IEEE Access
Multidisciplinary : Rapid Review : Open Access Journal

Received August 28, 2019, accepted September 12, 2019, date of publication September 16, 2019, date of current version September 30, 2019.

*Digital Object Identifier 10.1109/ACCESS.2019.2941651*

# A New Predictive Sliding Mode Control Approach for Networked Control Systems With Time Delay and Packet Dropout

**YU ZHANG, SHOUSHENG XIE, LITONG REN, AND LEDI ZHANG**
Aeronautical Engineering Department, Air Force Engineering University, Xi'an 710038, China

Corresponding author: Yu Zhang (frank_sharon1314@126.com)

This work was supported in part by the National Natural Science Foundation of China under Grant 51606219, Grant 51476187, and Grant 51506221.

**ABSTRACT** This paper investigates the predictive sliding mode control problem of networked control system with long-time delay and consecutive packet dropout in both sensor-controller link and controller-actuator link. A new modeling method that uses only one Markov chain to describe the time delay and packet dropout in a unified model is proposed. As a modification of the original law, a new chattering-free reaching law that is suitable for multiple-input systems is proposed and is later used as the reference trajectory of the designed predictive sliding mode controller. To overcome the influence of time delay and packet dropout, a novel predictive sliding mode controller equipped with a logic zero-order-holder and delay compensator is proposed, and the proposed compensation strategy is theoretically proven to be able to make the system completely free from the influence of long-time delay and consecutive packet dropout. Finally, a simulation example is given to illustrate the validity of the proposed controller.

**INDEX TERMS** Networked control systems, sliding mode control, predictive control, time delay, packet dropout.

## I. INTRODUCTION

Recently, networked control systems (NCSs) have attracted much attention because of their theoretical and practical significance. In NCSs, the actuators, controller and sensors of a physical plant are distributed in a large physical space and linked together by a communication network. The advantages of NCSs, such as low cost, reduced weight, and easy installation and maintenance, which were unavailable in the past, have been widely recognized [1], [2]. Despite these advantages, the introduction of communication network will inevitably lead to time delay, packet dropout, packet disordering, communication constraints, quantization errors, etc., which may cause degradation or even instability of the NCSs [3]–[9]. Among all the challenging issues that have emerged, time delay and packet dropout are recognized as the most common and critical problems of NCSs and thus have attracted considerable research interest [3], [4], [10]–[14].

System modeling and controller design serve as two main aspects of the study on NCSs with time delay and packet dropout. Various approaches have been proposed to model the time delay and packet dropout in NCSs so that their effects on system stability and performance can be fully examined. For NCSs with only time delay, time delay system (TDS) theory can be directly used by simply regarding the network-induced delay (or even data packet dropout as well) as a delay parameter of the control system [15], [16]. For NCSs with only packet dropouts, one of the most commonly used modeling methods is the switched systems approach, where a set of subsystems is constructed and the switching between subsystems is decided by the packet dropout conditions [17], [18]. However, in a real network, time delay and packet dropout usually exist simultaneously. Therefore, it is important to establish a model to handle packet dropout and time delay in a common framework. Moreover, consideration of the time delay and packet dropout in both the sensor-controller (S-C) link and controller-actuator (C-A) link is necessary because it can make the established model more practical [19]. A popular modeling method used by

The associate editor coordinating the review of this manuscript and approving it for publication was Zonghua Gu.

researchers to model NCSs with both time delay and packet dropout is called the stochastic system approach, where NCSs with time delay and packet dropout are transformed into systems with stochastic parameters with known expectation and variance. One way is to model packet dropout and time delay as stochastic variables satisfying the Bernoulli random binary distribution, with the probability of packet dropout or time delay being a given constant [20]–[22]. This method is usually used for NCSs considering only a ''one-step'' time delay and without consecutive packet dropout. For example, in [23], the random one-step transmission delays and packet dropouts are modeled by a Bernoulli distribution, and an observer-based feedback controller is designed to make the closed-loop networked system robustly and exponentially stable in terms of the mean square. Another stochastic system modeling method is to describe time delay and packet dropout as a Markov process taking values in a finite set [4], [9], [10], [14], [24]–[26]. For example, [14] considered time delay and packet dropout as a random Markov process, based on which a new packet reordering approach was proposed to cope with packet disordering; similarly, three Markov chains were used in [26] to describe time delays and packet dropouts in both S-C and C-A links. The Bernoulli process can be regarded as a special type of Markov process, but when used for system modeling, a difference between them exists; thus, they are discussed separately here. Notably, the abovementioned stochastic system modeling methods all focus on the time delay and packet dropout of the controller output signal rather than the actuator input signal, which means that the pattern of signals actually received by the actuator in each sampling period is not clear. However, this can result in two problems. One is the use of outdated data even when updated information is already available; the other is packet disordering. To address this problem, [27] introduced a zero-order-hold (ZOH) for actuators to ensure the use of the latest arrived signals. However, only packet dropout (actually, only time delay) is considered in [27]. Another serious problem is that most of the papers mentioned above consider time delay and packet dropout as two different phenomena that occur independently. However, when, for example, a control signal fails to reach the actuator on time, whether time delay or packet dropout occurs is usually difficult to determine, and when long-time delay and consecutive packet dropout are considered, the situation may worsen. Motived by the above discussion, this paper first focuses on how to establish a unified model to describe simultaneous long-time delay and consecutive packet dropout and how to reflect the inner connection between them.

Regarding controller design, sliding mode control (SMC) is well known for its robustness in handling uncertainties such as external disturbances and system modeling errors [7], [13], [24], [28]–[30]. Moreover, SMC allows the decoupling of overall system motion into partial components of lower dimensions, i.e., sliding motion and approaching motion. As a result, sliding mode controller design is composed of two steps. In the first step, a sliding surface is designed such that the plant dynamics is restricted to the surface equations and is robust to system parametric uncertainties and external disturbances. In the second step, a feedback control law is designed to make the system trajectory converge to the sliding surface in finite time [31]. These advantages make SMC a good choice for NCSs with network problems and uncertainties. For example, [32] proposed an event-triggered SMC for a class of uncertain NCSs with stochastic perturbation, exogenous disturbance and network-induced communication constraints. Reference [33] designed an integral SMC for a stochastic system under an imperfect quantization mechanism; packet dropout was also considered. Despite the mentioned advantages, the undesired chattering produced by the high-frequency switching of the control may be considered a problem in implementing the SMC methods for some real applications, especially for discrete time systems [34]. Moreover, the existence of time delay and packet dropout usually intensifies the chattering problem because the control signal needs to be switched immediately once the sliding mode state crosses the sliding surface. To overcome this problem, contributions have been made by researchers, and in recent years, the combination of model predictive control (MPC) and SMC has been suggested as an appropriate solution [30], [34]–[37]. Reference [34] proposes a good example of using predictive sliding mode control (PSMC) for time delay systems, suggesting that PSMC can be a good choice for systems with time delay. However, only state delay is considered in [34]. The situation of input delay is considered in [38], where a linear transformation method is adopted to eliminate the time delay term in the system expression and a sliding mode predictive controller is designed. However, this linear transformation method can be used only when the input delay information of the current sampling period is available to the controller. In other words, the controller needs to foresee the delay of the control signal even before the signal is sent out, which is fairly hard to realize in practice. Therefore, this paper aims to propose a new PSMC strategy where time delay and packet dropout can be properly compensated for and no foreseen information is required by the controller.

The main contributions of this paper are as follows: (i) A new modeling method is proposed for NCSs with both long-time delay and consecutive packet dropout, through which time delay and packet dropout are established in a unified model described by one Markov chain, the time scale of which is linear with respect to physical time. (ii) A chattering-free sliding mode reaching law, which is a modification of the original form such that it is suitable for multiple-input systems, is proposed. (iii) A new predictive sliding mode controller with a delay compensator, which uses the predicted control signal sequence to compensate for time delay and packet dropout, thus making full use of the unique feature of ''predictive control'', is proposed.

The remainder of this paper is organized as follows. The problem formulation and proposed NCS modeling method are given in Section 2. The main results are presented in

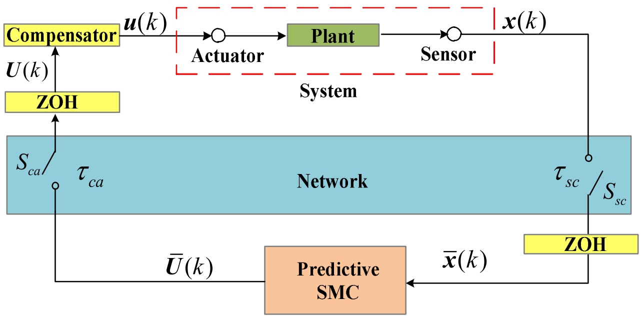



**FIGURE 1.** Scheme of NCSs equipped with logic ZOH and compensator.

Section 3, including the design of the chattering-free sliding mode reaching law, the predictive sliding mode controller, the logic-ZOH-based delay compensator, and the reachability analysis. A numerical example is provided in Section 4, and some conclusions are given in Section 5.

