1. INTRODUCTION

In air combat, it is of great importance to protect high-value aircraft from incoming missiles. Many passive defense methods have been proposed to increase the survivability of aircraft, such as random telegraphing and periodic wave maneuvers [1], [2]. However, with the development of missiles and guidance laws, the performance of these strategies in modern air combat conditions is decreasing. In recent years, active defense methods have attracted considerable attention from researchers [3], [4], [5], [6]. Compared with aircraft utilizing passive defense methods, aircraft with active defense methods have higher survivability and better penetration capability. Many active defense methods are derived assuming that a defender missile has perfect information about an incoming missile’s state and parameters, which is seldom the case in real scenarios [7]. Therefore, a separate identification scheme and estimator must be constructed to identify the missile’s guidance law, parameters, and states.

Guidance law identification is a classification problem that can be solved using a relatively simple neural network or interactive multiple-model (IMM) method. For state estimation, if the guidance law parameters are identified, only one Kalman filter (KF) is needed to estimate the state of an incoming missile. Moreover, an aircraft must maneuver to avoid missile attacks. It is difficult to ensure that the aircraft achieves real-time irradiation of the missile. In this case, if the key parameters can be obtained, the state of the missile can be predicted using these parameters, and the continuous irradiation requirements of the onboard sensors will be greatly reduced. Therefore, this article focuses on the identification of an incoming missile’s parameters.

The guidance and control parameters of a missile include guidance law parameter and a first-order lateral time constant. The required parameters cannot be obtained using direct measurement. Most previous parameter identification studies are based on a KF-based multiple model method. In [8], an estimation approach based on an IMM method is proposed to identify the guidance law and states of a missile. In [9], an adaptive receding horizon controller based on Bayesian inference is presented to identify the missile guidance law with perfect missile information. Most of the previous studies only focused on identifying the guidance law parameter using extended KFs [7], [10], [11] or unscented KFs [8], [12], [13] under the assumption that the first-order lateral time constant is a known constant. However, the flexibility is poor in identifying multiple parameters. To address the uncertainty of parameters (e.g., the drag coefficient, the guidance law, and the first-order lateral constant), multiple-model adaptive estimators (MMAEs) [14] and IMMs [15] are generally adopted in estimation models. Both the methods are derived under the assumption that the system is in a known finite set of possible regimes, and the parameters of the true system are fixed (in MMAEs) or allowed to transition between regimes with a preset transition probability matrix (in IMMs). If this assumption
does not hold, additional models should be added to identify the true state of the system, which may increase the computational burden to the aircraft. Moreover, to limit the model size, generally only the guidance law parameters are identified, while other parameters are considered to be a known constant or given empirically.

Identifying incoming missile parameters is essentially a dynamic multiple-parameter identification problem. With the development of artificial intelligence technology, long short-term memory (LSTM) [16], which is a kind of artificial neural network (ANN), has been widely used to solve time and dynamics problems in aircraft [17], [18], [19]. Compared with MMAEs and IMMs, where each KF corresponds to a specific scenario, an ANN is trained based on a dataset that contains samples extracted from various scenarios. Therefore, the generalization ability of ANNs is higher than that of KFs when multiple parameters need to be identified. Moreover, the computing speed of an ANN is faster since it does not require calculating each KF process. A simplified form of LSTM called a gated recurrent unit (GRU). Compared with conventional LSTM, the training speed of a GRU is improved, while the accuracy is identical. In [20], an identification model based on a GRU neural network is established to identify the guidance law of an incoming missile. The model maintains the assumption that the guidance law is in a finite set known by the aircraft. In [21], a regression guidance law identification model is built without adopting the above simplified assumption, but the model can only be used to identify the guidance law parameters. This article is based on and is an improvement upon [21]. A regression parameter identification model based on a GRU neural network is established in this article. The inputs to the model are available information between the missile and the aircraft. The outputs of the model are the regression results of both the guidance law parameter and the first-order lateral time constant. To the best of our knowledge, this is the first article to identify the first-order lateral time constant in air combat.

To increase the training speed and the identification accuracy of the established model, a neural network output processing method called the improved multiple-model mechanism (IMMM) is proposed, which can be applied to general multiple-output regression problems. The proposed IMM is a discrete regression parameter identification method. The aim of discretization is to convert the network output from an absolute value to a relative value. The aim of regression is to calculate the identified regression results. In this way, we can realize a smooth mapping from an unknown range to a required range. Moreover, compared with a conventional neural network whose outputs may deviate too much from the required range and induce system instability, the neural network using the IMM can ensure that the outputs lie within a reasonable range, which increases the robustness of the system.

A proportional navigation (PN) guidance law is a popular guidance law widely used in practice because it is easy to implement [22]. In many scenarios, it is practical to assume that an incoming missile implements a PN guidance law. Thus, in this study, we focus on the identification of the guidance law parameter and the first-order lateral time constant.

