Journal of Systems Engineering and Electronics
Vol. 35, No. 2, April 2024, pp.451 – 462

Time-varying parameters estimation with adaptive neural network EKF for missile-dual control system

YUAN Yuqi[1], ZHOU Di[1,*], LI Junlong[2], and LOU Chaofei[2]

1. School of Astronautics, Harbin Institute of Technology, Harbin 150001, China;
2. Beijing Institute of Electronic System Engineering, Beijing 100854, China

Abstract: In this paper, a filtering method is presented to estimate time-varying parameters of a missile dual control system with tail fins and reaction jets as control variables. In this method, the long-short-term memory (LSTM) neural network is nested into the extended Kalman filter (EKF) to modify the Kalman gain such that the filtering performance is improved in the presence of large model uncertainties. To avoid the unstable network output caused by the abrupt changes of system states, an adaptive correction factor is introduced to correct the network output online. In the process of training the network, a multi-gradient descent learning mode is proposed to better fit the internal state of the system, and a rolling training is used to implement an online prediction logic. Based on the Lyapunov second method, we discuss the stability of the system, the result shows that when the training error of neural network is sufficiently small, the system is asymptotically stable. With its application to the estimation of time-varying parameters of a missile dual control system, the LSTM-EKF shows better filtering performance than the EKF and adaptive EKF (AEKF) when there exist large uncertainties in the system model.

Keywords: long-short-term memory (LSTM) neural network, extended Kalman filter (EKF), rolling training, time-varying parameters estimation, missile dual control system.

DOI: 10.23919/JSEE.2024.000008

1. Introduction

The estimation of the parameters in a missile control system is still a very hot topic. It is very important to establish an accurate dynamic model in the design of missile control systems [1–3]. Since a high maneuverability is demanded for an interceptor missile, the missile must fly with a large angle of attack to create a large maneuver acceleration. Especially for dual controlled missiles with tail fins and reaction jets, when the jets work, they will produce a great impact on the motion of the missile. For dual controlled missiles which actuate large angle of attack maneuvers, the nonlinear characteristics of aerodynamics become obvious. In this case, the constant aerodynamic parameters assumption and the model linearization no longer hold. Then the estimation of time-varying aerodynamic parameters and the disturbance amplification coefficients on reaction thrust and reaction moment becomes an urgent problem for the control system design of dual controlled missiles. There are three main reasons for model uncertainty: (i) model uncertainty caused by unmodeled terms, such as external interference; (ii) model uncertainty caused by unknown parameters, such as time-varying aerodynamic parameters; (iii) linearization error caused by the linearization of a non-linear system. Therefore, even if time-varying aerodynamic parameters and disturbance amplification coefficients are considered in the establishment of the model for dual controlled missiles, there must still exist model uncertainties due to the complexity of the control system. Therefore, it is necessary to study an efficient estimation method for time-varying parameters in the presence of model uncertainties.

Kalman filter has been widely used in state estimations for linear Gaussian systems since it was proposed. For state estimations of nonlinear systems, some filters based on the modification of Kalman filtering have been proposed, such as extended Kalman filter (EKF), unscented Kalman filter (UKF), cubature Kalman filter (CKF), particle filter (PF) [4–6] and so on. Among them, EKF is widely used because of its simple structure and low computation burden. This classical nonlinear filtering is established by locally linearizing the system model around specific points. As we all know, it is a sub-optimal algorithm. When the linearization error is large and the model is uncertain, the performance of EKF will be greatly decreased. Jetto et al. [7] proposed an adaptive EKF (AEKF) to solve the problem of robot positioning. The performance of the filter is improved by adjusting the measurement and the process noise covariance matrices adaptively online. Robles et al. [8] applied the AEKF into the state prediction of aircraft with unknown aerodynamic parameters. The adaptive filter can effectively

Manuscript received August 24, 2022.
*Corresponding author.

overcome the interference caused by model uncertainty. Chen [9] proposed a method to convert the identification of time-varying parameters into the identification of constant parameter. In this method, the time-varying parameters curves are approximated by multiple polyline segments, and the identification process is completed by the least squares technique. However, there is a disadvantage in this method. To improve the accuracy of the approximation, the number of polyline segments must be increased such that the number of parameters to be estimated and the computation burden will be increased. Moreover, Chen has only discussed the identification of linear systems, rather than nonlinear systems.

When the nonlinearity, complexity and model uncertainty of the system increase, the above filters result in large estimation errors and even divergence. With the development of the neural network technology, some scholars apply neural networks to the state estimation of complex systems. Sun and Liu [10] used a chain neural network (CNN) to estimate the aerodynamic parameters of aircraft. However, this layered structure usually performs poorly in capturing the time dependencies, which are usually encountered in adaptive regression problems [11]. Actually, the capability of CNNs is quite limited in adaptive learning applications.

Recurrent neural networks (RNNs) can be used to capture the time dependency, because such networks have a feedback connection that enables them to store past information [12]. However, basic RNNs lack control structures such that long-term components cause either exponential growth or decay in the norm of gradients [13]. Thus, they are insufficient to capture long-term dependencies, which significantly restricts their performance in practical applications. In order to resolve this issue, a novel RNNs architecture with several control gates, i.e., long-short-term memory (LSTM) neural network, is introduced [14−16]. LSTM neural network achieves good performance in the time series prediction problem [17]. However, some scholars have proposed that when facing complex system processes, it is not reliable to use the neural network alone for prediction. Thus, many scholars combine the LSTM neural network with Kalman filter to preserve the physical process of the system and increase the reliability of the algorithm. Zhou et al. [18] used LSTM neural network to solve the initialization problem of Jacobian matrix in the EKF algorithm. By integrating LSTM neural network with UKF, the problem of noise in experimental data was solved [19]. Zheng et al. [20] nested LSTM neural network and EKF to study the trajectory prediction of non-cooperative vehicle. In this method, an LSTM neural network is used to modify the filter input and to compensate for the Kalman gain calculation error caused by model uncertainties. By training a large number of real trajectory data, a good prediction result is obtained. Bao et al. [21] used Kalman filter in combination with the LSTM network to estimate the bioelectricity size, and used LSTM network to fit each parameter matrix in Kalman filter algorithm to improve the filter performance. Ni et al. [22] and Hong et al. [23] used the LSTM network to replace the state equation of lithium battery system, and eventually brought it into Kalman filtering algorithm to estimate the state of lithium battery system. Zmitri et al. [24] used the LSTM network to modify the covariance matrix in the Kalman filtering algorithm to improve the filtering performance. Ni et al. [25] trained the LSTM network with input and output data of lithium battery system, and smoothed its output value by calculating the weighting averaged value of the output of the LSTM and that of Kalman filter algorithm, so as to get a more accurate estimate and reduce the burden of training the network.

In this paper, a novel AEKF is proposed for online identification of time-varying aerodynamic parameters of aircraft. The algorithm transforms the time-varying parameters identification problem into the time invariant parameters identification problem by using the Weierstrass approximation theorem in the filter design. Inspired by attention mechanism [26−28], online adjustment factor is introduced to adaptively adjust the modification ability of neural network to EKF. In the training process of neural network, multi-gradient descent training algorithm and rolling training method are proposed to realize online parameters identification, and Lyapunov second method is given to discuss the stability of the system. In the simulation analysis, the feasibility of this filtering algorithm is verified and the filtering performance is compared with EKF and Sage-Husa filter [29].

The rest of this paper is arranged as follows.

(i) In Section 2, we introduce the identification method of time-varying parameters of the missile dual control system, and make observability analysis of the designed filters.

(ii) In Section 3, the basic structure of the LSTM neural network and LSTM-EKF algorithm is introduced, and an adaptive LSTM-EKF with a correction factor is presented.

(iii) In Section 4, the method of training the LSTM-EKF is discussed. The rolling training method and the multi-gradient descent training method are introduced. After that, based on Lyapunov second method, we analyze the stability of LSTM-EKF algorithm. The results show that when the approximation error δ_t of the neural network is small enough, the system is asymptotically stable.

(iv) In Section 5, the simulation results for the application of the adaptive LSTM-EKF estimation method to the missile dual control system are provided.

(v) In Section 6, the conclusions are presented.

2. Estimation of time-varying parameters of missile dual control system

The dual control system is obviously different from the single control system. Take the dual control missile as an example. Generally, the missiles using this control strategy have strong maneuverability and a range of attack angle changes. When the thruster engines are ignited, the side direct force will directly affect the missile attitude, causing sudden changes in the flight state, which often causes significant changes in the flight aerodynamic parameters, making it difficult to identify the system parameters.