## II. PROBLEM FORMULATION

### A. PRELIMINARY DISCUSSION OF THE NCS EQUIPPED WITH A LOGIC ZOH AND A COMPENSATOR

NCSs equipped with logic ZOH and compensator are depicted in Fig. 1. At each time instant $k$, a state signal $\boldsymbol{x}(k)$ and its time stamp are encapsulated into a data packet and sent to the S-C channel logic ZOH through the communication network, and in each sampling period, the ZOH outputs the data packet $\bar{\boldsymbol{x}}(k)$ that has the newest time stamp information. Once this packet arrives at the controller, the controller immediately calculates and generates a new control signal (which, in this case, is a signal sequence $\bar{\boldsymbol{U}}(k) = [\bar{\boldsymbol{u}}^{\mathrm{T}}(k), \bar{\boldsymbol{u}}^{\mathrm{T}}(k+1), \ldots, \bar{\boldsymbol{u}}^{\mathrm{T}}(k+M-1)]^{\mathrm{T}}$ obtained from the PSMC). The signal sequence $\bar{\boldsymbol{U}}(k)$ is then encapsulated into a data packet with a time stamp and sent to the C-A channel logic ZOH through the communication network. Next, the ZOH outputs a signal sequence that has the latest time stamp information, denoted as $\boldsymbol{U}(k)$, to the compensator. Finally, the compensator chooses one control signal from $\boldsymbol{U}(k)$ according to its time stamp information and sends the control signal to the actuator.

The dynamics of the plant is given by:

$$\boldsymbol{x}(k+1) = \boldsymbol{A}\boldsymbol{x}(k) + \boldsymbol{B}\boldsymbol{u}(k) \tag{1}$$

where $\boldsymbol{x}(k) \in \boldsymbol{R}^n$ is the system-state variable; $\boldsymbol{u}(k) \in \boldsymbol{R}^m$ is the control input; and $\boldsymbol{A}$, $\boldsymbol{B}$ are matrices with appropriate dimensions.

Several assumptions are stated for the NCS.

*Assumption 1:* The system is controllable, and all system states are observable.

*Assumption 2:* The calculation time of the controller, ZOH and compensator is small enough to be neglected.

*Assumption 3:* The S-C time delay $\tau_{sc}$ is bounded, with the upper bound being $\bar{\tau}_{sc}$, and the C-A time delay $\tau_{ca}$ is bounded, with the upper bound being $\bar{\tau}_{ca}$. Packet dropouts on both sides are also bounded, with $\bar{\rho}_{sc}$ and $\bar{\rho}_{ca}$ being the largest number of consecutive packet dropouts of the S-C and C-A channels, respectively.

### B. A NEW MODELING METHOD FOR TIME DELAY AND PACKET DROPOUT

Here, we first take into account the case of time delay and packet dropout over the link from the controller to the actuator; then, the cases over the link from the sensor to the controller are addressed using a similar method. To illustrate the proposed method clearer, we first discuss the case of longtime delay, which is described by a specially defined Markov chain. Then, the inner connection between time delay and packet dropout is taken into account, and the time delay transition probability matrix is used such that the case of consecutive packet dropout is included as a special type of time delay. Thereafter, a unified time delay and packet dropout model is established. Notably, the existence of the ZOH makes it possible to address time delay and packet dropout in this way.

Assume that the C-A channel time delay at time instant $k$ is $\tau_{ca}(k)$, which takes values in the set $\boldsymbol{\Omega}_{\tau_{ca}} = \{0, 1, 2, \ldots\bar{\tau}_{ca}\}$. Suppose that $\tau_{ca}(k) = i, (i < \bar{\tau}_{ca})$, where $i$ is a nonnegative integer; then, there are $i+2$ possible cases of time delay at the next time instant $k+1$, that is, $\tau_{ca}(k+1) \in \{0, 1, 2, \ldots, i+1\}$. Specifically, $\tau_{ca}(k+1) = 0$ means that the controller output at time instant $k+1$ is successfully transmitted to the actuator at $k+1$ with no time delay; then, the value of i changes from $i$ to 0 at $k+1$, and the signals sent out earlier than $k+1$ are not used afterwards. $\tau_{ca}(k+1) = 1$ means that the controller output at time instant $k+1$ has not arrived at $k+1$. However, when the signal sent out at time instant $k$ arrives, the value of $i$ changes from $i$ to 1 at $k+1$, and the signals sent out earlier than $k$ are not used afterwards. The remaining cases except for $\tau_{ca}(k+1) = i+1$ can be explained in the same way. Finally, $\tau_{ca}(k+1) = i+1$ means that no newly arrived signal appears at $k+1$ and that the control input at $k+1$ remains the same as that at $k$; then, the value of $i$ is increased by one period, thus becoming $i+1$. Specifically, when $\tau_{ca}(k) = \bar{\tau}_{ca}$, since the time delay is bounded, only $\bar{\tau}_{ca}$ possible cases exist for $\tau_{ca}(k+1)$. Based on the above discussion, we denote $\pi_{ij}(i \in \boldsymbol{\Omega}_{\tau_{ca}}, j \in \{0, 1, \ldots, i+1\} \cap \boldsymbol{\Omega}_{\tau_{ca}})$ as the transition probability parameter; then, the C-A channel time delay state transition can be described as:

$$\begin{cases} \pi_{ij} = \Pr\left(\tau_{ca}(k+1) = j \,|\, \tau_{ca}(k) = i\right) \\ \sum_{j=0}^{\bar{\tau}_{ca}} \pi_{ij} = 1, \quad i \in \boldsymbol{\Omega}_{\tau_{ca}}, j \in \{0, 1, \ldots, i+1\} \cap \boldsymbol{\Omega}_{\tau_{ca}} \end{cases} \tag{2}$$

Define $\boldsymbol{\Pi} = \pi_{ij}$ as the C-A channel time delay transition possibility matrix, which can be expressed as (3) shown at the bottom of the next page.

Notably, the time scale adopted in this model is linear with respect to physical time. In other words, the increase in time delay over one sampling period is limited to one in this model.

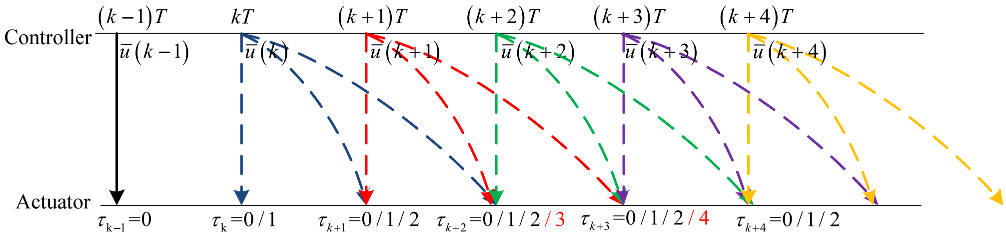



**FIGURE 2.** An example of the packet transition timing diagram when $\bar{\tau}_{ca} = 2$ and $\bar{\rho}_{ca} = 2$.

We can now take consecutive packet dropout into consideration. Packet dropout can be viewed as a special kind of time delay or infinite time delay. On the other hand, since the time delay is bounded, time delays beyond a predefined boundary can be viewed as packet dropout. This situation explains the inner connection between time delay and packet dropout. In this model, since the ZOH is applied and, as mentioned above, the increase in time delay is limited to one period, packet dropout can be directly included by expanding the transition probability matrix. However, a difference between the packet dropout model and time delay model exists, which needs to be considered when expanding the transition probability matrix. To clearly show how consecutive packet dropout is modeled, an example with $\bar{\tau}_{ca} = 2$ and $\bar{\rho}_{ca} = 2$ is given first, the packet transition timing diagram of which is shown in Fig. 2.

Fig. 2 assumes that $\bar{\boldsymbol{u}}(k-1)$ is successfully received by the actuator at time instant $k-1$, which means that $\tau_{ca}(k-1) = 0$. Then, at time instant $k$, as explained above, only two conditions exist: $\tau_{ca}(k) = 0$ and $\tau_{ca}(k) = 1$. Then, if $\tau(k) = 1$ ($\bar{\boldsymbol{u}}(k)$ doesn't arrive at $k$), three time delay conditions for time instant $k+1$ exist: $\tau_{ca}(k+1) = 0$, which means that $\bar{\boldsymbol{u}}(k+1)$ arrives at $k+1$; $\tau_{ca}(k+1) = 1$, which means that $\bar{\boldsymbol{u}}(k+1)$ does not arrive but $\bar{\boldsymbol{u}}(k)$ does arrive; or $\tau_{ca}(k+1) = 2$, which means that no newly arrived signal appears and $\bar{\boldsymbol{u}}(k-1)$ remains the control input. Similarly, if $\tau_{ca}(k+1) = 2$, four time delay conditions for time instant $k+2$ exist, with $\tau_{ca}(k+2) = 0, 1, 2$ sharing the same explanations as for the previous time instant. However, if the case of no newly arrived signal continues, since the upper bound of the time delay is $\bar{\tau}_{ca} = 2$, then packet $\bar{\boldsymbol{u}}(k)$ is certainly lost. However, because of the existence of the ZOH, $\bar{\boldsymbol{u}}(k-1)$ remains the control input, and the time delay of this condition can be viewed as $\tau_{ca}(k+2) = 3$. Subsequently, if $\tau_{ca}(k+2) = 3$, then four time delay conditions for time instant $k+3$ exist, with $\tau_{ca}(k+3) = 0, 1, 2$ sharing the same explanations as for the previous time instant. If the case of no newly arrived signal persists, then packet $\bar{\boldsymbol{u}}(k+1)$ is also lost, and $\bar{\boldsymbol{u}}(k-1)$ remains the control input, thus yielding $\tau_{ca}(k+3) = 4$. The case of $\tau_{ca}(k+3) = 3$ is not possible because it can occur only when $\bar{\boldsymbol{u}}(k)$ arrives at $k+3$, but, as explained above, $\bar{\boldsymbol{u}}(k)$ is already lost. Finally, when $\tau_{ca}(k+3) = 4$, only three time delay conditions for time instant $k+4$ exist because the largest number of consecutive packet dropouts is $\bar{\rho}_{ca} = 2$, which means that $\tau_{ca}(k+4)$ can be only 0, 1 or 2.