The rest of this article is organized as follows. Section II presents a nonlinear dynamic model of an engagement and an analysis of the parameter identification problem. The concept of the IMM, the structure of the parameter identification model, and the dataset establishment method are presented in Section III. A comprehensive performance analysis of the proposed IMM and the established identification model is presented in Section IV. Finally, Section V concludes this article.

II. PROBLEM FORMULATION

In this section, we present a mathematical model of an engagement, followed by the analysis of parameter identification problem characteristics.

A. Nonlinear Dynamic Model

We consider a scenario in which an incoming missile pursues an aircraft in a planar Cartesian inertial reference frame, as shown in Fig. 1. Subscripts $M$ and $A$ denote variables associated with the missile and the aircraft, respectively. The position in the inertial coordinate system is denoted by $[x, y]$. The velocity, normal acceleration, and flight-path angle are denoted by $V$, $a$, and $\theta$, respectively. The missile–aircraft range and line-of-sight (LOS) angle are denoted by $R$ and $q$, respectively.

We assume that both vehicles are skid-to-turn roll stabilized. Therefore, the engagement kinematics can be expressed in polar coordinates $(R, q)$ as follows:

$$V_R = -[V_A \cos(\theta_A + q) + V_M \cos(\theta_M - q)]$$

$$\dot{q} = \frac{V_A \sin(\theta_A + q) - V_M \cos(\theta_M - q)}{R}.$$  \hspace{1cm} (1)

We also assume the first-order lateral maneuver dynamics for both the missile and the aircraft, i.e.,

$$a_i = \frac{a_{c,i} - a_i}{\tau_i}, \quad i \in A, M$$  \hspace{1cm} (2)

Fig. 1. Cartesian inertial reference frame.
where \( a_c \) is the acceleration command and \( \tau \) is the first-order lateral time constant.

The missile is assumed to employ a PN guidance law. The acceleration command \( a_c \) is

\[
a_c = N\dot{q}V_R
\]

where \( N \) is the PN guidance law parameter and \( V_R \) is the relative velocity between the missile and the aircraft.

In addition, the effect of external forces on the velocity of the missile is considered. The dynamics can be expressed as

\[
\dot{V} = (T - D)/m - g \sin \theta_M
\]

\[
D = (\rho V^2/2)C_d S_M
\]

where \( g = 9.8 \text{ m/s}^2 \) is the gravity coefficient, \( T \) is the thrust force, \( D \) is the drag force, \( m \) is the mass of the missile, \( \rho \) is the density of air, \( C_d \) is the drag coefficient, and \( S_M \) is the reference area.

B. Measurement Model

We assume that the inertial vector of the aircraft

\[
[x_A, y_A, \theta_A, a_A, V_A]^T
\]

is known with high accuracy via a navigation system [7].

The aircraft is assumed to be equipped with a radar seeker. The direct measurement data include the relative distance \( R \) and LOS angle \( q \). The measurements \( \mathbf{M}_t \) and \( t \in \mathbb{N} \) are mutually independent and can be acquired at discrete time instances \( tT_p \), where \( T_p > 0 \) is a fixed measurement period. Meanwhile, \( \mathbf{M}_t \) are assumed to be contaminated by zero-mean white Gaussian noise. Thus, the measurement model can be expressed as

\[
\mathbf{M}_t = \begin{bmatrix} R(t) \\ q(t) \end{bmatrix} + \mathbf{w}(t)
\]

\[
\mathbf{w}(t) \sim \mathcal{N}(0,\mathbf{Q}), \quad \mathbf{Q} = \text{diag}(\sigma_R^2, \sigma_q^2)
\]

where \( \sigma_R \) and \( \sigma_q \) are the measurement noise of \( R \) and \( q \), respectively.

C. Analysis of the Parameter Identification Problem

The engagement data flow is shown in Fig. 2. The guidance system on the missile outputs an acceleration command according to the data from onboard sensors. In (1) and (2), the accessible information about the aircraft includes \( V_A, \theta_A, R, R, q, \) and \( \dot{q} \). Therefore, the equations can be written as

\[
V_M \cos(\theta_M - q) = f_1(V_R, V_A, \theta_A, q)
\]

\[
V_M \sin(\theta_M - q) = f_2(\dot{q}, R, V_A, \theta_A, q)
\]

where

\[
f_1(V_R, V_A, \theta_A, q) = -V_R - V_A \cos(\theta_A + q)
\]

\[
f_2(\dot{q}, R, V_A, \theta_A, q) = -\dot{q}R + V_A \sin(\theta_A + q).
\]

Dividing (11) by (10) yields

\[
\frac{f_2(\dot{q}, R, V_A, \theta_A, q)}{f_1(V_R, V_A, \theta_A, q)} = \frac{\theta_M}{\theta_A} + q.
\]