The missile body coordinate system $ox_1y_1z_1$ and the missile velocity coordinate system $ox_3y_3z_3$ are defined in Fig.1. The axis ox_1 presents the central axis of symmetry of the vehicle, and the axis ox_3 is along the velocity of the missile V. The dotted line is the projection of the velocity on the plane ox_1y_1. The axis oy_1 and axis oy_3 are all in the axisymmetric plane of the vehicle. The relationship between the two coordinate systems is determined by two angles, which are the angle of attack α and the sideslip angle β. Let ω_z, δ_z, and F_z denote the pitch rotational rate, the rudder deflection and the lateral thrust of the reaction jets, respectively. Define the aerodynamic parameters of the missile control system as $a_1 = -M_z^{\omega_z}/J_z$, $a_2(\alpha) = -M_z^{\alpha}(\alpha)/J_z$, $a_3 = -M_z^{\delta_z}\big/J_z$, and $l_z = -l/J_z$, where $M_z^{\omega_z}$ is the partial derivative of the pitching moment M_z with respect to the pitching angular rate ω_z, M_z^{α} is the partial derivative of the pitching moment M_z to the angle of attack α, $M_z^{\delta_z}$ is the partial derivative of the pitching moment M_z to the rudder deflection angle δ_z, and J_z is the component of moment of inertia on axis oz_3, l is the distance from the point of direct force to the center of mass.

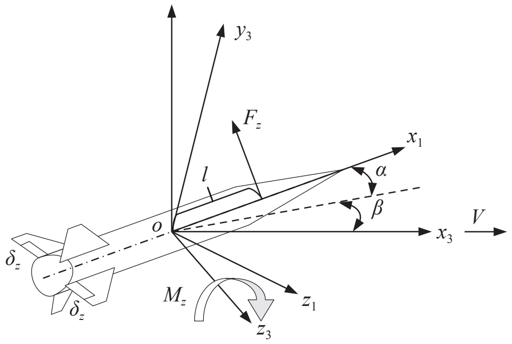


Fig. 1 Velocity coordinate system and body coordinate system

For a missile dual control system, accounting for the aerodynamic nonlinear properties, the attitude dynamic differential equation of the pitch channel can be written as

$$\dot{\omega}_z = -a_1\omega_z - a_2(\alpha)\alpha - a_3\delta_z - l_zF_z \tag{1}$$

where the aerodynamic parameter a_2 is considered as a function of α. According to the first law of Weierstrass approximation theorem, for any continuous function, a polynomial with the highest term order not greater than n can be found, and the difference between it and the continuous function is zero, i.e., the continuous function on the closed interval can be uniformly approximated by a polynomial series. Thus, the function $a_2(\alpha)$ can be written as

$$a_2(\alpha) = a_{20} + k_1\alpha + k_2\alpha^2 + \cdots \tag{2}$$

where $a_{20}, k_1, k_2, \cdots$ are unknown constants. When α changes within the critical range, this function can be approximated by a first-order polynomial with respect to α, i.e.,

$$a_2(\alpha) \approx a_{20} + k_{21}\alpha. \tag{3}$$

Substituting (3) into (1) gives the attitude dynamic equation of the pitch channel as

$$\dot{\omega}_z = -a_1\omega_z - a_{20}\alpha - k_{21}\alpha^2 - a_3\delta_z - l_zF_z. \tag{4}$$

Define a state vector $\boldsymbol{X}$ as

$$\boldsymbol{X} = \begin{bmatrix} x_1 & x_2 & x_3 & x_4 & x_5 & x_6 \end{bmatrix}^{\mathrm{T}} \tag{5}$$

where $x_1 = \omega_z$, $x_2 = a_1$, $x_3 = a_{20}$, $x_4 = k_{21}$, $x_5 = l_z$, and $x_6 = a_3$. Then, a state function is written as

$$\dot{\boldsymbol{X}} = f(\boldsymbol{X}) \tag{6}$$

where the state transition matrix

$$f(\boldsymbol{X}) = \begin{bmatrix} -x_2x_1 - x_3\alpha(t) - x_4\alpha^2(t) \\ -x_6\delta_z(t) - x_5F_{zy}(t) \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix}.$$

Since we have converted the problem to a time-invariant parameter identification problem, all states except x_1 are constant and their time derivative is 0.

The measurement variable is

$$\boldsymbol{Z}_k = \boldsymbol{h}_k\boldsymbol{X}_k + \boldsymbol{R}_k \tag{7}$$

and the measurement matrix is

$$\boldsymbol{h} = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}.$$

According to the weak observability theory of nonlinear systems, if the system has six state variables and one observation variable, the fifth order Lie derivative needs to be calculated. Now for a nonlinear system consisting of (6) and (7), the zero order Lie derivative is

$$d_0 = \boldsymbol{h}\boldsymbol{X} = x_1, \tag{8}$$

the first order Lie derivative is

$$d_1 = \frac{\partial d_0}{\partial t} + \frac{\partial d_0}{\partial \boldsymbol{X}} f(\boldsymbol{X}), \tag{9}$$

the second order Lie derivative is

$$d_2 = \frac{\partial d_1}{\partial t} + \frac{\partial d_1}{\partial \boldsymbol{X}} f(\boldsymbol{X}), \tag{10}$$

the third order Lie derivative is

$$d_3 = \frac{\partial d_2}{\partial t} + \frac{\partial d_2}{\partial \boldsymbol{X}} f(\boldsymbol{X}), \tag{11}$$

the fourth order Lie derivative is

$$d_4 = \frac{\partial d_3}{\partial t} + \frac{\partial d_3}{\partial \boldsymbol{X}} f(\boldsymbol{X}), \tag{12}$$

and the fifth order Lie derivative is

$$d_5 = \frac{\partial d_4}{\partial t} + \frac{\partial d_4}{\partial \boldsymbol{X}} f(\boldsymbol{X}). \tag{13}$$

Denote

$$\boldsymbol{\varGamma} = \begin{bmatrix} d_0 & d_1 & d_2 & d_3 & d_4 & d_5 \end{bmatrix}^{\mathrm{T}}, \tag{14}$$

then the observation matrix can be written as

$$\boldsymbol{O} = \frac{\partial \boldsymbol{\varGamma}}{\partial \boldsymbol{X}}. \tag{15}$$

According to the actual value range of each variable in the above equation, when ω_z, F_z, α, and state variable $\boldsymbol{X}$ are not be zero simultaneously, the observability matrix has a full rank, that is, the system is weakly observable.

3. Adaptive LSTM-EKF algorithm based on LSTM neural networks

3.1 LSTM neural network

The LSTM neural network shown in Fig. 2 contains three parts, i.e., a forget gate, an input gate, and an output gate. Except for the same hidden state $\boldsymbol{g}_t$ as the RNNs, it also introduces a cell state $\boldsymbol{c}_t$ for keeping the long-term memory information. At the current time t, it is to update the cell state through the forget gate and input gate data to achieve a long-term memory update. Then it is to mix the cell state $\boldsymbol{c}_t$, the last step hidden state $\boldsymbol{g}_{t-1}$ and the current step input $\boldsymbol{X}_t$ via the output gate to obtain a network output.

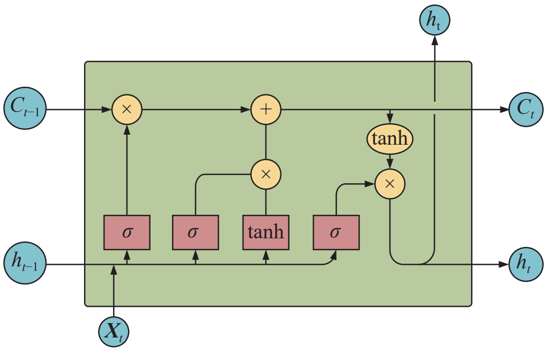


Fig. 2 LSTM neural network structure

The LSTM neural network is expressed as follows:

$$\boldsymbol{o}_t = \sigma(\mathbf{net}_{o,t}), \tag{16}$$

$$\mathbf{net}_{o,t} = \boldsymbol{W}_{og}\boldsymbol{g}_{t-1} + \boldsymbol{W}_{ox}\boldsymbol{X}_t + \boldsymbol{b}_o, \tag{17}$$

$$\boldsymbol{f}_t = \sigma(\mathbf{net}_{f,t}), \tag{18}$$

$$\mathbf{net}_{f,t} = \boldsymbol{W}_{fg}\boldsymbol{g}_{t-1} + \boldsymbol{W}_{fx}\boldsymbol{X}_t + \boldsymbol{b}_f, \tag{19}$$

$$\tilde{\boldsymbol{c}}_t = \tanh(\mathbf{net}_{o,t}), \tag{20}$$

$$\mathbf{net}_{\tilde{c},t} = \boldsymbol{W}_{cg}\boldsymbol{g}_{t-1} + \boldsymbol{W}_{cx}\boldsymbol{X}_t + \boldsymbol{b}_c, \tag{21}$$

$$\boldsymbol{i}_t = \sigma(\mathbf{net}_{i,t}), \tag{22}$$

$$\mathbf{net}_{i,t} = \boldsymbol{W}_{ig}\boldsymbol{g}_{t-1} + \boldsymbol{W}_{ix}\boldsymbol{X}_t + \boldsymbol{b}_i, \tag{23}$$

$$\boldsymbol{c}_t = \boldsymbol{f}_t \odot \boldsymbol{c}_{t-1} + \boldsymbol{i}_t \odot \tilde{\boldsymbol{c}}_t, \tag{24}$$

