unit of $\bar{e}$ is meters, and we just consider the position error.

Table 2. The average error $\bar{e}$ of the tracking process.

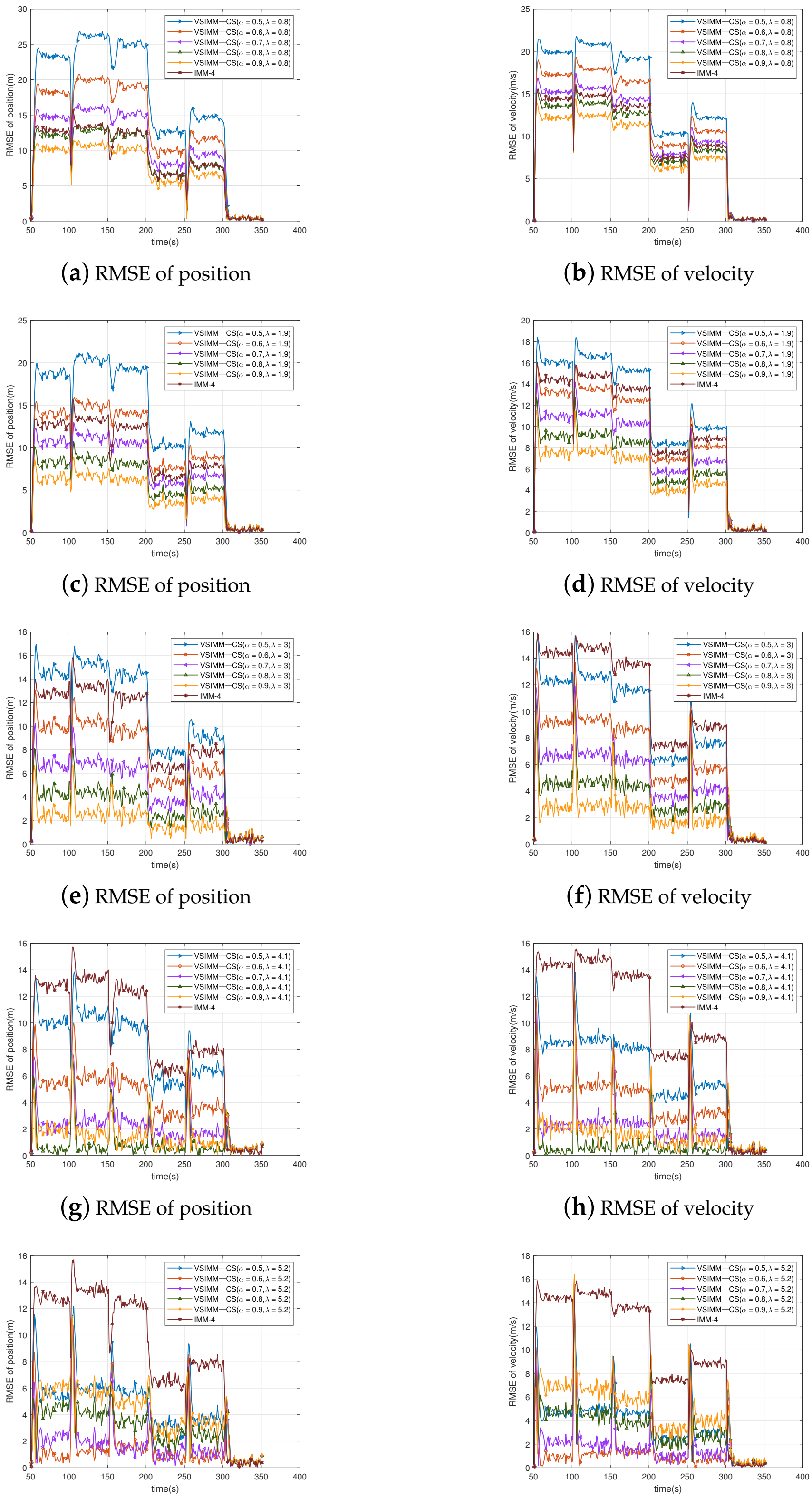
α λ	**0.5**	**0.6**	**0.7**	**0.8**	**0.9**
0.8	16.72	13.00	10.45	8.59	7.19
1.9	13.29	9.89	7.55	5.84	4.56
3	10.16	6.96	4.76	3.15	1.98
4.1	7.21	4.16	2.09	0.98	1.52
5.2	4.32	1.53	1.86	3.14	4.09

Different combinations of λ and α present different performances. Assume λ = 0.8, 1.9, 3, 4.1, 5.2 and $\alpha = 0.5, 0.6, 0.7, 0.8, 0.9$. Their root mean square errors (RMSE) are shown in Figure 7 with 50 times Montecarlo experiment.