Based on the illustration above, we denote $\hat{\tau}_{ca}$ as the C-A channel equivalent time delay that considers both time delay and packet dropout. Then, the transition probability matrix of $\hat{\tau}_{ca}$ is as follows as (4) shown at the top of the next page.

Therefore, we define $\boldsymbol{\Omega}_{\hat{\tau}_{ca}} = \{0, 1, 2, ..., (\bar{\tau}_{ca} + \bar{\rho}_{ca})\}$, and then the transition of equivalent time delay $\hat{\tau}_{ca}$ can be described as:

$$\begin{cases} \pi_{ij} = \Pr\left(\hat{\tau}_{ca}(k+1) = j \,\middle|\, \hat{\tau}_{ca}(k) = i\right) \\ \displaystyle\sum_{j=0}^{\bar{\tau}_{ca}+\bar{\rho}_{ca}} \pi_{ij} = 1, \\ \quad i \in \boldsymbol{\Omega}_{\hat{\tau}_{ca}}, j \in \{0, 1, \ldots, i+1\} \cap \boldsymbol{\Omega}_{\hat{\tau}_{ca}} \cup \{i+1\} \end{cases} \tag{5}$$

In the same way, the S-C channel equivalent time delay $\hat{\tau}_{sc}$ is given. We define $\boldsymbol{\Omega}_{\hat{\tau}_{sc}} = \{0, 1, 2, ..., (\bar{\tau}_{sc} + \bar{\rho}_{sc})\}$, and then the transition of the equivalent time delay (considering packet dropout) can be described as (7) shown at the top of the next page:

$$\begin{cases} \delta_{ij} = \Pr\left(\hat{\tau}_{sc}(k+1) = j \,\middle|\, \hat{\tau}_{sc}(k) = i\right) \\ \displaystyle\sum_{j=0}^{\bar{\tau}_{sc}+\bar{\rho}_{sc}} \delta_{ij} = 1, \\ \quad i \in \boldsymbol{\Omega}_{\hat{\tau}_{sc}}, j \in \{0, 1, \ldots, i+1\} \cap \boldsymbol{\Omega}_{\hat{\tau}_{sc}} \cup \{i+1\} \end{cases} \tag{6}$$

The transition probability matrix of the S-C channel equivalent time delay is defined as $\boldsymbol{\Delta} = \delta_{ij}$, and we have

Subsequently, since the time delay and packet dropout in forward and backward channels can be described by

$$\boldsymbol{\Pi} = \begin{bmatrix} \pi_{00} & \pi_{01} & 0 & 0 & \cdots & 0 \\ \pi_{10} & \pi_{11} & \pi_{12} & 0 & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & \ddots & \vdots \\ \pi_{(\bar{\tau}_{ca}-2)0} & \pi_{(\bar{\tau}_{ca}-2)1} & \pi_{(\bar{\tau}_{ca}-2)2} & \cdots & \pi_{(\bar{\tau}_{ca}-2)(\bar{\tau}_{ca}-1)} & 0 \\ \pi_{(\bar{\tau}_{ca}-1)0} & \pi_{(\bar{\tau}_{ca}-1)1} & \pi_{(\bar{\tau}_{ca}-1)2} & \cdots & \pi_{(\bar{\tau}_{ca}-1)(\bar{\tau}_{ca}-1)} & \pi_{(\bar{\tau}_{ca}-1)\bar{\tau}_{ca}} \\ \pi_{\bar{\tau}_{ca}0} & \pi_{\bar{\tau}_{ca}1} & \pi_{\bar{\tau}_{ca}2} & \cdots & \pi_{\bar{\tau}_{ca}(\bar{\tau}_{ca}-1)} & \pi_{\bar{\tau}_{ca}\bar{\tau}_{ca}} \end{bmatrix} \tag{3}$$

$$\mathbf{\Pi}=\begin{bmatrix}
\pi_{00} & \pi_{01} & 0 & & & & & & \cdots & 0\\
\pi_{10} & \pi_{11} & \pi_{12} & 0 & & & & & \cdots & 0\\
\vdots & \vdots & \ddots & \ddots & \ddots & & & & \vdots & \vdots\\
\pi_{\bar{\tau}_{ca}0} & \pi_{\bar{\tau}_{ca}1} & \cdots & \pi_{\bar{\tau}_{ca}\bar{\tau}_{ca}} & \pi_{\bar{\tau}_{ca}(\bar{\tau}_{ca}+1)} & 0 & & & \cdots & 0\\
\pi_{(\bar{\tau}_{ca}+1)0} & \pi_{(\bar{\tau}_{ca}+1)1} & \cdots & \pi_{(\bar{\tau}_{ca}+1)\bar{\tau}_{ca}} & 0 & \pi_{(\bar{\tau}_{ca}+1)(\bar{\tau}_{ca}+2)} & 0 & & \cdots & 0\\
\pi_{(\bar{\tau}_{ca}+2)0} & \pi_{(\bar{\tau}_{ca}+2)1} & \cdots & \pi_{(\bar{\tau}_{ca}+2)\bar{\tau}_{ca}} & 0 & 0 & \pi_{(\bar{\tau}_{ca}+2)(\bar{\tau}_{ca}+3)} & 0 & \cdots & 0\\
\vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \ddots & \ddots & \ddots & \vdots\\
\pi_{(\bar{\tau}_{ca}+\bar{\rho}_{ca}-2)0} & \pi_{(\bar{\tau}_{ca}+\bar{\rho}_{ca}-2)1} & \cdots & \pi_{(\bar{\tau}_{ca}+\bar{\rho}_{ca}-2)\bar{\tau}_{ca}} & 0 & 0 & \cdots & 0 & \pi_{(\bar{\tau}_{ca}+\bar{\rho}_{ca}-2)(\bar{\tau}_{ca}+\bar{\rho}_{ca}-1)} & 0\\
\pi_{(\bar{\tau}_{ca}+\bar{\rho}_{ca}-1)0} & \pi_{(\bar{\tau}_{ca}+\bar{\rho}_{ca}-1)1} & \cdots & \pi_{(\bar{\tau}_{ca}+\bar{\rho}_{ca}-1)\bar{\tau}_{ca}} & 0 & 0 & \cdots & 0 & 0 & \pi_{(\bar{\tau}_{ca}+\bar{\rho}_{ca}-1)(\bar{\tau}_{ca}+\bar{\rho}_{ca})}\\
\pi_{(\bar{\tau}_{ca}+\bar{\rho}_{ca})0} & \pi_{(\bar{\tau}_{ca}+\bar{\rho}_{ca})1} & \cdots & \pi_{(\bar{\tau}_{ca}+\bar{\rho}_{ca})\bar{\tau}_{ca}} & 0 & 0 & \cdots & 0 & 0 & 0
\end{bmatrix} \quad (4)$$

$$\mathbf{\Delta}=\begin{bmatrix}
\delta_{00} & \delta_{01} & 0 & & & & & & \cdots & 0\\
\delta_{10} & \delta_{11} & \delta_{12} & 0 & & & & & \cdots & 0\\
\vdots & \vdots & \ddots & \ddots & \ddots & & & & \vdots & \vdots\\
\delta_{\bar{\tau}_{sc}0} & \delta_{\bar{\tau}_{sc}1} & \cdots & \delta_{\bar{\tau}_{sc}\bar{\tau}_{sc}} & \delta_{\bar{\tau}_{sc}(\bar{\tau}_{sc}+1)} & 0 & & & \cdots & 0\\
\delta_{(\bar{\tau}_{sc}+1)0} & \delta_{(\bar{\tau}_{sc}+1)1} & \cdots & \delta_{(\bar{\tau}_{sc}+1)\bar{\tau}_{sc}} & 0 & \delta_{(\bar{\tau}_{sc}+1)(\bar{\tau}_{sc}+2)} & 0 & & \cdots & 0\\
\delta_{(\bar{\tau}_{sc}+2)0} & \delta_{(\bar{\tau}_{sc}+2)1} & \cdots & \delta_{(\bar{\tau}_{sc}+2)\bar{\tau}_{sc}} & 0 & 0 & \delta_{(\bar{\tau}_{sc}+2)(\bar{\tau}_{sc}+3)} & 0 & \cdots & 0\\
\vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \ddots & \ddots & \ddots & \vdots\\
\delta_{(\bar{\tau}_{sc}+\bar{\rho}_{sc}-2)0} & \delta_{(\bar{\tau}_{sc}+\bar{\rho}_{sc}-2)1} & \cdots & \delta_{(\bar{\tau}_{sc}+\bar{\rho}_{sc}-2)\bar{\tau}_{sc}} & 0 & 0 & \cdots & 0 & \delta_{(\bar{\tau}_{sc}+\bar{\rho}_{sc}-2)(\bar{\tau}_{sc}+\bar{\rho}_{sc}-1)} & 0\\
\delta_{(\bar{\tau}_{sc}+\bar{\rho}_{sc}-1)0} & \delta_{(\bar{\tau}_{sc}+\bar{\rho}_{sc}-1)1} & \cdots & \delta_{(\bar{\tau}_{sc}+\bar{\rho}_{sc}-1)\bar{\tau}_{sc}} & 0 & 0 & \cdots & 0 & 0 & \delta_{(\bar{\tau}_{sc}+\bar{\rho}_{sc}-1)(\bar{\tau}_{sc}+\bar{\rho}_{sc})}\\
\delta_{(\bar{\tau}_{sc}+\bar{\rho}_{sc})0} & \delta_{(\bar{\tau}_{sc}+\bar{\rho}_{sc})1} & \cdots & \delta_{(\bar{\tau}_{sc}+\bar{\rho}_{sc})\bar{\tau}_{sc}} & 0 & 0 & \cdots & 0 & 0 & 0
\end{bmatrix} \quad (7)$$

equivalent time delay $\hat{\tau}_{ca}$ and $\hat{\tau}_{sc}$, then according to [9], the equivalent time delay can be combined as $\hat{\tau} = \hat{\tau}_{ca} + \hat{\tau}_{sc}$, and the Markov chain used to describe the transition of the lumped equivalent time delay can be obtained by combining the Markov chains generated by $\mathbf{\Pi}$ and $\mathbf{\Delta}$.