Multiplying (10) by \( \cos(\theta_M - q) \) and multiplying (11) by \( \sin(\theta_M - q) \) yields

\[
V_M = f_1(V_R, V_A, \theta_A, q) \cos(\theta_M - q)
\]

\[
+ f_2(\dot{q}, R, V_A, \theta_A, q) \sin(\theta_M - q).
\]

The acceleration of the missile can be calculated according to data obtained from (14) and (15), as follows:

\[
a_M = \frac{V_M}{\dot{q} - \delta_t} = \frac{V_M}{\dot{\theta}_M + \cos \theta_M}.
\]

Substituting (4) into (3) yields

\[
\tau_M = \frac{NV_R \dot{q} - a_M}{\dot{a}_M}.
\]

The discrete forms of (16) and (17) are

\[
a_M = \frac{V_M}{\dot{\theta}_M,\dot{\theta}_{M-1},...} + \cos \theta_M
\]

\[
\tau_M = \frac{NV_R \dot{q} - \dot{a}_M}{\dot{a}_M}
\]

where subscripts \( t, t - 1, t - 2, ..., 1, \) and \( 0 \) are discrete time instances.

Rearranging (19) yields

\[
N = \frac{a_2 - a_1}{V_R \dot{q}_t - (a_1 - a_{t-2}) - V_{R,t-1} \dot{q}_{t-1} - (a_{t-1} - a_{t-1})}.
\]

**Remark** In theory, the analytic solutions to \( N \) and \( \tau_M \) can be directly calculated using the above equations. However, multiple differentiation will reduce the accuracy of identification results, and it is difficult to guarantee the accuracy of the required data considering the measurement noise. Therefore, the direct calculation of \( N \) and \( \tau_M \) may not work.

The identification of \( N \) and \( \tau_M \) is essentially a multiple parameter identification problem, which can be written as

\[
\left[ \tau_M, N \right] = f(R, q, \theta_A, V, v)|_{(0, t)}
\]
where \( f \) represents a complex nonlinear mapping relationship and subscript \((0, t)\) represents that the question is related to time. Thus, the key to solving the question becomes finding the mapping relationships between the kinematic information and parameters.

### III. IDENTIFICATION MODEL

In this section, we first present the concept of the IMMM, which can be applied to general multiple-output regression problems. Then, we present the structure of the parameter identification model, followed by dataset establishment.

**NOTATION**

For simplicity, in the rest of this article, we denote the sigmoid, tanh, and softmax functions as follows:

\[
\text{sigmoid}(\xi) = \frac{1}{1 + e^{-\xi}}
\]

\[
\text{tanh}(\xi) = \frac{e^\xi - e^{-\xi}}{e^\xi + e^{-\xi}}
\]

\[
\text{softmax}(\xi_i) = \frac{e^{\xi_i}}{\sum_j e^{\xi_j}}.
\]

**A. Improved Multiple-Model Mechanism**

The multiple-model mechanism (MMM) was proposed by Wang et al. [21]. The main idea of the MMM is to use regimes to represent different possible situations by connecting a multiple-model layer behind a conventional neural network. However, a conventional MMM can only be used in single-output problems. To solve this drawback, we introduce the concept of transfer learning to the MMM and propose an output processing method called the IMMM.

The proposed IMMM is a discrete regression parameter identification method. The aim of discretization is to convert the network output from an absolute value to a relative value. The aim of regression is to calculate the identified regression results. In this way, we can realize a smooth mapping from an unknown range to a required range. The main idea of the IMMM is to set several groups of regimes in a multiple-model layer, with each group corresponding to a regression result of the neural network. Regimes in different groups represent possible situations of different regression results. A certain regression result is the weighted sum of regimes in the corresponding group.

The structure of a neural network using an IMMM is shown in Fig. 3, where the symbol “S” represents the softmax activation function. There are three types of layers: an input layer, several hidden layers, and a multiple-model layer. The input layer is used to process the input data into an appropriate range. General input layer activation functions include the sigmoid and tanh functions, whose output ranges are \([0.0, 1.0]\) and \([-1.0, 1.0]\), respectively. Hidden layers are the main part of a neural network, which are composed of different kinds of neurons to extract features from samples.

The multiple-model layer is the main difference between a conventional neural network and a neural network using an IMMM. Regimes representing different possible values are established in the multiple-model layer. The outputs are discretized using the softmax function to obtain the weights of different regimes.

\[
G_i = [G_{i,1}, G_{i,2}, \ldots, G_{i,p_i}]^T = \text{softmax}(w_{out, i} h_{\text{last}} + b_{\text{out, i}})
\]

(22)

where \( G_{ij} \) is the \( j \)th regime’s weight of the \( i \)th regression result, \( p_i \) is the regime number of the \( i \)th regression result, \( w_{\text{out}} \) is the weight matrix between the last hidden layer and the \( i \)th group of regimes in the multiple-model layer, \( h_{\text{last}} \) is the output of the last hidden layer, and \( b_{\text{out}} \) is the bias vector of the \( i \)th group of regimes. There is more than one group of regimes. Each group of regimes corresponds to a regression result. The sum of a group of regime weights rather than the
sum of all weights is 1, i.e.,
\[ \sum_{j=1}^{p_i} G_{i,j} = 1. \]  

Remark 2: The connection between the last hidden layer and the multiple-model layer is separated into several groups instead of a full connection.