In most cases, the VSIMM-CS shows great performance on target tracking, and its errors are far lower than FIMM using 4 models. In some cases, the error is close to 0, and the proposed VSIMM-CS achieves a perfect performance. Thus, sacrificing a certain amount of computation for high accuracy is an acceptable approach.

When $\lambda = 0.8$, as shown in Figure 7a,b, most values of α do not present better performance than IMM-4. Especially, when $\alpha = 0.5$, it becomes the worst one and it produces twice as many errors as IMM-4. As α is increasing, the errors reduce gradually. Until $\alpha = 0.8$, the error curve is as same as IMM-4. When $\alpha = 0.9$, it becomes better than IMM-4. Obviously, in the situation of $\lambda = 0.8$, VSIMM-CS does not show its high precision. When $\lambda = 1.9$ in Figure 7c,d, compared with $\lambda = 0.8$, the tracking precision has been improved. Noting that when $\alpha = 0.7$ or 0.8, the errors are smaller than IMM-4. However, when $\alpha = 0.5$, it is still the worst one among them. It is surprising that, except $\alpha = 0.5$ when $\lambda = 3$, other values of α have greater performance than IMM-4. The relationship of these five algorithms has not changed. $\alpha = 0.9$ is still the best one and $\alpha = 0.5$ is still the worst one. Only when $\alpha = 0.5$, the performance is worse than IMM-4. While the minimum position error is about two meters, it is not close enough to 0. Clearly, up to now, λ does not achieve the most suitable value. When $\lambda = 4.1$, all VSIMM-CS algorithms have better performance than IMM-4, even $\alpha = 0.5$, $\alpha = 0.7$, and $\alpha = 0.9$ have similar tracking precision, and $\alpha = 0.9$ is not the best one anymore. When $\alpha = 0.7$, the position error is smaller than that in any situation as shown in Figure 7. When the true model does not equal to $(0, 0)'$, the position error is also very close to 0. It seems that the best combination is $\lambda = 4.1$ and $\alpha = 0.8$ in this system. In this case, the conclusion, referenced in Section 3, has been fully demonstrated. As shown in Figure 7i,j, VSIMM-CS still shows its great performance. All the error curves are less than half of IMM-4's, and $\alpha = 0.6$ has the highest accuracy. The simulation result of $\alpha = 0.7$ is very close to $\alpha = 0.6$. However, the best one shown in Figure 7g,h does not maintain its advantage in Figure 7i,j.

For $\alpha = 0.5$, as λ changes, its precision does not become better, except $\lambda < 5.2$. In these simulation results, it is always the worst one among VSIMM-CS algorithms. The main reason for this is that the value of α is too small. Similarly, $\alpha = 0.6$ still does not meet the requirements. Compared with $\alpha = 0.5$, its performance becomes better. It is not difficult to find that VSIMM-CS takes twice as much computation as IMM-4. However, in most cases, the promotion of $\alpha = 0.6$ is very limited, even it is worse than IMM-4 when $\lambda = 0.9$ or 1.9. Only when $\lambda = 5.2$ does it have the best performance, which is shown in Figure 7i,j. When $\alpha = 0.7$, the precision becomes acceptable. Though it is not the best one in all situations, the performance is greatly promoted. As λ increases, the tracking precision of the proposed algorithm becomes better. When $\alpha = 0.8$ and $\lambda \leq 3$, it only performs worse than $\alpha = 0.9$. Especially when $\lambda = 4.1$, its performance becomes the best one. However, if the value of λ is too large, referenced as Figure 7ij, its advantage disappears. When $\alpha = 0.9$, its performance is always the best one in Figure 7a–f. Similarly, when $\lambda = 5.2$, so much large value of λ would deteriorate the performance.



(**a**) RMSE of position

(**b**) RMSE of velocity

(**c**) RMSE of position

(**d**) RMSE of velocity

(**e**) RMSE of position

(**f**) RMSE of velocity

(**g**) RMSE of position

(**h**) RMSE of velocity

(**i**) RMSE of position

(**j**) RMSE of velocity

Figure 7. Root mean square error of VSIMM-CS.

A good demonstration of the average error $\bar{e}$ of the unworkable portfolio is shown in Table 2. If $\alpha < 0.7$, the critical point would not bee meet the condition at certain times.

When $\alpha = 0.7, 0.8, 0.9$, respectively, there is a high probability that the proposed algorithm achieves better performance.

In general, VSIMM-CS certainly achieves a big performance boost compared with FIMM, if λ and α are selected rationally. Different systems tend to have different suitable combinations of λ and α. Smaller α or larger λ do not promote the performance. According to the current system, the reasonable λ and α always reflect satisfactory results.