To further illustrate the novelty of the proposed method, we compare it with the widely used Markov-chain-based modeling method [10, 14, 25, 26, 37]. In these articles, the time delay is usually described as follows. Assume that the random delay $\tau(k)$ takes values in the set $\mathbf{\Omega}_\tau = \{1, 2, \ldots \bar{\tau}\}$ with the following mode transition probabilities:

$$\begin{cases}\pi_{ij} = \Pr\left(\tau(k+1) = j \,|\, \tau(k) = i\right)\\ \displaystyle\sum_{j=1}^{\bar{\tau}} \pi_{ij} = 1, \quad i, j \in \mathbf{\Omega}_\tau\end{cases} \quad (8)$$

The transition probability matrix is defined by

$$\mathbf{\Pi} = \begin{bmatrix}\pi_{11} & \pi_{12} & \cdots & \pi_{1\bar{\tau}}\\ \pi_{21} & \pi_{22} & \cdots & \pi_{2\bar{\tau}}\\ \vdots & \vdots & \ddots & \vdots\\ \pi_{\bar{\tau}1} & \pi_{\bar{\tau}2} & \cdots & \pi_{\bar{\tau}\bar{\tau}}\end{bmatrix} \quad (9)$$

First, this model does not consider the probability of $\tau(k) = 0$, which means that all data packets suffer from time delay. However, for a normal network, we believe that $\tau(k) = 0$ should be considered. Moreover, the time scale of this model is not linear over physical time, which may result in packet disordering. Finally, if we choose $\bar{\tau} = 5$, then the transition probability matrix (9) will be a $6 \times 6$ full matrix with 36 nonzero entries (where $\tau(k) = 0$ is also considered), but for the proposed model of this paper, only 26 nonzero entries are required, which reduces the work in obtaining the transition probability matrix.

In addition, for NCSs with time delay and packet dropout, the existing modeling methods usually introduce another Markov chain or an extra Bernoulli process to describe the state of packet dropout [14, 25, 26], but the modeling method proposed in this paper uses only one Markov chain to realize description of the time delay and packet dropout.

Therefore, the novelty of the proposed modeling method can be concluded as follows:

(i) It innovatively models a long time delay and consecutive packet dropout in a unified model described by only one Markov chain;

(ii) The time scale adopted in our Markov chain is linear over the physical time, i.e., the state transition in our Markov chain always occurs in one physical time instant;

(iii) Compared with traditional models, the transition probability matrix of our model is not a full matrix, and thus, less work is required to obtain the transition probability matrix.

*Remark 1:* Since we do not use the input hold strategy to compensate for the equivalent time delay, the equivalent time delay term $\hat{\tau}(k)$ is not added directly to the expression of system (1) as $\boldsymbol{u}(k - \hat{\tau}(k))$ but will be considered when designing the compensator-based PSMC.

## III. MAIN RESULTS

### A. DESIGN OF A CHATTERING-FREE REACHING LAW

First, for system (1), the linear sliding surface is defined by

$$\boldsymbol{s}(k) = \boldsymbol{C}\boldsymbol{x}(k) \quad (10)$$

where $\boldsymbol{C} \in \boldsymbol{R}^{m\times n}$ is a constant matrix chosen such that $\boldsymbol{CB}$ is invertible and an appropriate value of $\boldsymbol{C}$ can be determined through the pole placement method such that the system state on the sliding surface can converge to the equilibrium point [39].

The reaching-law-based SMC approach was first introduced in [40], and the reaching law for the sliding mode was designed as:

$$\boldsymbol{s}(k+1) = (1-qT)\boldsymbol{s}(k) - \varepsilon T sgn(\boldsymbol{s}(k)) \tag{11}$$

with $\varepsilon > 0, \ q > 0, \ 0 < 1 - qT < 1$.

However, the controller designed based on this approach cannot drive the system trajectory to the equilibrium point but only to its neighboring area with $|\boldsymbol{s}(k)| \geq \varepsilon T/(2-qT)$, and if $|\boldsymbol{s}(k)| = \varepsilon T/(2-qT)$, then $|\boldsymbol{s}(k+1)| = |\boldsymbol{s}(k)| = \varepsilon T/(2-qT)$, the result of which is equal-amplitude chattering [41].

To avoid the chattering problem of classic SMC, a new chattering-free sliding mode reaching law is proposed inspired by [34]. However, the reaching law in this paper is proposed for a multiple-input system, which means that sliding surface $\boldsymbol{s}(k)$ is not a scalar but a vector; hence, the switching condition of the reaching law should be newly defined. Since it is a multiple-input system with $\boldsymbol{u}(k) \in \boldsymbol{R}^m$, then $m$ sliding surfaces will exist. The vector form of the reaching law is given below:

$$\boldsymbol{s}(k+1) = \xi(\boldsymbol{s}(k))(1-qT)\boldsymbol{s}(k) - \varphi(\boldsymbol{s}(k))\varepsilon T sgn(\boldsymbol{s}(k)) \tag{12}$$

where

$$\xi(\boldsymbol{s}(k)) = \begin{cases} 1, & \|\boldsymbol{s}(k)\|_\infty > \eta \\ 0, & \|\boldsymbol{s}(k)\|_\infty \leq \eta \end{cases} \tag{13}$$

$$\varphi(\boldsymbol{s}(k)) = \begin{cases} \boldsymbol{I}_{m\times m}, & \|\boldsymbol{s}(k)\|_\infty > \eta \\ \dfrac{diag\left\{|s_1(k)|^2, |s_2(k)|^2, \ldots, |s_m(k)|^2\right\}}{\eta}, & \|\boldsymbol{s}(k)\|_\infty \leq \eta \end{cases} \tag{14}$$

with $0 < 1-qT < 1, 0 < \varepsilon T < 1$ and $\eta = \frac{\varepsilon T}{1-qT}$.

The reachability of the PSMC proposed based on this reaching law and how it is able to minimize chattering will be proved later in the last part of this section.

### B. SYNTHESIS OF PREDICTIVE SLIDING MODE CONTROLLER WITH CHATTERING-FREE REACHING LAW

We take reaching law (12) as a reference sliding mode trajectory, which is defined as

$$\begin{cases} \boldsymbol{s_r}(k+p) = (1-qT)\boldsymbol{s_r}(k+p-1)\xi(\boldsymbol{s_r}(k+p-1)) \\ \qquad -\varepsilon T sgn(\boldsymbol{s_r}(k+p-1))\varphi(\boldsymbol{s_r}(k+p-1)) \\ \boldsymbol{s_r}(k) = \boldsymbol{s}(k) = \boldsymbol{Cx}(k) \end{cases} \tag{15}$$

where the definitions of $\xi(\boldsymbol{s}(k))$ and $\varphi(\boldsymbol{s}(k))$ are the same as those in (12).

Using (1) as the prediction model, the predicted sliding mode state at time $k+p$ at time instant $k$ is calculated according to system (1) and sliding surface (10) as:

$$\begin{aligned} \boldsymbol{s_p}(k+p/k) &= \boldsymbol{Cx}(k+p) \\ &= \boldsymbol{CAx}(k+p-1) + \boldsymbol{CBu}(k+p-1) \\ &= \cdots = \boldsymbol{CA}^p\boldsymbol{x}(k) + \sum_{i=1}^{p}\boldsymbol{CA}^{i-1}\boldsymbol{Bu}(k+p-i) \end{aligned} \tag{16}$$