Regression results of the neural network using an IMMM are the weighted sum of the corresponding group of regimes
\[ O_i = \Lambda_i^T G_i = \sum_{j=1}^{p_i} \lambda_{i,j} G_{i,j} \]  
where \( O_i \) is the \( i \)th regression result and \( \lambda_{i,j} \) is the \( j \)th regime of the \( i \)th regression result.

Equations (22) and (24) realize a smoother mapping from an unknown range to a required range. Compared with a conventional neural network, the training speed and the accuracy of a neural network using an IMMM are increased because the initial output ranges of the latter are limited to a required range before training, as shown in (24). The relative weights between different regimes, rather than the absolute value of the output value, are important in the IMMM. Moreover, the use of an IMMM can ensure that the neural network outputs lie within a reasonable range, which increases the robustness of the system.

In this article, we apply the basic idea of transfer learning rather than directly using transfer learning. The basic idea of transfer learning shows that the front layers of a neural network are generally used to extract primary features, while the posterior layers are used to deeply analyze samples. When transfer learning is used to solve similar problems, generally only the last layers of a previously trained model are trained, while the front layers remain unchanged. In a network using the proposed IMMM, the multiple-model layer is the last layer of the neural network. According to the basic idea of transfer learning, we only need to differentiate parameter identification from the last layer, rather than from the hidden layer. In this way, the scale of the network can be greatly reduced to increase the training efficiency and identification performance.

B. Structure of the Parameter Identification Model

The parameter identification model established in this article is based on a GRU neural network using an IMMM. The diagram of a basic GRU neuron is shown in Fig. 4, which is composed of four parts: a reset gate \( r_t \), an update gate \( z_t \), a candidate state \( \tilde{h}_t \), and an output state \( h_t \).

The structure of the parameter identification model based on the GRU neural network using an IMMM is shown in Fig. 5, where each subscript \( t \in N \) represents a specific time; \( I_t \) is the model input data; \( G_{i,j} \) is the \( j \)th regime’s weight of the \( i \)th parameter, where \( i = 1, 2 \) denote the guidance law parameter \( N \) and the first-order lateral time constant \( \tau_{l1} \), respectively; and \( \lambda_{i,j} \) is the \( j \)th regime of the \( i \)th parameter.

The parameter identification inputs are the available kinematic information between the aircraft and the missile, including
\[ I_t = [R_t, V_{R,t}, q_t, \dot{q}_t, \theta_{A,t}, \alpha_{A,t}]^T. \]  

If the length of the measured data is uncertain. The data input to the trained network are set according to a preset input step of network \( K \) and the length of the measured data \( \kappa \)
\[ l = \min(K, \kappa). \]

The ranges of the input attributes are different. If the data are directly input into the network, attributes with a large range will have a greater influence on the network.
training, while attributes with a small range may be ignored by the network. Therefore, the measured data should be normalized into dimensionless values with the same range using min–max normalization:

\[
\zeta(i) = \frac{\zeta'(i) - I(i)_{\text{min}}}{I(i)_{\text{max}} - I(i)_{\text{min}}} \tag{36}
\]

where \(\zeta(i)\) is the processed data, \(\zeta'(i)\) is the initial data, \(I(i)_{\text{min}}\) is the minimum value of the \(i\)th input data, and \(I(i)_{\text{max}}\) is the maximum value of the \(i\)th input data.

The activation function of the input layer is tanh, which is used to compress or expand the input range to \([-1, 1]\) and increase the nonlinear characteristic of the model. Neurons in hidden layers are basic GRU neurons, whose structure and calculation equations are shown in Fig. 4 and in (25)–(28). The multiple-model layer is built according to the structure in Fig. 3. The weights of different regimes are calculated using (24). The maximum and minimum regimes of each group are set as the maximum and minimum values of the corresponding data in the training dataset, respectively. The outputs of the model are the identification results of the guidance law parameter \(N\) and the first-order lateral time constant \(\tau_M\).

C. Dataset Establishment

To increase the training efficiency, the length of all samples is the same as the preset length \(K\). The inputs are shown in (33). The labels include a PN constant \(N\) and a first-order lateral time constant \(\tau_M\).

The dataset is built based on the nonlinear dynamic model in Section II-A using Latin hypercube sampling. Compared with a Monte Carlo simulation, Latin hypercube sampling can increase the coverage of a dataset in the sample space. Samples are extracted from engagement simulations using sliding windows. Sample inputs are shown in (33), while labels are identical to the model outputs. The training database and the testing database are established separately using different trajectories, which can better verify the generalization ability of the trained network.