$$\boldsymbol{g}_t = \boldsymbol{o}_t \odot \tanh(\boldsymbol{c}_t), \tag{25}$$

where $\odot$ denotes the element-wise multiplication, $\boldsymbol{c}_t \in \mathbf{R}^{n_s}$ is the cell state vector, and $\boldsymbol{X}_t \in \mathbf{R}^{n_x}$ is the input vector. Furthermore, $\boldsymbol{i}_t$, $\boldsymbol{f}_t$ and $\boldsymbol{o}_t$ are the input, forget and output gates respectively. $\boldsymbol{b}_o$, $\boldsymbol{b}_f, \boldsymbol{b}_c$, and $\boldsymbol{b}_i$ denote the bias term of each gate. The sigmoid function $\sigma(\cdot)$ and the hyperbolic tangent function $\tanh(\cdot)$ apply point wise to the vector elements. The weight matrices are $\boldsymbol{W}_{og} \in \mathbf{R}^{n_s \times n_s}$, $\boldsymbol{W}_{ox} \in \mathbf{R}^{n_s \times n_x}$, $\boldsymbol{W}_{fg} \in \mathbf{R}^{n_s \times n_s}$, $\boldsymbol{W}_{fx} \in \mathbf{R}^{n_s \times n_x}$, $\boldsymbol{W}_{cg} \in \mathbf{R}^{n_s \times n_s}$, $\boldsymbol{W}_{cx} \in \mathbf{R}^{n_s \times n_x}$, $\boldsymbol{W}_{ig} \in \mathbf{R}^{n_s \times n_s}$, $\boldsymbol{W}_{ix} \in \mathbf{R}^{n_s \times n_x}$, $\boldsymbol{b}_o \in \mathbf{R}^{1 \times n_s}$, $\boldsymbol{b}_f \in \mathbf{R}^{1 \times n_s}$, $\boldsymbol{b}_c \in \mathbf{R}^{1 \times n_s}$, and $\boldsymbol{b}_i \in \mathbf{R}^{1 \times n_s}$.

3.2 LSTM-EKF algorithm

Consider a discrete nonlinear system:

$$\begin{cases} \boldsymbol{X}_k = f(\boldsymbol{X}_{k-1}) + \boldsymbol{Q}_k \\ \boldsymbol{Z}_k = h(\boldsymbol{X}_k) + \boldsymbol{R}_k \end{cases} \tag{26}$$

where $\boldsymbol{X}_k$ and $\boldsymbol{Z}_k$ are the state vector and measurement vector respectively. $\boldsymbol{Q}_k$ and $\boldsymbol{R}_k$ are the process noise and the measurement noise matrix respectively.

Normally, the EKF is not an optimal estimator due to the uncertainty of the model and the linearization error. In practical estimation problems, it is hard to accurately describe some complex high-order modes of the estimated system. The process noise caused by this uncertainty is usually a non Gaussian noise, and it is hard to be described by a process noise covariance matrix precisely. Moreover, in the EKF algorithm, the linearization of the model at the operation point is required for each step of estimation, which will introduce the linearization error of the model.

In order to improve the estimation accuracy of EKF and its adaptability to inaccurate models, we embed an RNN which is called gain-modified network (GMN) to deal with the unmodeled uncertainty and the linearization error of the model.

The GMN is used to identify and compensate for the influences of the unmodeled uncertainty and the linearization error of the model on the estimation. These uncertainties make bad influence on the error covariance matrix $\boldsymbol{P}_{k|k-1}$ and the Kalman gain $\boldsymbol{K}_k$. The GMN is used to modify the Kalman gain.

Define $\boldsymbol{X}_k$ as state vector, $\hat{\boldsymbol{X}}_{k-1|k-1}$ as the estimation of the last step, $\boldsymbol{P}_{k-1|k-1}$ as the error covariance matrix of the last step, and $\tilde{\boldsymbol{K}}_k$ as the output of the GMN. The GMN is embedded into EKF before the update of the state and the error covariance matrix as shown in Fig. 3.

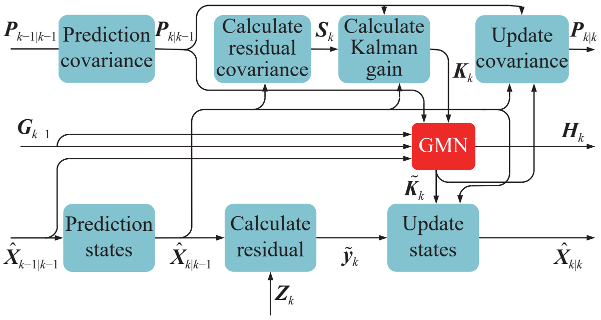


Fig. 3 LSTM-EKF algorithm structure

The process of the LSTM-EKF is as follows:

(i) Prediction of state and error covariance matrix:

$$\hat{\boldsymbol{X}}_{k|k-1} = f(\hat{\boldsymbol{X}}_{k-1|k-1}) + \boldsymbol{Q}_k, \tag{27}$$

$$\boldsymbol{P}_{k|k-1} = \boldsymbol{F}_k \boldsymbol{P}_{k-1|k-1} \boldsymbol{F}_k^{\mathrm{T}} + \boldsymbol{Q}_k. \tag{28}$$

(ii) Calculation of Kalman gain:

$$\tilde{\boldsymbol{y}}_k = z_k - \boldsymbol{h}\left(\hat{\boldsymbol{X}}_{k|k-1}\right), \tag{29}$$

$$\boldsymbol{S}_k = \boldsymbol{H}_k \boldsymbol{P}_{k|k-1} \boldsymbol{H}_k^{\mathrm{T}} + \boldsymbol{R}_k, \tag{30}$$

$$\boldsymbol{K}_k = \boldsymbol{P}_{k|k-1} \boldsymbol{H}_k^{\mathrm{T}} \boldsymbol{S}_k^{-1}, \tag{31}$$

$$(\boldsymbol{g}_k, \tilde{\boldsymbol{K}}_k) = \mathrm{GMN}(\boldsymbol{g}_{k-1}, \boldsymbol{K}_k, \hat{\boldsymbol{X}}_{k-1|k-1}, \boldsymbol{P}_{k-1|k-1}), \tag{32}$$

where $\boldsymbol{H}_k = \partial \boldsymbol{h} / \partial \boldsymbol{X}\big|_{\hat{\boldsymbol{X}}_{k|k-1}}$ is the state transition matrix.

(iii) Updates of status and error covariance matrix:

$$\hat{\boldsymbol{X}}_{k|k} = \hat{\boldsymbol{X}}_{k|k-1} + \tilde{\boldsymbol{K}}_k \tilde{\boldsymbol{y}}_k, \tag{33}$$

$$\boldsymbol{P}_{k|k} = (\boldsymbol{I} - \boldsymbol{K}_k \boldsymbol{H}_k) \boldsymbol{P}_{k|k-1}. \tag{34}$$

3.3 Adaptive LSTM-EKF algorithms

Although the LSTM neural network has a good ability to predict time series, it will be difficult for neural network to accurately fit the dynamics of a system with abrupt changes. For example, for the missile with dual control system, when the reaction jets are ignited, the flight state of the missile will have a sudden change, which will lead to a declining of the network training estimation. In order to deal with this problem, an adaptive correction factor is introduced to adjust the network output. The factor is determined by the network output error.

Define $\boldsymbol{e}$ as the error of the network, i.e.,

$$\boldsymbol{e}_k = \tilde{\boldsymbol{K}}_k - \boldsymbol{K}_k. \tag{35}$$

In order to eliminate the network output error caused by the declining of network fitting ability caused by the abrupt change, a correction factor ε_k is introduced, and its calculation rule is defined as

$$\varepsilon_k = \frac{|\boldsymbol{e}_k|}{\left||\boldsymbol{e}_k| + \tilde{\boldsymbol{K}}_k\right|}. \tag{36}$$

A network output correction law is designed as

$$\hat{\tilde{\boldsymbol{K}}}_k = (1 - \varepsilon_k)\tilde{\boldsymbol{K}}_k + \varepsilon_k \boldsymbol{K}_k. \tag{37}$$

According to (36), when the error of LSTM network increases, the value of ε_k also increases. More precisely, we can adaptively adjust the modification ability of the neural network through the training error of it. As we all know, when the filter tends to converge, the value of filter gain is very small. In this case, if there is a large fluctuation in the network input, it is obvious that the output value of the network will also fluctuate, which is very detrimental to the filtering performance. By introducing the correction factor ε_k in (36), the value of the adjustment factor can be increased when the filter tends to converge, thereby reducing the modification effect of the neural network on the filter.

Thus, the network output can be adjusted and corrected adaptively according to the network output error.

4. Training method of LSTM-EKF

4.1 Rolling training method

Although the LSTM neural network can be employed in a long-term dependent series estimation, when the length of the target series is too large, the burden of one-step training is great, which is a disadvantage of this algorithm. To decrease the computational burden, a rolling learning method has been proposed, where a sliding window [30] is used to segment a long-term series prediction into several short-term series predictions, as shown in Fig. 4.