Considering the predicted sliding mode state at time $k+1, k+2, \ldots, k+N$ at time $k$, a vector form of (16) can be obtained:

$$\boldsymbol{S_p}(k+1) = \boldsymbol{\Phi X}(k) + \boldsymbol{\Psi U}(k) \tag{17}$$

where

$$\begin{aligned} \boldsymbol{S_p}(k+1/k) &= [\boldsymbol{s_p}(k+1/k), \boldsymbol{s_p}(k+2/k), \ldots, \boldsymbol{s_p}(k+N/k)]^T, \\ \boldsymbol{X}(k) &= \left[\boldsymbol{x}^{\mathrm{T}}(k), \ \boldsymbol{x}^{\mathrm{T}}(k), \ \cdots, \boldsymbol{x}^{\mathrm{T}}(k)\right]^{\mathrm{T}}, \\ \boldsymbol{U}(k) &= [\boldsymbol{u}^{\mathrm{T}}(k), \boldsymbol{u}^{\mathrm{T}}(k+1), \ldots, \boldsymbol{u}^{\mathrm{T}}(k+M-1)]^{\mathrm{T}}, \\ \boldsymbol{\Phi} &= diag\left\{\boldsymbol{CA}, \boldsymbol{CA}^2, \ \cdots, \boldsymbol{CA}^N\right\}, \end{aligned}$$

$$\boldsymbol{\Psi} = \begin{bmatrix} \boldsymbol{CB} & 0 & \cdots & \cdots & 0 \\ \boldsymbol{CAB} & \boldsymbol{CB} & \cdots & \cdots & 0 \\ \cdots & \cdots & \cdots & \cdots & \cdots \\ \boldsymbol{CA}^{M-1}\boldsymbol{B} & \cdots & \cdots & \boldsymbol{CAB} & \boldsymbol{CB} \\ \cdots & \cdots & \cdots & \cdots & \cdots \\ \boldsymbol{CA}^{N-2}\boldsymbol{B} & \cdots & \cdots & \boldsymbol{CA}^{N-M}\boldsymbol{B} & \boldsymbol{CA}^{N-M-1}\boldsymbol{B} \\ \boldsymbol{CA}^{N-1}\boldsymbol{B} & \cdots & \cdots & \boldsymbol{CA}^{N-M+1}\boldsymbol{B} & \boldsymbol{CA}^{N-M}\boldsymbol{B} \end{bmatrix},$$

with $N$ being the prediction horizon and $M$ being the control horizon.

Then, the error between the actual sliding mode state $\boldsymbol{s}(k)$ and its predicted value at time $k-p$ is introduced as feedback compensation to correct the future predicted sliding mode state $\boldsymbol{s_p}(k+p/k)$. The predicted sliding mode state of $\boldsymbol{s}(k)$ at time $k-p$ is denoted as $\boldsymbol{s_y}(k/k-p)$, which is obtained according to (16) as:

$$\boldsymbol{s_p}(k/k-p) = \boldsymbol{CA}^p\boldsymbol{x}(k-p) + \sum_{i=1}^{p}\boldsymbol{CA}^{i-1}\boldsymbol{Bu}(k-i) \tag{18}$$

Define the prediction error as $\boldsymbol{e}(k) = \boldsymbol{s}(k) - \boldsymbol{s_p}(k/k-p)$; then, we have

$$\begin{aligned} \hat{\boldsymbol{s}}_p(k+p) &= \boldsymbol{s_p}(k+p/p) + \boldsymbol{h_p e}(k) \\ &= \boldsymbol{CA}^p\boldsymbol{x}(k) + \sum_{i=1}^{p}\boldsymbol{CA}^{i-1}\boldsymbol{Bu}(k+p-i) + \boldsymbol{h_p e}(k) \end{aligned} \tag{19}$$

where $\hat{\boldsymbol{s}}_p(k+p)$ is the predicted value of the sliding mode state at time $k+p$ after feedback compensation is introduced and $\boldsymbol{h_p} \in \boldsymbol{R}^{m\times m}$ is the correction coefficient.

Similarly, the vector form of (19) is:

$$\hat{\boldsymbol{S}}_{\boldsymbol{y}}(k+1) = \boldsymbol{S_y}(k+1) + \boldsymbol{H_p E}(k) \tag{20}$$

where

$$\begin{aligned} \hat{\boldsymbol{S}}_p(k+1) &= [\hat{\boldsymbol{s}}_p(k+1), \hat{\boldsymbol{s}}_p(k+2), \ldots, \hat{\boldsymbol{s}}_p(k+N)]^T, \\ \boldsymbol{E}(k) &= [\boldsymbol{s}(k) - \boldsymbol{s_p}(k/k-1), \cdots, \boldsymbol{s}(k) - \boldsymbol{s_p}(k/k-N)]^T, \\ \boldsymbol{H_p} &= diag\{\boldsymbol{h}_{p1}, \boldsymbol{h}_{p2}, \ldots, \boldsymbol{h}_{pn}\}. \end{aligned}$$

Subsequently, since PSMC aims to minimize the error between the predicted sliding mode state and its desired counterpart given by the reference trajectory, the following cost function is defined:

$$j_p = \sum_{i=1}^{N} q_i[\hat{s}_p(k+i) - s_r(k+i)]^2 + \sum_{j=1}^{M} r_i[u(k+j-1)]^2 \tag{21}$$

where $q_i$ is a nonnegative weight coefficient matrix that determines the weight of errors of different sampling points in the performance index $j_p$ and $r_i$ is a positive weight coefficient matrix that is used to constrain the control variables.

Notably, the cost function is not a globally invariant function but a rolling updated performance index, which is usually a minimized value in a limited time horizon.

The vector form of (21) can be rewritten as:

$$\begin{aligned} J_p &= [\hat{S}_y(k+1) - S_r(k+1)]^T Q[\hat{S}_y(k+1) - S_r(k+1)] \\ &\quad + U(k)^T RU(k) \\ &= [\Phi Z(k) + \Psi U(k) + \Gamma(k) + H_p E(k) - S_r(k+1)]^T Q \\ &\quad \cdot [\Phi Z(k) + \Psi U(k) + \Gamma(k) + H_p E(k) - S_r(k+1)] \\ &\quad + U(k)^T RU(k) \end{aligned} \tag{22}$$

where

$$Q = diag[q_1, q_2, \cdots q_N] and R = diag[r_1, r_2, \cdots, r_M].$$

Let $\frac{\partial J_p}{\partial U(k)} = 0$, yielding the PSMC law:

$$U(k) = -(\Psi^T Q \Psi + R)^{-1} \Psi^T Q\ [\Phi X(k) + H_p E(k) - S_r(k+1)] \tag{23}$$

Note that $U(k) = [u^{\mathrm{T}}(k), u^{\mathrm{T}}(k+1), \ldots, u^{\mathrm{T}}(k+M-1)]^{\mathrm{T}}$ is not one control signal but a sequence of control signals. Normally, when time delay and packet dropout are not considered, only the first signal $u(k)$ is used as the control input. However, when considering time delay and packet dropout, the subsequent control signals $u(k+1), \ldots, u(k+M-1)$ can be ideal choices for compensation. In this way, the concept of "predictive control" is fully utilized in addressing NCS network problems.

### C. SYNTHESIS OF PSMC WITH LOGIC-ZOH-BASED DELAY COMPENSATOR

First, since time delay and packet dropout are considered, we denote the $U(k)$ obtained in (23) as $\bar{U}(k) = [\bar{u}^{\mathrm{T}}(k), \bar{u}^{\mathrm{T}}(k+1), \ldots, \bar{u}^{\mathrm{T}}(k+M-1)]^{\mathrm{T}}$, which indicates that it is the controller output. Then, we denote the ZOH output as $U(k)$. Considering the lumped equivalent time delay $\hat{\tau}(k)$, the relation between $\bar{U}(k)$ and $U(k)$ is:

$$U(k) = \begin{cases} \bar{U}(k) & \hat{\tau}(k) = 0 \\ \bar{U}(k-1) & \hat{\tau}(k) = 1 \\ \vdots & \vdots \\ \bar{U}(k-\bar{\hat{\tau}}) & \hat{\tau}(k) = \bar{\hat{\tau}} \end{cases} \tag{24}$$

where $\bar{\hat{\tau}} = \bar{\tau}_{ca} + \bar{\rho}_{ca} + \bar{\tau}_{sc} + \bar{\rho}_{sc}$ is the upper bound of the lumped equivalent time delay $\hat{\tau}(k)$.

Then, the compensator considering time delay and packet dropout is designed according to the following logic: If $\hat{\tau}(k) = 0$, the newest signal sequence is $U(k) = \bar{U}(k)$, and then the first control signal of $\bar{U}(k)$, i.e., $\bar{u}(k)$, is used as the actuator input, which can be denoted as $\bar{u}(k)_k$. If $\hat{\tau}(k) = 1$, the newest signal sequence is $U(k) = \bar{U}(k-1)$, and then the second control signal of $\bar{U}(k-1)$, i.e., $\bar{u}(k-1+1)$, is used as the actuator input, which can be denoted as $\bar{u}(k)_{k-1}$. If we set the control horizon as $M = \bar{\hat{\tau}} + 1$, then the compensator can provide compensation signals for all possible time delay and packet dropout conditions, i.e., $\bar{u}(k - \bar{\hat{\tau}} + \bar{\hat{\tau}})$ can be used as the actuator input if the equivalent time delay is $\hat{\tau}(k) = \bar{\hat{\tau}}$, which is denoted as $\bar{u}(k)_{k-\bar{\hat{\tau}}}$. The above compensation strategy can be described by the following function:

$$u(k) = g(\hat{\tau}(k))U(k) = g(\hat{\tau}(k))\bar{U}(k - \hat{\tau}(k)) \tag{25}$$

where

$$g(x) = [g_0, g_1, \ldots, g_i, \ldots, g_{\bar{\tau}+\bar{\rho}}], \quad g_i = \begin{cases} I_{m\times m} & i = x \\ 0_{m\times m} & i \neq x \end{cases} \tag{26}$$

That is,

$$u(k) = \begin{cases} [I_{m\times m}, 0, \ldots, 0]\bar{U}(k) = \bar{u}(k)_k & \hat{\tau}(k) = 0 \\ [0, I_{m\times m}, \ldots, 0]\bar{U}(k-1) = \bar{u}(k)_{k-1} & \hat{\tau}(k) = 1 \\ \vdots & \\ [\underbrace{0 \ldots 0}_{\hat{\tau}(k)}, I_{m\times m}, 0 \ldots 0]\bar{U}(k - \hat{\tau}(k)) = \bar{u}(k)_{k-\hat{\tau}(k)} & \hat{\tau}(k) = \hat{\tau}(k) \\ \vdots & \\ [0, \ldots, 0, I_{m\times m}]\bar{U}(k - \bar{\hat{\tau}}) = \bar{u}(k)_{k-\bar{\hat{\tau}}} & \hat{\tau}(k) = \bar{\hat{\tau}} \end{cases} \tag{27}$$

*Remark 2:* The superscript of control input $\bar{u}(k)$ designates the sampling period on which the control input is calculated. The subscripts of the predicted value of $\bar{u}(k)$ in different sampling periods are different because the reference trajectory is updated in every sampling period according to the newest sliding mode state $s(k)$. However, we later prove that if the system model is precise enough and the disturbance is small enough to be neglected, then the predicted value of $\bar{u}(k)$ at different sampling points can remain the same; thus, the influence of the time delay and packet dropout can be eliminated.