Note that the conditions in which the missile miss the aircraft are also included in the dataset because not all missiles can hit the aircraft in practice. Meanwhile, only data with a relative distance \(R\) greater than 300 m are used. When the relative distance between the missile and the aircraft is very small, the kinematics data, such as the LOS angular speed \(\dot{q}\) and the acceleration command \(a_c\), may change dramatically. If the data in this period are also added to the training dataset, it will cause difficulties for neural network training.

**Remark 3** Latin hypercube sampling is used twice in the process of building the dataset. It is first used to set the initial parameters of engagements and subsequently used to extract samples from the engagement simulations.
The loss function of the network is the mean square error (MSE) of batch samples

\[
L = \frac{L_N + L_s}{2}
\]  
\[
L_N = \frac{1}{n} \sum_{i=1}^{n} \frac{(\hat{N}_i - N_i)^2}{(N_{max} - N_{min})^2}
\]  
\[
L_s = \frac{1}{n} \sum_{i=1}^{n} \frac{(\hat{\tau}_{M,i} - \tau_{M,i})^2}{(\tau_{M,max} - \tau_{M,min})^2}
\]

(40)

where \( L \) is the loss value, \( i \) is the \( i \)th sample, and \( n \) is the batch size.

IV. PERFORMANCE ANALYSIS

In this section, the effectiveness of the IMMM and the performance of the established parameter identification model are verified through numerical simulations. We first present simulation parameters and engagement scenarios. Second, a comparison between training conventional identification models and identification models with an IMMM is presented. Then, we present a sample identification run and Monte Carlo simulations to demonstrate the performance of the established identification model. The identification accuracy under different drag coefficients is presented at the end of this section.

A. Simulation Parameters and Scenarios

Remark 4 For both dataset establishment and the subsequent simulations, the simulation parameters and engagement scenarios are the same as presented in this subsection.

The initial distance and LOS angle between the missile and the aircraft are \( R_0 \in [6000, 10000] \) m and \( \theta_0 \in [-30^\circ, 30^\circ] \), respectively. The aircraft performs a bang-bang maneuver during the engagement, with a first-order lateral time constant \( \tau_L = 0.6 \) s, an initial velocity angle \( \theta_L = [-30^\circ, 30^\circ] \), and a constant velocity \( V_A \in [0.8, 1.0] \) Ma, where \( \text{Ma} = 340 \) m/s is the velocity of sound. The amplitude and frequency of the bang-bang maneuver are \( \eta = 8 \) g and \( \xi = 1/8 \), respectively.

The missile is launched at \( t = 0 \), with an initial velocity \( V_{M,0} \in [2.0, 3.0] \) Ma, an initial velocity angle \( \theta_{M,0} = \theta_0 \), and a reference area \( S = 0.101 \) m². The drag coefficient of the missile is shown in Fig. 6.

The PN constant \( N \) is generally chosen between 3 and 5. In the process of network training, to prevent extreme situations, the range of labels is generally slightly larger than the required range. Thus, the guidance law parameter and the first-order lateral time constant of the missile are \( N \in [2.5, 5.5] \) and \( \tau_L \in [0.1, 0.4] \), respectively.

The measurement sampling rate of the aircraft’s radar is \( f_m = 100 \) Hz. The measurement noises are \( \sigma_R = 5 \) m and \( \sigma_\Psi = 1 \) mrad. The simulations are performed using TensorFlow-1.13.0, and the graphics card is a GeForce RTX 3090. The batch size and the number of training iterations are set as 2048 and 200,000, respectively. The initial learning rate is 0.002. The decay rate of the learning rate is 0.99/100.

B. IMM Performance

Remark 5 The sample inputs used for training are not contaminated by noise.

The comparison between training conventional identification models and identification models with an IMMM is presented in this subsection. The structure of the former models is built on the conventional model presented in [21]. The number of hidden layers is 3, with each hidden layer containing 128 basic GRU neurons. The preset input time length is 1 s. We compare the training performance of the following models:

a) conventional GRU network;

b) GRU network with an IMMM, the number of regimes \( p_1 = p_2 = 2 \);

c) GRU network with an IMMM, the number of regimes \( p_1 = p_2 = 3 \);

d) GRU network with an IMMM, the number of regimes \( p_1 = p_2 = 5 \);

e) GRU network with an IMMM, the number of regimes \( p_1 = p_2 = 11 \);

f) conventional LSTM network;

g) LSTM network with an IMMM, the number of regimes \( p_1 = p_2 = 11 \).

The regime values are shown in Table I.