6. Conclusions

In this paper, a new variable structure interacting multiple-model algorithm, VSIMM-CS, is proposed. Its model set is generated in real-time, and the generated model set is based on the original model set. Considering the error properties of a linear system and the symmetry of model set structure, the two theories called proportional reduction optimization method and symmetric model set optimization method are presented. The main purpose of the two theories is to reduce errors. Without considering the effect of noise, VSIMM-CS eliminates errors perfectly. To better locate the real model, the expected model optimization method is proposed. The excepted model generated by this method is closer than any other model. Simulation results show different combinations of α and λ have different performances. In most cases, VSIMM-CS achieved better tracking results. It is acceptable to sacrifice a certain amount of computation for high accuracy. A huge performance boost can be obtained by the precise selection of α and λ. When the performance achieves in the optimal situation, α and λ are 0.8 and 4.1, respectively. In different simulation conditions, the results may be different. Many factors may influence the values of α and λ, such as noise, original model set, and true mode space, etc., and our following research aims to focus on these factors. However, the unreasonable selection of α and λ leads to worse results. Simulation results also highlight the rationality and feasibility of this novel approach.

Author Contributions: Writing—original draft preparation, Q.W.; investigation, Q.W. and G.L.; writing—review and editing, G.L., W.J. and S.Z.; project administration, S.Z. and G.L.; supervision, W.S.; funding acquisition, S.Z. All authors have read and agreed to the published version of the manuscript.

Funding: This work was supported in part by the National Natural Science Foundation of China under Grants 62001227, 61971224 and 62001232.

Data Availability Statement: Not applicable.

Acknowledgments: The authors thank the reviewers for their great help on the article during its review progress.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Gorji, A.A.; Tharmarasa, R.; Kirubarajan, T. Performance measures for multiple target tracking problems. In Proceedings of the 14th International Conference on Information Fusion, Chicago, IL, USA, 5–8 July 2011; pp. 1–8.
2. Poore, A.B.; Gadaleta, S. Some assignment problems arising from multiple target tracking. *Math. Comput. Model.* **2006**, *43*, 1074–1091. [CrossRef]
3. Huang, X.; Tsoi, J.K.; Patel, N. mmWave Radar Sensors Fusion for Indoor Object Detection and Tracking. *Electronics* **2022**, *11*, 2209. [CrossRef]
4. Wei, Y.; Hong, T.; Kadoch, M. Improved Kalman filter variants for UAV tracking with radar motion models. *Electronics* **2020**, *9*, 768. [CrossRef]
5. Li, X.R.; Bar-Shalom, Y. Multiple-model estimation with variable structure. *IEEE Trans. Autom. Control* **1996**, *41*, 478–493.
6. Magill, D. Optimal adaptive estimation of sampled stochastic processes. *IEEE Trans. Autom. Control* **1965**, *10*, 434–439. [CrossRef]
7. Lainiotis, D. Optimal adaptive estimation: Structure and parameter adaption. *IEEE Trans. Autom. Control* **1971**, *16*, 160–170. [CrossRef]
8. Tudoroiu, N.; Khorasani, K. Satellite fault diagnosis using a bank of interacting Kalman filters. *IEEE Trans. Aerosp. Electron. Syst.* **2007**, *43*, 1334–1350. [CrossRef]
9. Kirubarajan, T.; Bar-Shalom, Y.; Pattipati, K.R.; Kadar, I. Ground target tracking with variable structure IMM estimator. *IEEE Trans. Aerosp. Electron. Syst.* **2000**, *36*, 26–46. [CrossRef]