### D. REACHABILITY ANALYSIS

As shown in (1), no modeling error or external disturbance occurs; in other words, the established system model is precise enough and the external disturbance is small enough to be neglected. Now, we prove the sliding mode reachability under this condition.

Considering the predicted control signal sequence $\bar{U}(k)$ given by (23), we assume that no constraint exists for the control variable, and thus, we have $\boldsymbol{R} = \mathbf{0}$. Then, (23) becomes:

$$\bar{\boldsymbol{U}}(k) = -\boldsymbol{\Psi}^{-1}\left[\boldsymbol{\Phi X}(k) + \boldsymbol{H_p E}(k) - \boldsymbol{S_r}(k+1)\right] \quad (28)$$

Since no modeling error exists and the external disturbance is small enough to be neglected, we can say that no error exists between the sliding mode state $\boldsymbol{s}_p(k/k-p)$ and its actual value $\boldsymbol{s}(k)$, thus making $\boldsymbol{e}(k) = \boldsymbol{s}(k) - \boldsymbol{s}_p(k/k-p) \approx 0$. Therefore, (28) can be further written as:

$$\bar{\boldsymbol{U}}(k) = -\boldsymbol{\Psi}^{-1}\left[\boldsymbol{\Phi X}(k) - \boldsymbol{S_r}(k+1)\right] \quad (29)$$

Then, the control input $\boldsymbol{u}(k)$ for different equivalent time delay conditions is calculated according to (29) as:

$$\begin{aligned}\boldsymbol{u}(k) &= \bar{\boldsymbol{u}}(k)_{k-\hat{\tau}(k)}\\ &= (-\boldsymbol{CB})^{-1}\left[\boldsymbol{CAC}^{-1}\boldsymbol{s}_r(k)_{k-\hat{\tau}(k)} - \boldsymbol{s}_r(k+1)_{k-\hat{\tau}(k)}\right]\end{aligned} \quad (30)$$

Now, we prove how the control signal can be free from the influence of equivalent time delay $\hat{\tau}(k)$if the proposed compensation strategy is applied.

First, we assume that the control signal sequence at time instant $k_0$ arrives on time, i.e., $\hat{\tau}(k_0) = 0$. Then, according to (30), we have

$$\begin{aligned}\boldsymbol{u}(k_0) &= \bar{\boldsymbol{u}}(k_0)_{k_0-0}\\ &= (-\boldsymbol{CB})^{-1}\left[\boldsymbol{CAC}^{-1}\boldsymbol{s}_r(k_0)_{k_0} - \boldsymbol{s}_r(k_0+1)_{k_0}\right]\end{aligned} \quad (31)$$

From the reference trajectory function (15), $\boldsymbol{s}_r(k_0) = \boldsymbol{s}(k_0) = \boldsymbol{Cx}(k_0)$; thus, we have

$$\boldsymbol{u}(k_0) = \bar{\boldsymbol{u}}(k_0)_{k_0} = (-\boldsymbol{CB})^{-1}\left[\boldsymbol{CAx}(k_0) - \boldsymbol{s}_r(k_0+1)_{k_0}\right] \quad (32)$$

Then, the actual sliding mode state at time instant $k_0+1$ is

$$\begin{aligned}\boldsymbol{s}(k_0+1) &= \boldsymbol{Cx}(k_0+1) = \boldsymbol{C}\left[\boldsymbol{Ax}(k_0) + \boldsymbol{Bu}(k_0)\right]\\ &= \boldsymbol{CAx}(k_0) - \boldsymbol{CAx}(k_0) + \boldsymbol{s}_r(k_0+1)_{k_0} = \boldsymbol{s}_r(k_0+1)_{k_0}\end{aligned} \quad (33)$$

Equation (33) means that the actual sliding mode state at time instant $k_0 + 1$ is the same as the one defined in the reference trajectory at time instant $k_0$. Then, according to (15) and (33), we have

$$\begin{aligned}\boldsymbol{s_r}(k_0+2)_{k_0+1} &= (1-qT)\boldsymbol{s_r}(k_0+1)_{k_0+1}\xi(\boldsymbol{s_r}(k_0+1)_{k_0+1})\\ &\quad - \varepsilon T sgn(\boldsymbol{s_r}(k_0+1)_{k_0+1})\varphi(\boldsymbol{s_r}(k_0+1)_{k_0+1})\\ &= (1-qT)\boldsymbol{s_r}(k_0+1)_{k_0}\xi(\boldsymbol{s_r}(k_0+1)_{k_0})\\ &\quad - \varepsilon T sgn(\boldsymbol{s_r}(k_0+1)_{k_0})\varphi(\boldsymbol{s_r}(k_0+1)_{k_0})\\ &= \boldsymbol{s_r}(k_0+2)_{k_0}\end{aligned} \quad (34)$$

Therefore, if $\hat{\tau}(k_0+1) = 0$, then

$$\begin{aligned}&\boldsymbol{u}(k_0+1)\\ &= \bar{\boldsymbol{u}}(k_0+1)_{k_0+1}\\ &= (-\boldsymbol{CB})^{-1}\left[\boldsymbol{CAC}^{-1}\boldsymbol{s}_r(k_0+1)_{k_0+1} - \boldsymbol{s}_r(k_0+2)_{k_0+1}\right]\\ &= (-\boldsymbol{CB})^{-1}\left[\boldsymbol{CAC}^{-1}\boldsymbol{s}_r(k_0+1)_{k_0+1} - \boldsymbol{s}_r(k_0+2)_{k_0+1}\right]\end{aligned} \quad (35)$$

If $\hat{\tau}(k_0+1) = 1$, then

$$\begin{aligned}\boldsymbol{u}(k_0+1) &= \bar{\boldsymbol{u}}(k_0+1)_{k_0}\\ &= (-\boldsymbol{CB})^{-1}\left[\boldsymbol{CAC}^{-1}\boldsymbol{s}_r(k_0+1)_{k_0} - \boldsymbol{s}_r(k_0+2)_{k_0}\right]\end{aligned} \quad (36)$$

Based on (33) and (34), the control signals calculated in (35) and (36) are the same, i.e.,

$$\boldsymbol{u}(k_0+1) = \bar{\boldsymbol{u}}(k_0+1)_{k_0+1} = \bar{\boldsymbol{u}}(k_0+1)_{k_0} \quad (37)$$

which means that the time delay $\tau(k_0+1) = 1$ does not affect the final used control input signal and is equivalent to the situation of $\tau(k_0+1) = 0$. Then, in the same way, by repeating the above process, the system can be proved to be free from the influence of equivalent time delay if $\hat{\tau}(k) \le M - 1$. This outcome means that if $\bar{\boldsymbol{U}}(k)$ arrives on time, then the following $M - 1$ control periods are free from the influence of time delay and packet dropout. Thus, if we set $M = \bar{\hat{\tau}} + 1$, we can say that the system can be totally free from the influence of time delay and packet dropout from time instant $k$ onward. Apparently, the first control signal is expected to arrive on time such that the system can be free from the influence of time delay and packet dropout from the very beginning. This scenario can be realized by setting the time instant at which the first control signal sequence arrives as the initial time. This consideration is reasonable because no control input exists before the first control signal sequence arrives and the system state remains the same as that at the actual initial time.

Now, we prove the reachability of the resulting SMC law. First, based on the above analysis, the resulting compensator-based sliding mode controller for system (1) is

$$\begin{aligned}\boldsymbol{u}(k) &= \bar{\boldsymbol{u}}(k)_{k-\tau(k)} = \bar{\boldsymbol{u}}(k)_k\\ &= (-\boldsymbol{CB})^{-1}\left[\boldsymbol{CAx}(k) - \boldsymbol{s}_r(k+1)_k\right]\\ &= (-\boldsymbol{CB})^{-1}\left[\boldsymbol{CAx}(k) - (1-qT)\boldsymbol{s}(k)\xi(\boldsymbol{s}(k))\right.\\ &\quad \left. + \varepsilon T sgn(\boldsymbol{s}(k))\varphi(\boldsymbol{s}(k))\right]\end{aligned} \quad (38)$$

*Theorem 1:* For NCSs in the form of (1), if the linear sliding surface is chosen as (10) and the SMC law is chosen as (38), then the trajectory of system sliding mode state $s_i(k)$, $(i = 1, \ldots, m)$ will enter the sliding surface neighborhood region $\boldsymbol{\Omega}$ from any initial state within at most $m_i^*$ steps and then will never escape from it, where

$$\boldsymbol{\Omega} = \left\{ s_i(k) : |s_i(k)| \le \eta = \frac{\varepsilon T}{1-qT} \right\} \quad (39)$$

$$m_i^* = [m_i] + 1 \quad (40)$$

$$m_i = \frac{s_i^2(0) - \eta^2}{\eta^2} \quad (41)$$

*Proof:* First, we prove that system sliding mode state $s_i(k)$, $(i = 1, \ldots, m)$ will enter the region $\boldsymbol{\Omega}$ within at most $m_i^*$ steps if it is not in it at the beginning.