The training process is presented in Fig. 7 and Table II. Note that compared with the conventional identification model, the MSE before training the model using an IMMM decreases from 0.3366 to less than 0.0834. This can be explained by the fact that the initial output ranges of the model using the IMMM are limited to the required range before training, as shown in (24). Note that compared with a conventional LSTM network, the MSE after training a conventional GRU network is only slightly reduced. At the same time, networks with an IMMM perform much better than conventional networks. This phenomenon shows that the use of an IMMM greatly improves training efficiency, validating the analysis in Section III-A.
C. Sample Run

To intuitively demonstrate the performance of the established parameter identification model, a sample example run is presented in this subsection before turning to statistical Monte Carlo simulations. The initial distance and initial LOS angle for the sample run are $R_0 = 7000$ m and $q_0 = 10^\circ$, respectively. The initial flight angle of the aircraft is $5^\circ$. The constant velocity of the aircraft is $V_A = 0.9$ Ma. The guidance law parameter and the first-order lateral time constant of the missile are $N = 3.0$ and $\tau_M = 0.20$, respectively.

The transition process of the weights of different regimes and the identification results of the model are presented in Fig. 8. The weights of different regimes are identical at the beginning of the simulation; thus, the initial identification results for $N$ and $\tau_M$ are 4.0 and 0.25, respectively. The identification results of both the guidance law parameter and the first-order lateral time constant converge after approximately 0.65 and 1 s, respectively. This is because the calculation of $\tau_M$ [as shown in (19)] has one more differential calculation than that of $N$ [as shown in (20)], and each differential calculation will increase the difficulty of identification. After 1 s, the identification results of the model fluctuate near the true value, which verifies the accuracy and stability of the parameter identification model. This is important for practical applications. As we know, the premise of estimation of an incoming missile is that the missile can be illuminated by our radar. However, in practice, an aircraft needs to maneuver to avoid missile attacks, and it may not find the incoming missile at the moment the missile is launched in practice. This means that the aircraft may find the missile and start to estimate $N$ and $\tau_M$ at any moment during the engagement. Thus, we need to guarantee the accuracy of the trained network at each data sample time. Note that in an IMMM, the weights of regimes constantly change during the simulation, while their weighted sum, i.e., the identification results, remains stable. This is different from an IMM or MMAE, in which the weight of the true situation converges to nearly 100% at the end of the simulation [7].

D. Monte Carlo Simulation

A Monte Carlo simulation with 6000 independent runs is performed to show the performance of the established identification model. In each simulation, the initial parameters are randomly set from the corresponding ranges in

---

**TABLE I**

| Number of regimes | $\Lambda_1$ | $\Lambda_2$ |
|-------------------|-------------|-------------|
| $p_1 = p_2 = 2$   | $[2.5, 5.5]^T$ | $[0.1, 0.4]^T$ |
| $p_1 = p_2 = 3$   | $[2.5, 4.0, 5.5]^T$ | $[0.1, 0.25, 0.4]^T$ |
| $p_1 = p_2 = 5$   | $[2.5, 3.25, 4.0, 4.75, 5.5]^T$ | $[0.1, 0.175, 0.25, 0.325, 0.40]^T$ |
| $p_1 = p_2 = 11$  | $[2.5, 2.8, 3.1, 3.4, 3.7, 4.0, 4.3, 4.6, 4.9, 5.2, 5.5]^T$ | $[0.10, 0.13, 0.16, 0.19, 0.22, 0.25, 0.28, 0.31, 0.34, 0.37, 0.40]^T$ |

**TABLE II**

| Model | MSE before training | MSE after training ($\times 10^{-5}$) |
|-------|---------------------|-------------------------------------|
| GRU, conventional | 0.3366 | 9.1124 |
| LSTM, conventional | 0.3412 | 9.9049 |
| GRU, $p_1 = p_2 = 2$ | 0.0824 | 5.4432 |
| GRU, $p_1 = p_2 = 3$ | 0.0813 | 6.8746 |
| GRU, $p_1 = p_2 = 5$ | 0.0834 | 5.4245 |
| GRU, $p_1 = p_2 = 11$ | 0.0828 | 6.4526 |
| LSTM, $p_1 = p_2 = 11$ | 0.0813 | 7.5095 |
Section IV-A. The samples used for the Monte Carlo simulation are extracted from trajectories that are not included in the training dataset, which can show the robustness and generalization ability of the model. The results of the simulation are presented in Fig. 9. The MSE of $N$ is $8.467 \times 10^{-5}$, while the MSE of $\tau_M$ is $1.717 \times 10^{-3}$. This phenomenon implies that the model has a better performance in identifying the guidance law parameter, also validating the analysis in Sections II-C and IV-C.

Meanwhile, the identification error decreases at the beginning and remains at a low level. At the end of the identification, the error starts to increase again. This is because, at the beginning, the outputs of the network converge, and the error decreases. At the final stage of the engagement, the relative distance between the missile and the aircraft decreases. In this case, the LOS angular speed $\dot{q}$ and the acceleration command $\alpha_M$ will become large, as shown in (2) and (4). According to Section III-C, the kinematics information in this period is not included in the training dataset. Thus, the identification error increases at the end of the engagement. Note that the network divergence at the end of the engagement does not affect the implementation of an active defense method or an evasion maneuver. First, in the early stage of the engagement, the aircraft can quickly and accurately identify the incoming missile’s parameters. Second, when the relative distance is very close, the highest level mission of the aircraft is to implement a maneuver to avoid interception. In this case, it is difficult for radar to illuminate the incoming missile, not to mention the parameter identification.