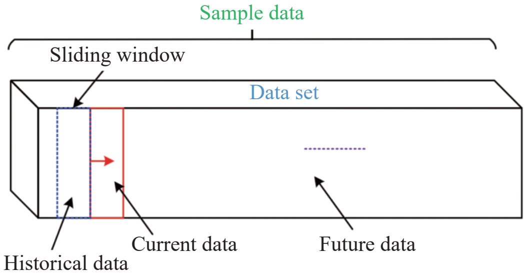


Fig. 4 Roll training method

The rolling training function is as follows:

$$\boldsymbol{E}_p^L=\begin{bmatrix}\boldsymbol{E}_k\\ \boldsymbol{E}_{k-1}\\ \vdots\\ \boldsymbol{E}_{k-p+1}\end{bmatrix}=\begin{bmatrix}\boldsymbol{y}_k^d-\boldsymbol{y}_k\\ \boldsymbol{y}_{k-1}^d-\boldsymbol{y}_{k-1}\\ \vdots\\ \boldsymbol{y}_{k-p+1}^d-\boldsymbol{y}_{k-p+1}\end{bmatrix} \tag{38}$$

where p presents the length of sliding window, $\boldsymbol{E}_k$ is the error matrix at the time step k, $\boldsymbol{y}_k^d$ and $\boldsymbol{y}_k$ denote the desired value and networks output value at step k respectively.

4.2 Multi-gradient descent training method

In the optimization problem with multiple constraints, it is difficult to completely describe the system dynamics by using a single input-output error gradient descent method. In the training process, the training error between the output value and the real value of the filter is a target constraint, but in the iterative process of the Kalman filter, the Kalman gain also plays an important role in describing the system dynamics. Therefore, in this system, the difference between the real value of the Kalman gain and the output value of the gain modified network is taken as the second error. Because the network errors are all about their functions, in the actual training process, the sum of squares of error $\boldsymbol{E}_1$ and $\boldsymbol{E}_2$ are compared, and then the larger one is selected as the error value obtained by gradient.

Define a loss function $\boldsymbol{E}$ as

$$\boldsymbol{E}=\begin{cases}\boldsymbol{E}_1, & |\boldsymbol{E}_1|\geqslant|\boldsymbol{E}_2|\\ \boldsymbol{E}_2, & |\boldsymbol{E}_1|<|\boldsymbol{E}_2|\end{cases} \tag{39}$$

where

$$\boldsymbol{E}_1=\sum_{i=1}^{n}(\hat{\boldsymbol{X}}_{k|k,i}-\boldsymbol{X}_i)^2, \tag{40}$$

$$\boldsymbol{E}_2=\sum_{k=1}^{j}(\tilde{\boldsymbol{K}}_k-\boldsymbol{K}_k)^2. \tag{41}$$

When $\boldsymbol{E}=\boldsymbol{E}_1$, the gradients of the loss function with respect to the weights of the LSTM network $\boldsymbol{W}_{og}$, $\boldsymbol{W}_{fg}$, $\boldsymbol{W}_{cg}$, $\boldsymbol{W}_{ig}$ are given by

$$\frac{\partial\boldsymbol{E}}{\partial\boldsymbol{W}_{og}}=\frac{\partial\boldsymbol{E}}{\partial\mathbf{net}_{o,t}}\cdot\frac{\partial\mathbf{net}_{o,t}}{\partial\boldsymbol{W}_{og}}, \tag{42}$$

$$\frac{\partial\boldsymbol{E}}{\partial\boldsymbol{X}_t}=\operatorname{diag}\left[2(\hat{\boldsymbol{X}}_{k|k,i}-\boldsymbol{X}_i)\right], \tag{43}$$

$$\frac{\partial\boldsymbol{X}_t}{\partial\boldsymbol{g}_t}=\operatorname{diag}\left[\boldsymbol{z}_{k,i}-h\left(\hat{\boldsymbol{X}}_{k|k-1,i}\right)\right], \tag{44}$$

$$\frac{\partial\boldsymbol{g}_t}{\partial\boldsymbol{o}_t}=\frac{\partial\boldsymbol{g}_t}{\partial\boldsymbol{f}_t}=\frac{\partial\boldsymbol{g}_t}{\partial\boldsymbol{i}_t}=\operatorname{diag}[\tanh(\boldsymbol{c}_t)], \tag{45}$$

$$\frac{\partial\boldsymbol{g}_t}{\partial\boldsymbol{c}_t}=\operatorname{diag}[\boldsymbol{o}_t\odot(\boldsymbol{I}-\tanh(\boldsymbol{c}_t)^2)], \tag{46}$$

$$\frac{\partial\boldsymbol{o}_t}{\partial\mathbf{net}_{o,t}}=\operatorname{diag}[\boldsymbol{o}_t\odot(\boldsymbol{I}-\boldsymbol{o}_t)], \tag{47}$$

$$\frac{\partial\boldsymbol{f}_t}{\partial\mathbf{net}_{f,t}}=\operatorname{diag}[\boldsymbol{f}_t\odot(\boldsymbol{I}-\boldsymbol{f}_t)], \tag{48}$$

$$\frac{\partial\boldsymbol{i}_t}{\partial\mathbf{net}_{i,t}}=\operatorname{diag}[\boldsymbol{i}_t\odot(\boldsymbol{I}-\boldsymbol{i}_t)], \tag{49}$$

$$\frac{\partial\tilde{\boldsymbol{c}}_t}{\partial\mathbf{net}_{\tilde{c},t}}=\operatorname{diag}[\boldsymbol{I}-\tilde{\boldsymbol{c}}_t^2], \tag{50}$$

$$\frac{\partial\mathbf{net}_{o,t}}{\partial\boldsymbol{W}_{og}}=\frac{\partial\mathbf{net}_{f,t}}{\partial\boldsymbol{W}_{fg}}=\frac{\partial\mathbf{net}_{i,t}}{\partial\boldsymbol{W}_{ig}}=\frac{\partial\mathbf{net}_{\tilde{c},t}}{\partial\boldsymbol{W}_{og}}=\boldsymbol{g}_{t-1}, \tag{51}$$

$$\frac{\partial\boldsymbol{E}}{\partial\boldsymbol{W}_{fg}}=\frac{\partial\boldsymbol{E}}{\partial\mathbf{net}_{f,t}}\cdot\frac{\partial\mathbf{net}_{f,t}}{\partial\boldsymbol{W}_{fg}}, \tag{52}$$

$$\frac{\partial\boldsymbol{E}}{\partial\boldsymbol{W}_{cg}}=\frac{\partial\boldsymbol{E}}{\partial\mathbf{net}_{c,t}}\cdot\frac{\partial\mathbf{net}_{o,t}}{\partial\boldsymbol{W}_{cg}}, \tag{53}$$

$$\frac{\partial\boldsymbol{E}}{\partial\boldsymbol{W}_{ig}}=\frac{\partial\boldsymbol{E}}{\partial\mathbf{net}_{i,t}}\cdot\frac{\partial\mathbf{net}_{o,t}}{\partial\boldsymbol{W}_{ig}}. \tag{54}$$

According to the chain rule, it follows that

$$\frac{\partial\boldsymbol{E}}{\partial\mathbf{net}_{o,t}}=\frac{\partial\boldsymbol{E}}{\partial\boldsymbol{X}_t}\cdot\frac{\partial\boldsymbol{X}_t}{\partial\boldsymbol{g}_t}\cdot\frac{\partial\boldsymbol{g}_t}{\partial\boldsymbol{o}_t}\cdot\frac{\partial\boldsymbol{o}_t}{\partial\mathbf{net}_{o,t}}, \tag{55}$$

$$\frac{\partial\boldsymbol{E}}{\partial\mathbf{net}_{f,t}}=\frac{\partial\boldsymbol{E}}{\partial\boldsymbol{X}_t}\cdot\frac{\partial\boldsymbol{X}_t}{\partial\boldsymbol{g}_t}\cdot\frac{\partial\boldsymbol{g}_t}{\partial\boldsymbol{f}_t}\cdot\frac{\partial\boldsymbol{f}_t}{\partial\mathbf{net}_{f,t}}, \tag{56}$$

$$\frac{\partial\boldsymbol{E}}{\partial\mathbf{net}_{\tilde{c},t}}=\frac{\partial\boldsymbol{E}}{\partial\boldsymbol{E}_t}\cdot\frac{\partial\boldsymbol{X}_t}{\partial\boldsymbol{g}_t}\cdot\frac{\partial\boldsymbol{g}_t}{\partial\boldsymbol{c}_t}\cdot\frac{\partial\boldsymbol{c}_t}{\partial\mathbf{net}_{\tilde{c},t}}, \tag{57}$$

$$\frac{\partial\boldsymbol{E}}{\partial\mathbf{net}_{i,t}}=\frac{\partial\boldsymbol{E}}{\partial\boldsymbol{X}_t}\cdot\frac{\partial\boldsymbol{X}_t}{\partial\boldsymbol{g}_t}\cdot\frac{\partial\boldsymbol{g}_t}{\partial\boldsymbol{i}_t}\cdot\frac{\partial\boldsymbol{i}_t}{\partial\mathbf{net}_{i,t}}. \tag{58}$$

Substituting (55) through (51) into (42) yields:

$$\frac{\partial \boldsymbol{E}}{\partial \boldsymbol{W}_{og,i}} = 2(\hat{\boldsymbol{X}}_{k|k,i} - \boldsymbol{X}_i)\cdot(\boldsymbol{z}_{k,i} - h\left(\hat{\boldsymbol{X}}_{k|k-1,i}\right))\cdot(\mathrm{diag}[\tanh(\boldsymbol{c}_t)])\cdot$$
$$(\mathrm{diag}[\boldsymbol{o}_t \odot (\boldsymbol{I} - \boldsymbol{o}_t)])\boldsymbol{g}_{t-1}. \tag{59}$$

In the same way, substituting (55) through (51) into (52) through (54), the gradients of the loss function with respect to the weights of the LSTM network $\boldsymbol{W}_{fg}$, $\boldsymbol{W}_{cg}$, $\boldsymbol{W}_{ig}$, $\boldsymbol{b}_o$, $\boldsymbol{b}_f$, $\boldsymbol{b}_c$, $\boldsymbol{b}_i$ can be obtained.