Substituting (38) into system model (1), and according to sliding mode function (10), we have

$$\begin{aligned}\boldsymbol{s}(k+1) &= \boldsymbol{C}\boldsymbol{x}(k+1) = \boldsymbol{C}\left[\boldsymbol{A}\boldsymbol{x}(k) + \boldsymbol{B}\boldsymbol{u}(k)\right] \\ &= (1-qT)\boldsymbol{s}(k)\xi(\boldsymbol{s}(k)) - \varepsilon T sgn(\boldsymbol{s}(k))\varphi(\boldsymbol{s}(k)) \end{aligned} \tag{42}$$

We can write (42) in the following component-wise form:

$$s_i(k+1) = \xi(\boldsymbol{s}(k))(1-qT)s_i(k) - \varphi(\boldsymbol{s}(k))\varepsilon T sgn(s_i(k)) \tag{43}$$

Considering the $i$th sliding surface $s_i(k)$, if $s_i(k)$ is outside the region $\boldsymbol{\Omega}$, which means that $s_i(k) > \eta$ or $s_i(k) < -\eta$, then $\|\boldsymbol{s}(k)\|_\infty > \eta$. Therefore, (43) becomes

$$s_i(k+1) = (1-qT)s_i(k) - \varepsilon T sgn(s_i(k)) \tag{44}$$

Defining a Lyapunov function $V_i(k) = s_i^2(k)$, the following is obtained:

$$\begin{aligned}\Delta V_i(k) &= s_i^2(k+1) - s_i^2(k) \\ &= -\left[qTs_i(k) + \varepsilon T sgn(s_i(k))\right] \\ &\quad \cdot \left[2s_i(k) - qTs_i(k) - \varepsilon T sgn(s_i(k))\right] \end{aligned} \tag{45}$$

Then, the following two conditions are discussed:

① If $s_i(k) > \eta$, then

$$\Delta V_i(k) = -\left[qTs_i(k) + \varepsilon T\right] \cdot \left[2s_i(k) - qTs_i(k) - \varepsilon T\right] \tag{46}$$

Since $s_i(k) > \eta = \frac{\varepsilon T}{1-qT}$, we have

$$qTs_i(k) + \varepsilon T \geq qT \cdot \frac{\varepsilon T}{1-qT} + \varepsilon T = \frac{\varepsilon T}{1-qT} = \eta > 0 \tag{47}$$

Then, from $s_i(k) > \frac{\varepsilon T}{1-qT}$, the following can also be obtained:

$$s_i(k) > qTs_i(k) + \varepsilon T \tag{48}$$

Therefore, we have

$$2s_i(k) - qTs_i(k) - \varepsilon T \;\geq qTs_i(k) + \varepsilon T \;\geq \eta > 0 \tag{49}$$

Consequently,

$$\Delta V_i(k) = -\left[qTs_i(k) + \varepsilon T\right] \cdot \left[2s_i(k) - qTs_i(k) - \varepsilon T\right] \leq -\eta^2 < 0 \tag{50}$$

② If $s_i(k) < -\eta$, then, by a similar proof procedure, the relation $\Delta V_i(k) \leq -\eta^2$ can be shown to still hold.

Therefore, when $s_i(k) \notin \boldsymbol{\Omega}$, it can be concluded that $\Delta V_i(k) = s_i^2(k+1) - s_i^2(k) \leq -\eta^2 < 0$.

Moreover, from (50),

$$\begin{aligned}\Delta V_i(k) &= s_i^2(k) - s_i^2(k-1) \leq -\eta^2 \\ &\Leftrightarrow s_i^2(k) \leq s_i^2(k-1) - \eta^2 \leq s_i^2(k-2) - 2\eta^2 \\ &\leq \cdots \leq s_i^2(0) - k\eta^2 \end{aligned} \tag{51}$$

This formula means that if

$$s_i^2(k) \leq s_i^2(0) - k\eta^2 = \eta^2 \tag{52}$$

then we have $s_i(k) \in \boldsymbol{\Omega}$. Therefore, the solution $k$ of Eq. (52) would be the maximum step of the approaching phase. However, the solution of (52) may not be an integer. Therefore, we denote $m_i = \frac{s_i^2(0)-\eta^2}{\eta^2}$ as the real number solution of (52). Then, we can say that after at most $m_i^* = [m_i] + 1$ steps, the system sliding mode state will enter the sliding surface neighborhood region $\boldsymbol{\Omega}$. Therefore, after at most $\max\left\{m_i^*\right\}$, $i = 1, 2, \ldots, m$ steps, all sliding mode states will enter the region $\boldsymbol{\Omega}$.

Second, we prove that if $s_i(k) \in \boldsymbol{\Omega}$, then $s_i(k+1) \in \boldsymbol{\Omega}$.

When $s_i(k) \in \boldsymbol{\Omega}$, $-\eta \leq s_i(k) \leq \eta$. However, since $\|\boldsymbol{s}(k)\|_\infty = \max\{|s_i(k)|\}$, $i = 1, 2, \ldots, m$, it cannot be determined whether $\|\boldsymbol{s}(k)\|_\infty > \eta$ or $\|\boldsymbol{s}(k)\|_\infty < \eta$. Therefore, the following two cases are discussed.

1 When $\|\boldsymbol{s}(k)\|_\infty > \eta$, Eq. (43) becomes

$$s_i(k+1) = (1-qT)s_i(k) - \varepsilon T sgn(s_i(k)) \tag{53}$$

When $0 \leq s_i(k) \leq \eta$,

$$s_i(k+1) = (1-qT)s_i(k) - \varepsilon T \leq (1-qT)\frac{\varepsilon T}{1-qT} - \varepsilon T = 0 \tag{54}$$

and

$$s_i(k+1) = (1-qT)s_i(k) - \varepsilon T \geq -\varepsilon T \geq -\frac{\varepsilon T}{1-qT} = -\eta \tag{55}$$

Thus, we have $-\eta \leq s_i(k+1) \leq 0$, which means that $s_i(k+1) \in \boldsymbol{\Omega}$.

Similarly, when $-\eta \leq s_i(k) \leq 0$, we obtain $0 \leq s_i(k+1) \leq \eta$ and thus $s_i(k+1) \in \boldsymbol{\Omega}$.

2 When $\|\boldsymbol{s}(k)\|_\infty < \eta$, Eq. (43) becomes

$$s_i(k+1) = -\varepsilon T \frac{|s_i(k)|^2}{\eta} sgn(s_i(k)) \tag{56}$$

When $0 \leq s_i(k) \leq \eta$, we have

$$s_i(k+1) = -\varepsilon T \frac{|s_i(k)|^2}{\eta} \leq 0 \tag{57}$$

and

$$s_i(k+1) = -\varepsilon T \frac{|s_i(k)|^2}{\eta} \geq -\varepsilon T\eta \geq -\eta \tag{58}$$

Therefore, if $0 \leq s_i(k) \leq \eta$, then $-\eta \leq s_i(k+1) \leq 0$, which means that $s_i(k+1) \in \boldsymbol{\Omega}$.

When $-\eta \leq s_i(k) \leq 0$, we have

$$s_i(k+1) = \varepsilon T \frac{|s_i(k)|^2}{\eta} \geq 0 \tag{59}$$

and

$$s_i(k+1) = \varepsilon T \frac{|s_i(k)|^2}{\eta} \leq \varepsilon T\eta \leq \eta \tag{60}$$

Therefore, if $-\eta \leq s_i(k) \leq 0$, then $0 \leq s_i(k+1) \leq \eta$, which means that $s_i(k+1) \in \boldsymbol{\Omega}$.

Therefore, we can conclude that if $s_i(k) \in \mathbf{\Omega}$, then $s_i(k+1) \in \mathbf{\Omega}$.

We now discuss how the proposed method is able to suppress chattering.

*Theorem 2:* For NCSs in the form of (1), if the linear sliding surface chosen is (10) and the SMC law chosen is (38), then for an arbitrary small positive real number $\sigma > 0$, a positive integer $N$ exists such that for any $k > N$, $|s_i(k)| < \sigma$.