An additional Monte Carlo simulation with different $N$ and $\tau_M$ is carried out to explore the performance of the model under different situations. The PN constant $N$ is generally chosen between 3 and 5. Therefore, the guidance law parameter is

$$N \in \{3.0, 4.0, 5.0\}$$

and the first-order lateral time constants are

$$\tau_M \in \{0.10, 0.20, 0.30, 0.40\}.$$

The other parameters are randomly set according to Section IV-A. The number of trajectories of each situation is 300. We randomly extracted 50 samples from each trajectory. The performance of the parameter identification model is presented in Table III. As shown, the model has good performance under all conditions, which verifies the accuracy and robustness of the parameter identification model.
Fig. 9. Results of the Monte Carlo simulation. (a) MSE of $N$. (b) MSE of $\tau_M$. (c) Normalization MSE.

TABLE III

| MSE ($10^{-3}$) | $\tau_M$  |
|-----------------|-----------|
| $N$             | 0.10      | 0.20      | 0.30      | 0.40      |
| 3.0             | 2.1049    | 2.1683    | 2.1378    | 2.1083    |
| 4.0             | 2.0720    | 2.0066    | 2.0416    | 2.0121    |
| 5.0             | 2.1323    | 2.3942    | 2.0697    | 1.8303    |

E. Influence of the Drag Coefficient

In this subsection, we present the performance of the model under different drag coefficients $C_d$. We increase or decrease the drag coefficient $\delta_d$ times and perform a 300-run Monte Carlo simulation for each condition. The changes in the MSE are shown in Fig. 10. Although the identification accuracy of the model decreases with changes in the drag coefficient, it can still be used as a reference for the implementation of the defense method. Note that the influence of $\delta_d$ on the identification performance of $\tau_M$ is larger than that of $N$. This result occurs because the first-order lateral time constant $\tau_M$ is generally associated with the drag coefficient $C_d$, so it is more sensitive to deviations in $C_d$. 
V. CONCLUSION

In this article, a regression parameter identification model based on a gated neural network was established. The model inputs were information available between an aircraft and an incoming missile, while the outputs were regression results of the guidance law parameter and the first-order lateral time constant. Numerical simulations verified the performance of the model and showed that the guidance law parameter has a higher identification accuracy than the first-order lateral time constant.

To increase the model training speed and accuracy, the idea of transfer learning was introduced to an IMM, and an output processing method called the IMM was proposed in this article, which can be applied to general multiple-output regression problems. The main idea of the IMM was to set several groups of regimes in a multiple-model layer, with each group corresponding to a regression result of the neural network. A certain regression result was the weighted sum of regimes in the corresponding group. The proposed IMM was a discrete regression parameter identification method. The aim of discretization was to convert the output of the network from an absolute value to a relative value. The aim of regression was to calculate the identified regression results. In this way, we can realize a smooth mapping from an unknown range to a required range. Simulation results showed that compared with conventional models, the models using an IMM have a faster training speed and higher identification accuracy. Moreover, a model using an IMM can ensure that the model outputs lie within a reasonable range. Although the network is trained with the same drag curve and the identification accuracy of the model decreases with changes in the drag coefficient, it can still be used as a reference for the implementation of the defense method.

In future research, the online identification of the drag coefficient and the change in guidance laws and parameters will be considered.