For the gradients of the loss function with respect to weights $\boldsymbol{W}_{ox}$, $\boldsymbol{W}_{fx}$, $\boldsymbol{W}_{cx}$, $\boldsymbol{W}_{ix}$, we replace the second term of (52) through (54) with $\partial \mathbf{net}_{(\cdot),t} / \partial \boldsymbol{W}_{(\cdot)x} = \boldsymbol{X}_t$, so that, we can obtain the formula of $\partial \boldsymbol{E} / \partial \boldsymbol{W}_{(\cdot)x}$, also the same way for $\partial \boldsymbol{E} / \partial \boldsymbol{b}_{(\cdot)}$.

When $\boldsymbol{E} = \boldsymbol{E}_2$, the training method of the LSTM-EKF is the same as the traditional method, so it will not be repeated. Up to now, the weight updating formulas of the learning EKF has been obtained.

4.3 LSTM neural network training stability analysis

To determine the stability of LSTM neural network training, it is essential to determine whether the output values of the network $\boldsymbol{y}_t$ approximate the nominal values $\boldsymbol{y}_d$ after training. Let's define the loss function $\boldsymbol{\delta}_t = \boldsymbol{K}_t - \hat{\boldsymbol{K}}_t$, state error function $\boldsymbol{E}_t = \boldsymbol{X}_t - \hat{\boldsymbol{X}}_t$. Reference to the neural network theory, the network output error is bounded after enough training iterations. Thus there is $\|\boldsymbol{\delta}_t\| \leqslant \sigma$. Based on the Lyapunov second method, designing the Lyapunov function:

$$V = \boldsymbol{E}_t^{\mathrm{T}} \boldsymbol{P}_t^{-1} \boldsymbol{E}_t. \tag{60}$$

Obviously (60) is positive definite.

Define $\Delta V = V_{t+1} - V_t$, so that

$$\Delta V = \boldsymbol{E}_{t+1}^{\mathrm{T}} \boldsymbol{P}_{t+1}^{-1} \boldsymbol{E}_{t+1} - \boldsymbol{E}_t^{\mathrm{T}} \boldsymbol{P}_t^{-1} \boldsymbol{E}_t \leqslant$$
$$\boldsymbol{E}_{t+1}^{\mathrm{T}} (\boldsymbol{P}_{t+1} - q\boldsymbol{I})^{-1} \boldsymbol{E}_{t+1} - \boldsymbol{E}_t^{\mathrm{T}} \boldsymbol{P}_t^{-1} \boldsymbol{E}_t. \tag{61}$$

According to (33) and (34), we can obtain

$$\boldsymbol{E}_{t+1} = \left(\boldsymbol{I} - \hat{\boldsymbol{K}}_t \boldsymbol{H}_t\right) \boldsymbol{E}_{t-1}, \tag{62}$$

$$\boldsymbol{P}_{t+1} - q\boldsymbol{I} = \left(\boldsymbol{I} - \hat{\boldsymbol{K}}_t \boldsymbol{H}_t\right) \boldsymbol{P}_t, \tag{63}$$

where $\boldsymbol{Q}_t$ is the process noise matrix and $\boldsymbol{Q}_t = q\boldsymbol{I}$. Also (62) can be written as

$$\boldsymbol{E}_{t+1} - \boldsymbol{E}_t = -\hat{\boldsymbol{K}}_t \boldsymbol{H}_t \boldsymbol{E}_t. \tag{64}$$

According to (62) and (63), we can obtain

$$(\boldsymbol{P}_{t+1} - q\boldsymbol{I})^{-1} \boldsymbol{E}_{t+1} = \boldsymbol{P}_t^{-1} \boldsymbol{E}_t. \tag{65}$$

Substituting (65) into (61), there is

$$\Delta V \leqslant \boldsymbol{E}_{t+1}^{\mathrm{T}} \boldsymbol{P}_t^{-1} \boldsymbol{E}_{t+1} - \boldsymbol{E}_t^{\mathrm{T}} \boldsymbol{P}_t^{-1} \boldsymbol{E}_t \leqslant$$
$$(\boldsymbol{E}_{t+1} - \boldsymbol{E}_t)^{\mathrm{T}} \boldsymbol{P}_t^{-1} \boldsymbol{E}_t. \tag{66}$$

Substituting (64) into (66), there is

$$\Delta V \leqslant -\boldsymbol{E}_t^{\mathrm{T}} \boldsymbol{H}_t^{\mathrm{T}} \hat{\boldsymbol{K}}_t^{\mathrm{T}} \boldsymbol{P}_t^{-1} \boldsymbol{E}_t. \tag{67}$$

According to the formula of EKF, we can obtain

$$\boldsymbol{E}_t = \boldsymbol{X}_t - \hat{\boldsymbol{X}}_t = \boldsymbol{X}_{t-1} + \boldsymbol{K}_t \boldsymbol{Z}_t - \left(\boldsymbol{X}_{t-1} + \hat{\boldsymbol{K}}_t \boldsymbol{Z}_t\right) =$$
$$\boldsymbol{\delta}_t \boldsymbol{Z}_t \leqslant \sigma \boldsymbol{Z}_t. \tag{68}$$

Substituting (68) into (67), there is

$$\Delta V \leqslant -\sigma^2 \boldsymbol{Z}_t^{\mathrm{T}} \boldsymbol{H}_t^{\mathrm{T}} \hat{\boldsymbol{K}}_t^{\mathrm{T}} \boldsymbol{P}_t^{-1} \boldsymbol{Z}_t \leqslant$$
$$-\sigma^2 \boldsymbol{Z}_t^{\mathrm{T}} \boldsymbol{H}_t^{\mathrm{T}} \left(\boldsymbol{K}_t^{\mathrm{T}} - \boldsymbol{\delta}_t^{\mathrm{T}}\right) \boldsymbol{P}_t^{-1} \boldsymbol{Z}_t \leqslant$$
$$-\sigma^2 \begin{pmatrix} \boldsymbol{Z}_t^{\mathrm{T}} \boldsymbol{H}_t^{\mathrm{T}} \boldsymbol{K}_t^{\mathrm{T}} \boldsymbol{P}_t^{-1} \boldsymbol{Z}_t \\ -\boldsymbol{Z}_t^{\mathrm{T}} \boldsymbol{H}_t^{\mathrm{T}} \boldsymbol{\delta}_t^{\mathrm{T}} \boldsymbol{P}_t^{-1} \boldsymbol{Z}_t \end{pmatrix}. \tag{69}$$

Let's introduce $\boldsymbol{M}_t = \boldsymbol{H}_t \boldsymbol{P}_t \boldsymbol{H}_t^{\mathrm{T}} + r\boldsymbol{I}$, so $\boldsymbol{K}_t = \boldsymbol{P}_t \boldsymbol{H}_t^{\mathrm{T}} \boldsymbol{M}_t^{-1}$, $\boldsymbol{R}_t = r\boldsymbol{I}$ is the measurement matrix. Obviously, the matrix $\boldsymbol{M}_t$ and $\boldsymbol{M}_t^{-1}$ are axially symmetric:

$$\Delta V \leqslant -\sigma^2 \begin{pmatrix} \boldsymbol{Z}_t^{\mathrm{T}} \boldsymbol{H}_t^{\mathrm{T}} \boldsymbol{M}_t^{-1} \boldsymbol{H}_t \boldsymbol{Z}_t \\ -\boldsymbol{Z}_t^{\mathrm{T}} \boldsymbol{H}_t^{\mathrm{T}} \boldsymbol{\delta}_t^{\mathrm{T}} \boldsymbol{P}_t^{-1} \boldsymbol{Z}_t \end{pmatrix}. \tag{70}$$

Because $\boldsymbol{M}_t = \boldsymbol{H}_t \boldsymbol{P}_t \boldsymbol{H}_t^{\mathrm{T}} + r\boldsymbol{I}$, $\boldsymbol{M}_t^{-1} \leqslant r\boldsymbol{I}$, (70) can be rewritten as

$$\Delta V \leqslant -\sigma^2 \begin{pmatrix} \boldsymbol{Z}_t^{\mathrm{T}} \boldsymbol{H}_t^{\mathrm{T}} r \boldsymbol{H}_t \boldsymbol{Z}_t \\ -\boldsymbol{Z}_t^{\mathrm{T}} \boldsymbol{H}_t^{\mathrm{T}} \boldsymbol{\delta}_t^{\mathrm{T}} \boldsymbol{P}_t^{-1} \boldsymbol{Z}_t \end{pmatrix} \leqslant -\sigma^2 \boldsymbol{Z}_t^{\mathrm{T}} \boldsymbol{H}_t^{\mathrm{T}} \left(r\boldsymbol{H}_t - \boldsymbol{\delta}_t^{\mathrm{T}} \boldsymbol{P}_t^{-1}\right) \boldsymbol{Z}_t. \tag{71}$$

To ensure that ΔV is negative definite or negative semi-definite, we should make that the matrix $\boldsymbol{H}_t^{\mathrm{T}}(r\boldsymbol{H}_t - \boldsymbol{\delta}_t^{\mathrm{T}} \boldsymbol{P}_t^{-1})$ is positive definite or positive semi-definite.