10. Grossman, R.L.; Nerode, A.; Ravn, A.P.; Rischel, H. *Hybrid Systems*; Springer: Berlin/Heidelberg, Germany, 1993; Volume 736.
11. Branicky, M.S. Introduction to hybrid systems. In *Handbook of Networked and Embedded Control Systems*; Birkhäuser: Basel, Switzerland, 2005; pp. 91–116.
12. Li, X.R. Multiple-model estimation with variable structure. II. Model-set adaptation. *IEEE Trans. Autom. Control* **2000**, *45*, 2047–2060.
13. Labbe, R. Kalman and bayesian filters in python. *Chap* **2014**, *7*, 4.
14. Zhang, G.; Lian, F.; Gao, X.; Kong, Y.; Chen, G.; Dai, S. An Efficient Estimation Method for Dynamic Systems in the Presence of Inaccurate Noise Statistics. *Electronics* **2022**, *11*, 3548. [CrossRef]
15. Rong Li, X.; Jilkov, V. Survey of maneuvering target tracking. Part V. Multiple-model methods. *IEEE Trans. Aerosp. Electron. Syst.* **2005**, *41*, 1255–1321. [CrossRef]
16. Bar-Shalom, Y. *Multitarget-Multisensor Tracking: Applications and Advances*; Artech House, Inc.: Norwood, MA, USA, 2000; Volume iii.
17. Blom, H.A.; Bar-Shalom, Y. The interacting multiple model algorithm for systems with Markovian switching coefficients. *IEEE Trans. Autom. Control* **1988**, *33*, 780–783. [CrossRef]
18. Ma, Y.; Zhao, S.; Huang, B. Multiple-Model State Estimation Based on Variational Bayesian Inference. *IEEE Trans. Autom. Control* **2019**, *64*, 1679–1685. [CrossRef]
19. Wang, G.; Wang, X.; Zhang, Y. Variational Bayesian IMM-filter for JMSs with unknown noise covariances. *IEEE Trans. Aerosp. Electron. Syst.* **2019**, *56*, 1652–1661. [CrossRef]
20. Li, H.; Yan, L.; Xia, Y. Distributed robust Kalman filtering for Markov jump systems with measurement loss of unknown probabilities. *IEEE Trans. Cybern.* **2021**, *52*, 10151–10162. [CrossRef]
21. Johnston, L.; Krishnamurthy, V. An improvement to the interacting multiple model (IMM) algorithm. *IEEE Trans. Signal Process.* **2001**, *49*, 2909–2923. [CrossRef]
22. Fan, X.; Wang, G.; Han, J.; Wang, Y. Interacting Multiple Model Based on Maximum Correntropy Kalman Filter. *IEEE Trans. Circuits Syst. II Express Briefs* **2021**, *68*, 3017–3021. [CrossRef]
23. Davis, R.R.; Clavier, O. Impulsive noise: A brief review. *Hear. Res.* **2017**, *349*, 34–36. [CrossRef]
24. Nie, X. Multiple model tracking algorithms based on neural network and multiple process noise soft switching. *J. Syst. Eng. Electron.* **2009**, *20*, 1227–1232.
25. Mazor, E.; Averbuch, A.; Bar-Shalom, Y.; Dayan, J. Interacting multiple model methods in target tracking: A survey. *IEEE Trans. Aerosp. Electron. Syst.* **1998**, *34*, 103–123. [CrossRef]
26. Gao, W.; Wang, Y.; Homaifa, A. Discrete-time variable structure control systems. *IEEE Trans. Ind. Electron.* **1995**, *42*, 117–122.
27. Li, X.R.; Bar-Shakm, Y. Mode-set adaptation in multiple-model estimators for hybrid systems. In Proceedings of the 1992 American Control Conference, Chicago, IL, USA, 24–26 June 1992; pp. 1794–1799.
28. Pannetier, B.; Benameur, K.; Nimier, V.; Rombaut, M. VS-IMM using road map information for a ground target tracking. In Proceedings of the 2005 7th International Conference on Information Fusion, Philadelphia, PA, USA, 25–28 July 2005; Volume 1, 8p.
29. Xu, L.; Li, X.R. Multiple model estimation by hybrid grid. In Proceedings of the 2010 American Control Conference, Baltimore, MD, USA, 30 June–2 July 2010; pp. 142–147.
30. Li, X.R.; Zwi, X.; Zwang, Y. Multiple-model estimation with variable structure. III. Model-group switching algorithm. *IEEE Trans. Aerosp. Electron. Syst.* **1999**, *35*, 225–241.
31. Li, X.R.; Jilkov, V.P.; Ru, J. Multiple-model estimation with variable structure-part VI: expected-mode augmentation. *IEEE Trans. Aerosp. Electron. Syst.* **2005**, *41*, 853–867.
32. Lan, J.; Li, X.R. Equivalent-Model Augmentation for Variable-Structure Multiple-Model Estimation. *IEEE Trans. Aerosp. Electron. Syst.* **2013**, *49*, 2615–2630. [CrossRef]
33. Li, X.R.; Zhang, Y. Multiple-model estimation with variable structure. V. Likely-model set algorithm. *IEEE Trans. Aerosp. Electron. Syst.* **2000**, *36*, 448–466.
34. Sun, F.; Xu, E.; Ma, H. Design and comparison of minimal symmetric model-subset for maneuvering target tracking. *J. Syst. Eng. Electron.* **2010**, *21*, 268–272. [CrossRef]
35. Callier, F.M.; Desoer, C.A. *Linear System Theory*; Springer Science & Business Media: Berlin, Germany, 2012.
36. Kalman, R.E. A new approach to linear filtering and prediction problems. *J. Basic Eng.* **1960**, *82*, 35–45. [CrossRef]

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.