REFERENCES

[1] J. Speyer, “An adaptive terminal guidance scheme based on an exponential cost criterion with application to homing missile guidance,” IEEE Trans. Autom. Control, vol. AC-21, no. 3, pp. 371–375, Jun. 1976.
[2] E. J. Ohlmeyer, “Root-mean-square miss distance of proportional navigation missile against sinusoidal target,” J. Guid., Control, Dyn., vol. 19, no. 3, pp. 563–568, 1996.
[3] E. Garcia, D. W. Casbeer, Z. E. Fuchs, and M. Pachter, “Cooperative missile guidance for active defense of air vehicles,” IEEE Trans. Aeros. Electron. Syst., vol. 54, no. 2, pp. 706–721, Apr. 2018.
[4] S. R. Kumar and D. Mukherjee, “Cooperative active aircraft protection guidance using line-of-sight approach,” IEEE Trans. Aeros. Electron. Syst., vol. 57, no. 2, pp. 957–967, Apr. 2021.
[5] Y. Wang, H. Wang, D. Lin, and W. Wang, “Nonlinear guidance laws for maneuvering target interception with virtual look angle constraint,” IEEE Trans. Aeros. Electron. Syst., vol. 58, no. 4, pp. 2807–2822, Aug. 2022.
[6] Y. Wang, H. Wang, D. Lin, and W. Wang, “Nonlinear modified bias proportional navigation guidance law against maneuvering targets,” J. Franklin Inst., vol. 359, no. 7, pp. 2949–2975, 2022.
[7] V. Shaferman and T. Shima, “Cooperative multiple-model adaptive guidance for an aircraft defending missile,” J. Guid. Control, Dyn., vol. 33, no. 6, pp. 1801–1813, 2010.
[8] X. Zou, D. Zhou, R. Du, and J. Liu, “Active defense guidance law via cooperative identification and estimation,” J. Guid. Control, Dyn., vol. 41, no. 11, pp. 2507–2512, 2018.
[9] J. Karelahti and K. Virtanen, “Adaptive controller for the avoidance of an unknown guided air combat missile,” in Proc. IEEE 46th Conf. Decis. Control, 2007, pp. 1306–1313.
[10] R. E. Kalman, “A new approach to linear filtering and prediction problems,” J. Basic Eng., vol. 82, pp. 35–45, 1960.
[11] R. E. Kalman and R. S. Bucy, “New results in linear filtering and prediction theory,” J. Basic Eng., vol. 83, pp. 95–108, 1961.
[12] F. L. Lewis, Optimal Estimation: With an Introduction to Stochastic Control Theory. Hoboken, NJ, USA: Wiley, 1986.
[13] J. Yun and C.-K. Ryoo, “Missile guidance law estimation using modified interactive multiple model filters,” J. Guid., Control, Dyn., vol. 37, no. 2, pp. 484–496, 2014.
[14] D. Magill, “Optimal adaptive estimation of sampled stochastic processes,” IEEE Trans. Autom. Control, vol. AC-10, no. 4, pp. 434–439, Oct. 1965.
[15] X.-R. Li and Y. Bar-Shalom, “Multiple-model estimation with variable structure,” IEEE Trans. Autom. Control, vol. 41, no. 4, pp. 478–493, Apr. 1996.
[16] S. Hochreiter and J. Schmidhuber, “Long short-term memory,” Neural Comput., vol. 9, no. 8, pp. 1735–1780, 1997.
[17] S. K. Ibrahim, A. Ahmed, M. A. E. Zeidan, and I. E. Ziedan, “Machine learning methods for spacecraft telemetry mining,” IEEE Trans. Aeros. Electron. Syst., vol. 55, no. 4, pp. 1816–1827, Aug. 2019.
[18] P. Liu, Y. Liu, T. Huang, Y. Lu, and X. Wang, “Decentralized automotive radar spectrum allocation to avoid mutual interference using reinforcement learning,” IEEE Trans. Aeros. Electron. Syst., vol. 57, no. 1, pp. 190–205, Feb. 2021.
[19] G. Cruz and A. Bernardino, “Learning temporal features for detection on maritime airborne video sequences using convolutional LSTM,” IEEE Trans. Geosci. Remote Sens., vol. 57, no. 9, pp. 6565–6576, Sep. 2019.
[20] Y. Wang, S. Fan, J. Wang, G. Wu, and S. He, “Fast guidance law identification approach for incoming missile based on GRU network,” Acta Aeronaut. Astronaut. Sinica, vol. 42, no. 2, 2022, Art. no. 32024.
[21] Y. Wang, S. Fan, J. Wang, and G. Wu, “Quick identification of guidance law for an incoming missile using multiple-model mechanism,” Chin. J. Aeronaut., vol. 35, no. 9, pp. 282–292, 2022.
[22] R. Fonod and T. Shima, “Wingman-based estimation and guidance for a sensorless PN-guided pursuer,” IEEE Trans. Aeros. Electron. Syst., vol. 56, no. 3, pp. 1754–1766, Jun. 2020.

Yinhan Wang received the B.Eng. degree in flight vehicle design and engineering in 2020 from the School of Aerospace Engineering, Beijing Institute of Technology, Beijing, China, where he is currently working toward the Ph.D. degree in flight vehicle design and engineering with the School of Aerospace Engineering. His research interests include guidance and control of missiles and application of artificial intelligence.
Jiang Wang received the Ph.D. degree in flight vehicle design and engineering from the Beijing Institute of Technology, Beijing, China, in 2008. He is currently a Professor with the School of Aerospace Engineering, Beijing Institute of Technology. His research interests include guidance and control of missiles and unmanned aerial vehicle formation control.

Shipeng Fan received the Ph.D. degree in flight vehicle design and engineering from the School of Aerospace Engineering, Beijing Institute of Technology, Beijing, China, in 2015. He was with the Beijing Aerospace Institute of Automatic Control, Beijing, until 2020. He is currently an Associate Professor with the School of Aerospace Engineering, Beijing Institute of Technology. His research interests include navigation, guidance and control of flight vehicles, path planning technology, and application of artificial intelligent methods.