According to (7), the matrix $\boldsymbol{H}_t^{\mathrm{T}}(r\boldsymbol{H}_t - \boldsymbol{\delta}_t^{\mathrm{T}} \boldsymbol{P}_t^{-1})$ can be rewritten as

$$\boldsymbol{H}_t^{\mathrm{T}} \left(r\boldsymbol{H}_t - \boldsymbol{\delta}_t^{\mathrm{T}} \boldsymbol{P}_t^{-1}\right) = \begin{bmatrix} r - \boldsymbol{\delta}_t^{\mathrm{T}} \boldsymbol{P}_{t,1}^{-1} & \boldsymbol{\delta}_t^{\mathrm{T}} \boldsymbol{P}_{t,2}^{-1} & \ldots & \boldsymbol{\delta}_t^{\mathrm{T}} \boldsymbol{P}_{t,i}^{-1} \\ 0 & 0 & \ldots & 0 \\ \vdots & \vdots & & \vdots \\ 0 & 0 & \cdots & 0 \end{bmatrix} \tag{72}$$

where $\boldsymbol{P}_{t,i}^{-1}$ presents the ith column of the matrix $\boldsymbol{P}_t^{-1}$. Obviously, except for the first order principal minor, the other principal minors are all zero matrix. Therefore, as long as the first order principal minor is bigger than zero, the matrix is semi positive definite. Therefore, the problem is transformed into ensuring that the inequality is satisfied as

$$r - \boldsymbol{\delta}_t^{\mathrm{T}} \boldsymbol{P}_{t,1}^{-1} > 0. \tag{73}$$

According to the previous definition, $\boldsymbol{\delta}_t$ is the approximation error of the LSTM neural network. Through enough training iterations, the approximation error $\boldsymbol{\delta}_t$ is bounded. When the value of $\boldsymbol{\delta}_t$ satisfies inequality (73), ΔV is positive semi-definite, V is positive definite, so the system is asymptotically stable.

5. Simulations

In the simulation process, the LSTM-EKF algorithm proposed in this paper is compared with the traditional EKF algorithm and the Sage-Husa filter algorithm with adaptive process noise. The results will verify the modification of the LSTM neural network to Kalman gain and the adaptability of the adaptive LSTM-EKF to model uncertainties.

In simulations, the true values of the aerodynamic parameters are given by $a_1 = 0.075$, $a_3 = 24.5$, and the true value of the reaction jets control parameter is given by $l_z = 0.05$. Based on previous experimental experiences, the aerodynamic parameters a_2 increase with the rising of α, and when α changes from 0° to 30°, the value of a_2 rises to two times of its initial value. Thus, the true values of the parameters about a_2 are set to be $a_{20} = 11.5$ and $k_{21} = 14$. In order to stimulate the observability of the control system, let the missile pitch angle track a command ϑ_c containing a cos signal and a ramp signal, i.e.,

$$\vartheta_c = p_1 \cos(p_2 t) + p_3 t \tag{74}$$

where p_1, p_2, and p_3 are adjustable parameters.

In order to estimate l_z which is a parameter about reaction jets control, three lateral thrust pulses are actuated on the missile. The amplitude of each pulse is 200 N and the ignition duration of each pulse is 4 ms, the interval between two adjacent pulses is 36 ms, and the first pulse is output after 1 s from the start of the simulation.

During the network training process, missile flight data are used to train the network with a total time of 10 s and a single step interval of t=0.001 s, that is, the total number of training samples is 10 000. Set the rolling training sliding window size n=20. Thus in this simulation, the total number of sliding windows is 10 000. The training data of the neural network uses a set of flight data generated by computer simulation for an aircraft. The aircraft dynamics and kinematics model is referenced in [31]. The cost functions that evaluate the training performance of the neural network are defined as

$$\text{Error\_Cost}_1 = \boldsymbol{E}_1^{\mathrm{T}} \cdot \boldsymbol{E}_1, \tag{75}$$

$$\text{Error\_Cost}_2 = \boldsymbol{E}_2^{\mathrm{T}} \cdot \boldsymbol{E}_2. \tag{76}$$

The variations of α, and the pulse are shown in Fig. 5 and Fig. 6.

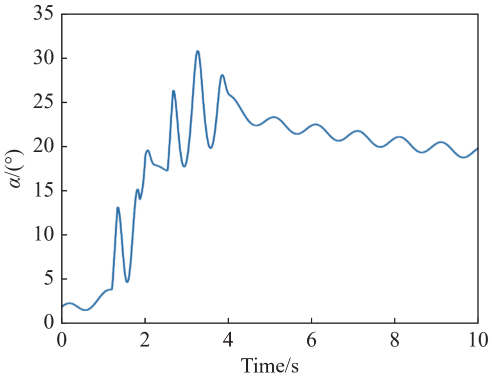


Fig. 5 Variation of angle of attack

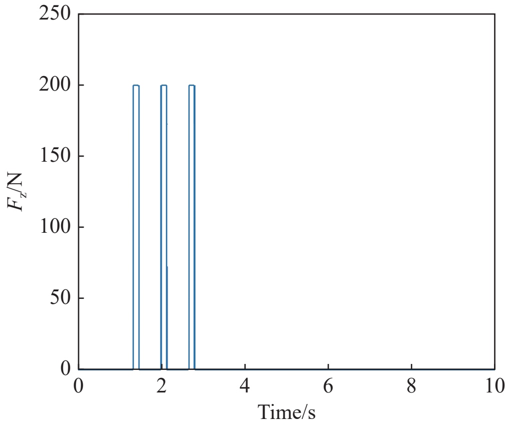
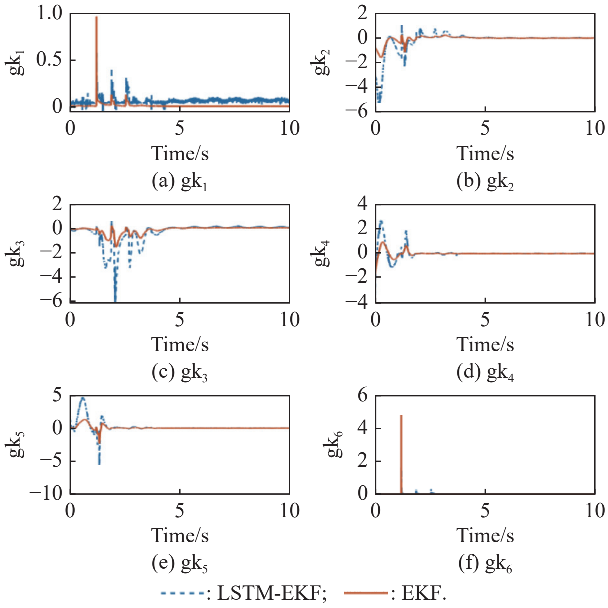


Fig. 6 Pulse engine ignition curve

Fig. 5 shows the variation of α and Fig. 6 shows the ignition of the pulse engine. It can be seen from the figure that α is inclined and sinusoidal with the given amount of control signal as shown in (74). The reason for using the above control signal is to improve the filtering performance, and the corresponding relationship between α with the aerodynamic parameters a_2 mentioned above is established only when α is positive, which guarantees that α is positive in the whole flight process. The pulse engine control force shown in Fig. 6 is introduced to identify the direct force parameters l_z. Since the control here is open-loop only for the purpose of identifying the parameters, it can also be seen from Fig. 5 that α changes dramatically under the action of the direct force.

Fig. 7 shows the comparison of each channel of the LSTM-EKF gain and the EKF gain. Where, gk_1–gk_6 present the elements of the Kalman gain vector. The Kalman filter principle shows that when the model uncertainty increases, the accuracy of the state predicted by the model decreases, and accordingly, the measurement information is more reliable. From Fig 7, it can be seen that the filter gain of LSTM-EKF under the action of LSTM neural network is significantly higher than that of traditional EKF, especially in the state channel without measurement information. This shows that LSTM neural network has significantly modified the filter gain, and the same results can be obtained from the following simulation results.



(a) gk_1 (b) gk_2 (c) gk_3 (d) gk_4 (e) gk_5 (f) gk_6

Fig. 7 Kalman gain of EKF and LSTM-EKF

Figs.8–13 show the estimation of each state of the filter. From the filter design, it is known that the pitch rotational rate ω_z of channel 1 can be measured, so the model uncertainty has little influence on this state estimation. However, it can be seen in the magnified image that the LSTM-EKF performance is slightly better than that of EKF and AEKF. Fig.s 9–13 are the estimations of other states without measurements, so the performance of these state estimation is strongly influenced by model uncertainty. It can be seen from the figures that LSTM-EKF is superior to traditional EKF and AEKF in both convergence speed and filter estimation accuracy. This shows that LSTM-EKF has obvious superiority in the presence of large model uncertainty.

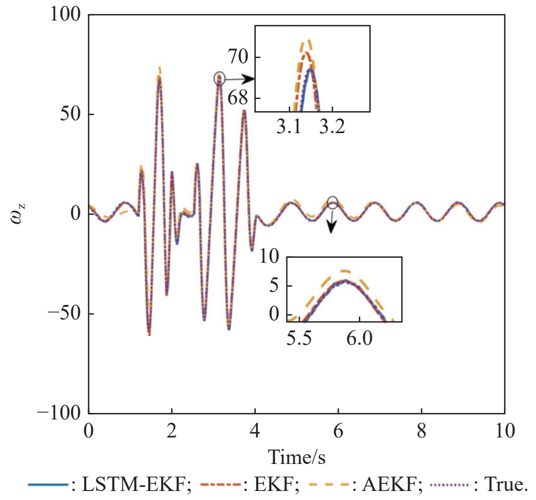


Fig. 8 Estimation of pitch angular rate

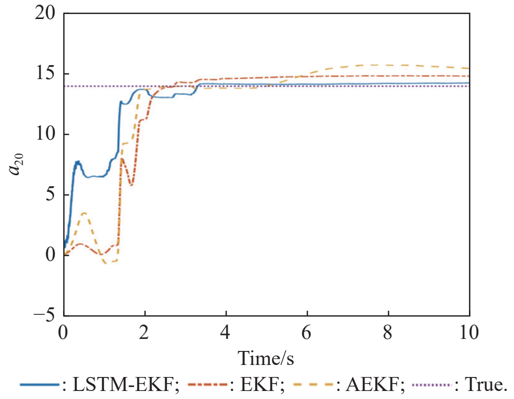


Fig. 9 Estimation of a_{20}

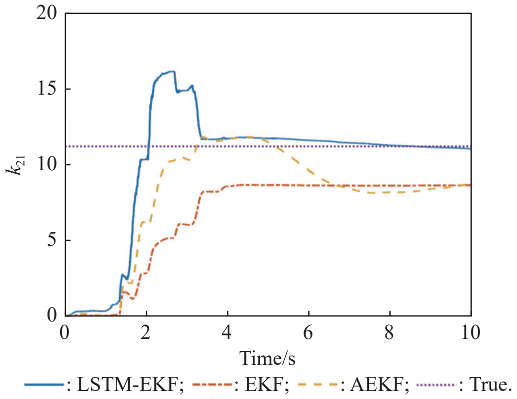


Fig. 10 Estimation of k_{21}

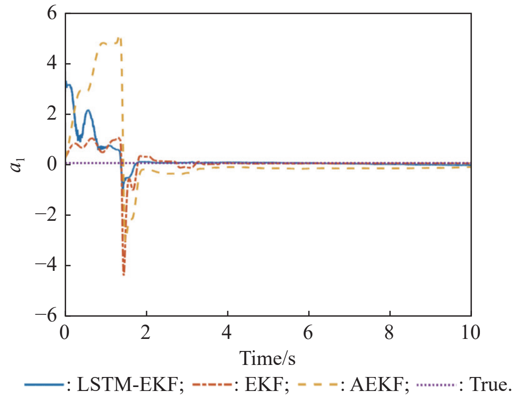


Fig. 11 Estimation of a_1

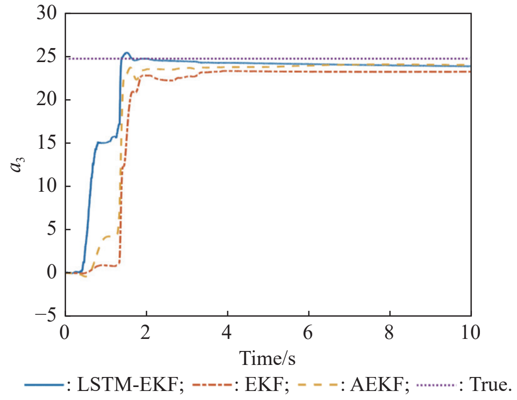


Fig. 12 Estimation of a_3

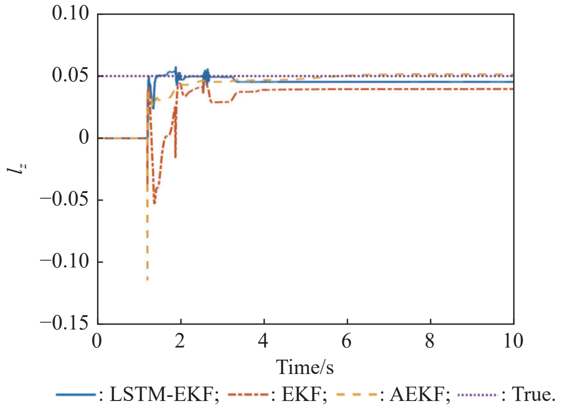


Fig. 13 Estimation of l_z

Fig 14 shows the estimation of time-varying parameters a_2. Equation (3) shows that the aerodynamic parameter a_2 is a function of α when α is within the operating range of the aircraft. By polynomial approximation, the problem of estimating time-varying parameter is reduced to that of estimating several time invariant parameters. Therefore, the estimation of time-varying parameter depends on the estimation of the constant parameters a_{20} and k_{21}. As can be seen from Fig.14, since LSTM-EKF is better than EKF and AEKF in estimation accuracy, the estimation performance of a_2 is also better than EKF and AEKF, and LSTM-EKF has a faster convergence rate.

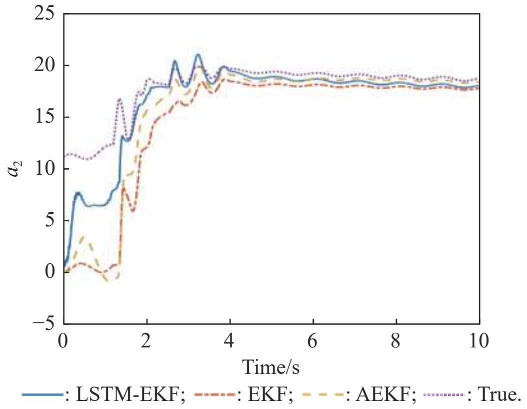
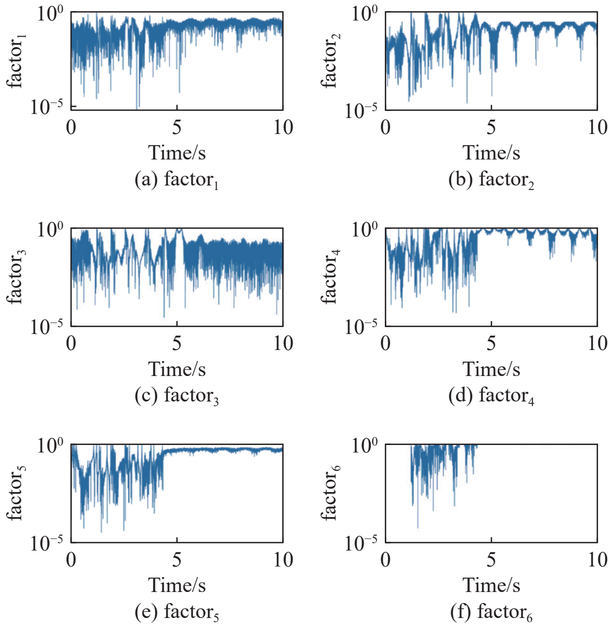


Fig. 14 Estimation of a_2

Fig.15 shows the variation of the correction factors ε for each channel. There are dynamic and steady processes in the working process of the filter. When the filter is in the dynamic process, the correction factor value is smaller, so the modified value of LSTM neural network is trusted more, which improves the convergence rate of the filter. When the pulse engines are ignited, the value of the correction factor increases significantly, because the direct force action is abrupt. In order to suppress the influence of the abrupt state on the system, the correction factor should be increased appropriately, so as to improve the system stability and filter performance. When the filter enters the steady state, the gain of each channel tends to zero, and the correction factor is large, even to 1, thus avoiding unnecessary steady-state error. Especially in channel 6, when the pulse engine is not ignited, the filter is in steady state, as can be seen from Fig. 15, at which time the correction factor is 1.



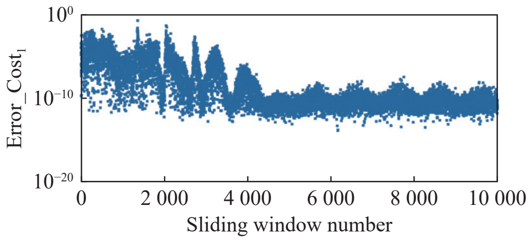
(a) factor_1 (b) factor_2

(c) factor_3 (d) factor_4

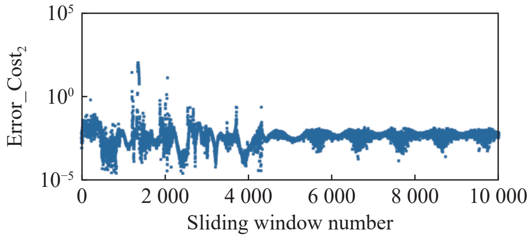
(e) factor_5 (f) factor_6

Fig. 15 Variation of correction factor ε

Fig.16 shows the training performance of each sliding window. It can be seen from Fig. 16 that most of the window training errors are within the specified range except for individual windows under the constraint of double gradient descent. When the pulse engines are ignited, the LSTM neural network is insensitive to mutations due to its memory of previous information, which affects network convergence. However, the influence only exists in individual windows. As the training information increases, the LSTM neural network gradually finds the ignition rule of the pulse engine. It can be seen that the training error is lower than that of the first ignition when the second and third pulse engines are ignited.



(a) Error_Cost_1



(b) Error_Cost$_2$

Fig. 16 Training performance of each sliding windows

6. Conclusions

An EKF with an adaptive factor is presented to estimate the time-varying parameters of the missile dual control system. The Kalman gain is modified by the LSTM neural network to improve the performance of EKF in the presence of large model uncertainties. Compared with other estimation methods using neural networks, this method retains the physical process of the system through Kalman filtering, and its filtering performance is more reliable. The multi-gradient descent constraint is used in the network training process, and the derivation process of the whole system error back propagation is given to fit the dynamic process of the system. In order to reduce the network error caused by the time dependence in the long-term prediction of the LSTM neural network, an adaptive correction factor is introduced to correct the network output in real time. The rolling training mode is used to implement an online training correction and an online prediction logic. In the simulation experiment, the LSTM-EKF algorithm proposed in this paper is superior to the traditional EKF and AEKF.

References

[1] RICCO R A, TEIXEIRA B. Least−squares parameter estimation for state−space models with state equality constraints. International Journal of Systems Science, 2021, 53(1): 1−13.

[2] LU X J. Online identification of aircraft parameters using state-space model. Changsha: National University of Defense Technology, 2016. (in Chinese)

[3] ZHOU S. Operational identification of time-varying model parameters for thermal structures of high-speed aerial vehicles. Acta Aeronautica ET Astronautica Sinica, 2015, 36(1): 373−380.

[4] ARULAMPALAM M S, MASKELL S, GORDON N, et al. A tutorial on particle filters for online nonlinear/non-Gaussian Bayesian tracking. IEEE Trans. on Signal Processing, 2002, 50(2): 174–188.

[5] JULIER S J, UHLMANN J K. Unscented filtering and nonlinear estimation. Proceedings of the IEEE, 2004, 92(3): 401–422.

[6] ARASARATNAM I, HAYKIN S. Cubature Kalman filters. IEEE Trans. on Automatic Control, 2009, 54(6): 1254–1269.

[7] JETTO L, LONGHI S, VENTURINI G. Development and experimental validation of an adaptive extended Kalman filter for the localization of mobile robots. IEEE Trans. on Robotics and Automation. 1999, 15(2): 219−229.

[8] ROBLES J J, CARDENAS−MANSILLA G, LEHNERT R. Adaptive selection of anchors in the extended Kalman filter tracking algorithm. Proc. of the IEEE 11th Workshop on Positioning Navigation and Communication, 2014. DOI: 10.1109/WPNC.2014.6843294.

[9] CHEN X. An identification method of fast time-varying parameters adapted to aircraft control systems. Acta Aeronautica ET Astronautica Sinica, 1990, 11(9): 474−479.

[10] SUN Y, LIU Y. Adaptive synchronization control and parameters identification for chaotic fractional neural networks with time-varying delays. Neural Processing Letters, 2021, 53(4): 2729–2745.

[11] WANG S, ZHAO X, YU Q, et al. Identification of driver braking intention based on long short-term memory (LSTM) network. IEEE Access, 2020, 8: 180422–180432.

[12] HERMANS M, SCHRAUWEN B. Training and analyzing deep recurrent neural networks. Advances in Neural Information Processing Systems, 2013, 2013(1): 190–198.

[13] BENGIO Y, SIMARD P. Learning long-term dependencies with gradient descent is difficult. IEEE Trans. on Neural Networks, 1994, 5(2): 157–166.

[14] LO J T, BASSU D. Adaptive multilayer perceptrons with long-and short-term memories. IEEE Trans. on Neural Networks, 2002, 13(1): 22−33.

[15] YU Y, SI X S, HU C H, et al. A review of recurrent neural networks: LSTM cells and network architectures. Neural Computation, 2019, 31(7): 1235−1270.

[16] HOCHREITER S, SCHMIDHUBER J. Long short-term memory. Neural Computation, 1997, 9(8): 1735−1780.

[17] XING Y, LYU C. Dynamic state estimation for the advanced brake system of electric vehicles by using deep recurrent neural networks. IEEE Trans. on Industrial Electronics, 2020, 67(11): 9536−9547.

[18] ZHOU Z Y, ZHANG R X, ZHU Z F. Robust Kalman filtering with long short-term memory for image-based visual servo control. Multimedia Tools and Applications, 2019, 78(18): 26341−26371.

[19] MA C Y, WANG A N, CHEN G, et al. Hand joints-based gesture recognition for noisy dataset using nested interval unscented Kalman filter with LSTM network. The Visual Computer, 2018, 34(6/8): 1053−1063.

[20] ZHENG T Y, YAO Y, HE F H. An RNN-based learnable extended Kalman filter design and application. Proc. of the IEEE 18th European Control Conference, 2019. DOI: 10.23919/ECC.2019.8796088.

[21] BAO T Z, ZHAO Y H. A deep Kalman filter network for hand kinematics estimation using sEMG. Pattern Recognition Letters, 2021, 143: 88–94.

[22] NI Z C, YANG Y, XIU X. Battery state of charge estimation using long short-term memory network and extended Kalman filter. Proc. of the IEEE 39th Chinese Control Conference, 2020. DOI: 10.23919/CCC50068.

[23] HONG J, WANG Z, CHEN W, et al. Synchronous multi-parameter prediction of battery systems on electric vehicles using long short-term memory networks. Applied Energy, 2019, 254: 113648.

[24] ZMITRI M, FOURATI H, PRIEUR C. Inertial velocity estimation for indoor navigation through magnetic gradient-based EKF and LSTM learning model. Proc. of the IEEE/RSJ International Conference on Intelligent Robots and Systems, 2020. DOI: 10.1109/IROS45743.

[25] NI Y, XIA Z L, ZHAO F T. An online multistep-forward

voltage-prediction approach based on an LSTM-TD model and KF algorithm. Computer, 2021, 54(8): 56–65.

[26] ZHU X L, HAO K R, XIE R M, et al. Soft sensor based on extreme gradient boosting and bidirectional converted gates long short-term memory self-attention network. Neurocomputing, 2021, 434: 126–136.

[27] MENG Y, LIU L. A deep learning approach for a source code detection model using self-attention. Complexity, 2020, 2020: 5027198.

[28] BABAK A, ALIREZA G B, HAMID K, et al. A novel attention-based LSTM cell post-processor coupled with Bayesian optimization for streamow prediction. Journal of Hydrology, 2021, 601: 126526.

[29] SAGE A P, HUSA G W. Algorithms for sequential adaptive estimation of prior statistics. Proc. of the IEEE 8th Symposium on Adaptive Processes Decision and Control, 1969. DOI: 10.1109/SAP.1969.269927.

[30] CHEN Z, SHU X, LI G, et al. Stage of charge estimation of lithium-ion battery packs based on improved cubature Kalman filter with long short-term memory model. IEEE Trans. on Transportation Electrification, 2020, 73(3): 1271–1284.

[31] RIDGELY D B, DRAKE D, TRIPLETT L. Dynamic control allocation of a missile with tails and reaction jets. Proc. of the AIAA Guidance, Navigation and Control Conference and Exhibit, 2007. DOI: 10.2514/6.2007-6671.

Biographies

YUAN Yuqi was born in 1994. He received his B.E. and M.E. degrees from Harbin Engineering University in 2016 and 2019. He is now studying for a Ph.D. degree in Harbin Institute of Technology. His research interests include neural network, system identification, and nonlinear filtering.
E-mail: 994631152@qq.com

ZHOU Di was born in 1969. He received his B.E. and Ph.D. degrees in automatic control from Harbin Institute of Technology, Harbin, China, in 1991 and 1996, respectively. He once worked as a postdoctoral fellow in the Automation Department, Tsinghua University, Beijing, China, and a research fellow in the Department of Mechanical Engineering, Sophia University, Japan. He is now a professor in School of Astronautics, Harbin Institute of Technology. His research interests include nonlinear control, nonlinear filtering, and missile guidance and control.
E-mail: zhoud@hit.edu.cn

LI Junlong was born in 1964. He received his Ph.D. degree from Harbin Institute of technology and now works as a researcher in Beijing Institute of Electronic System Engineering. His research interests include the overall design of aircraft , the navigation, guidance, and control.
E-mail: hit_szL@163.com

LOU Chaofei was born in 1974. He received his Ph.D. degree from Beijing Institute of Electronic System Engineering, Beijing, China, in 2007. He is now working as a senior engineer in Beijing Institute of Electronic System Engineering. His research interests include navigation, guidance, and control.
E-mail: louchaofei@ yahoo.com.cn