*Proof:*

According to Theorem 1, after at most $\max\left\{m_i^*\right\}$, $i = 1, 2, \ldots, m$ steps, all sliding mode states will enter the region $\mathbf{\Omega}$. Therefore, after at most $\max\left\{m_i^*\right\}$, $i = 1, 2, \ldots, m$ steps, we have $\|\boldsymbol{s}(k)\|_\infty < \eta$, and Eq. (43) becomes

$$\begin{aligned} s_i(k+1) &= -\varepsilon T \cdot \frac{|s_i(k)|^2}{\eta} sgn(s_i(k)) \\ &= -\varepsilon T \cdot \frac{|s_i(k)|^2}{\eta} \frac{s_i(k)}{|s_i(k)|} = -\frac{\varepsilon T\, |s_i(k)|}{\eta} s_i(k) \end{aligned} \tag{61}$$

Since the trajectory of system sliding mode state $s_i(k)$ is in $\mathbf{\Omega}$ with $|s_i(k)| \le \eta$ and $0 < \varepsilon T < 1$, we have

$$|s_i(k+1)| = \left| \frac{\varepsilon T\, |s_i(k)|}{\eta} s_i(k) \right| \le \varepsilon T\, |s_i(k)| < |s_i(k)| \tag{62}$$

Equation (62) proves that $|s_i(k+1)| < |s_i(k)|$ is always true, which means that the value of $|s_i(k)|$ decreases in every step until it converges to zero and that the equal-amplitude chattering case will never occur. In this way, chattering can be minimized.

*Remark 3:* For discrete time systems, the accurate arrival of the sliding mode state on the sliding surface is usually impossible to realize; thus, complete elimination of chattering is actually not possible in discrete time systems, and the expression "chattering-free" used here is just a conventional way to describe these chattering suppression methods [39]–[42].

## IV. SIMULATION EXAMPLE

In this section, a numerical example is given to illustrate the effectiveness of the proposed methods. Consider the following aero-engine rotor speed and total pressure control system small deviation linearization model given in [43], where $\boldsymbol{x} = [n_{\mathrm{L}} \quad n_{\mathrm{H}} \quad p_3]^{\mathrm{T}}$ and $\boldsymbol{u} = [m_{\mathrm{f}} \quad A_8]^{\mathrm{T}}$, with $n_{\mathrm{L}}$ being the low-pressure rotor speed; $n_{\mathrm{H}}$ being the high-pressure rotor speed; $p_3$ being the compressor exit total pressure; $m_{\mathrm{f}}$ being the fuel flow; and $A_8$ being the critical section area of the nozzle. The working condition of the aero-engine is set as $H = 0\ km$, $Ma = 0$, $n_{\mathrm{H}} = 100\%$, thus yielding the following parameter matrices for system (1):

$$\boldsymbol{A} = \begin{bmatrix} -0.8641 & 0.1491 & -0.01559 \\ -0.0073 & 0.9445 & -0.00532 \\ 0.4759 & -0.08775 & 0.5990 \end{bmatrix},$$

$$\boldsymbol{B} = \begin{bmatrix} 0.01935 & 0.00468 \\ 0.01731 & 0.01059 \\ 0.1853 & -0.0959 \end{bmatrix}.$$

In the aero-engine control system small deviation linearization model, the state variables and control inputs have been relativized. Therefore, we set the initial states of the system as $\boldsymbol{x}(0) = \begin{bmatrix} 1.2 & 0.6 & -0.8 \end{bmatrix}^T$, which means that the initial deviations of $n_{\mathrm{L}}$, $n_{\mathrm{H}}$ and $p_3$ are 1.2%, 0.6% and -0.8%, respectively. The system sampling interval is $T = 10\ ms$.

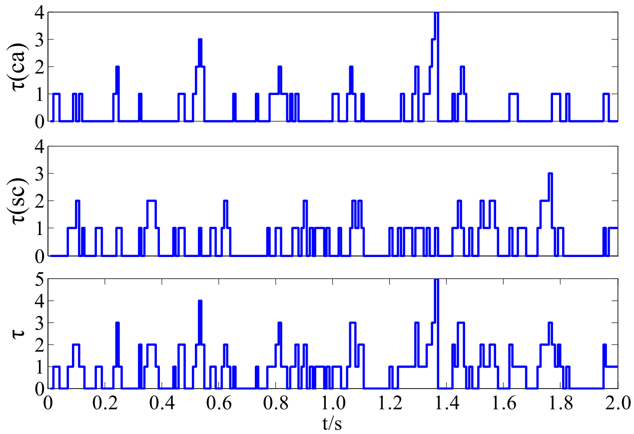



**FIGURE 3. Distributions of $\tau_{sc}(k)$, $\tau_{ca}(k)$ and $\tau(k)$.**

The upper bounds of the C-A channel time delay and consecutive packet dropout are $\bar{\tau}_{ca} = 2$ and $\bar{\rho}_{ca} = 2$, and those of the S-C channel are $\bar{\tau}_{sc} = 2$ and $\bar{\rho}_{sc} = 2$. The transition probability matrices $\mathbf{\Pi}$ and $\mathbf{\Delta}$ are defined as:

$$\mathbf{\Pi} = \begin{bmatrix} 0.8 & 0.2 & 0 & 0 & 0 \\ 0.6 & 0.3 & 0.1 & 0 & 0 \\ 0.4 & 0.3 & 0.2 & 0.1 & 0 \\ 0.3 & 0.3 & 0.3 & 0 & 0.1 \\ 0.5 & 0.4 & 0.1 & 0 & 0 \end{bmatrix},$$

$$\mathbf{\Delta} = \begin{bmatrix} 0.7 & 0.3 & 0 & 0 & 0 \\ 0.5 & 0.4 & 0.1 & 0 & 0 \\ 0.3 & 0.3 & 0.2 & 0.2 & 0 \\ 0.4 & 0.2 & 0.2 & 0 & 0.2 \\ 0.4 & 0.4 & 0.2 & 0 & 0 \end{bmatrix}$$

According to the transition probability matrices, two Markov chains are produced stochastically to show the distributions of the C-A and S-C channel equivalent time delay, and then the distribution of lumped time delay $\hat{\tau}(k)$ is obtained. The distributions of $\hat{\tau}_{ca}(k)$, $\hat{\tau}_{sc}(k)$ and $\hat{\tau}(k)$ are shown in Fig. 3.

By choosing the pole location of the closed-loop system as $[-0.5]$, the sliding surface parameter is calculated through the pole placement method as

$$\boldsymbol{C} = \begin{bmatrix} 0.5028 & 1 & 0 \\ 24.5525 & 0 & 1 \end{bmatrix}.$$

The selected sliding mode reaching law parameters are $q = 8$ and $\varepsilon = 0.5$.

The selected PSMC parameters are as follows:

(1) The control horizon is $M = \bar{\tau} + 1 = \bar{\tau}_{sc} + \bar{\rho}_{sc} + \bar{\tau}_{ca} + \bar{\rho}_{ca} + 1 = 9$, and the predictive horizon is $N = 12$;

(2) The weight coefficient matrices $\boldsymbol{Q}$ and $\boldsymbol{R}$ chosen are $\boldsymbol{Q} = \boldsymbol{I}_{2N \times 2N}$ and $\boldsymbol{R} = 0_{2M \times 2M}$.

The simulation results of the proposed sliding mode predictive controller (denoted as PSMC) are compared with

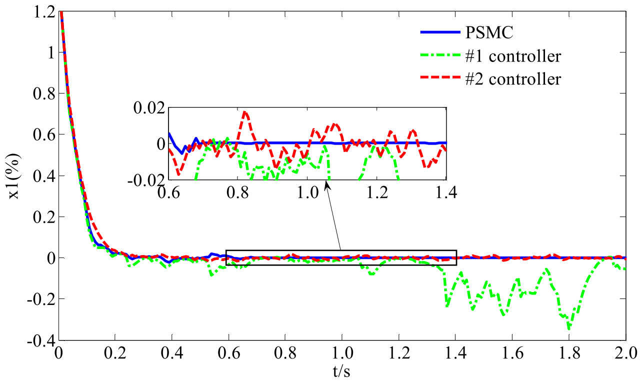



**FIGURE 4.** Response curves of state $x_1(k)$ for different controllers.

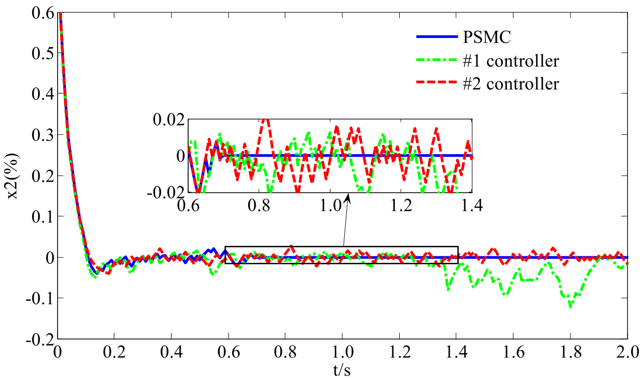



**FIGURE 5.** Response curves of state $x_2(k)$ for different controllers.

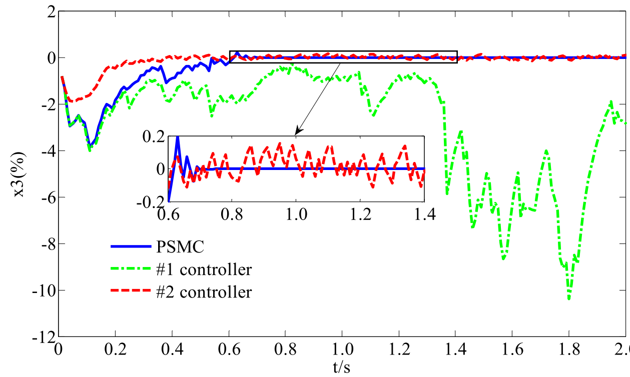



**FIGURE 6.** Response curves of state $x_3(k)$ for different controllers.

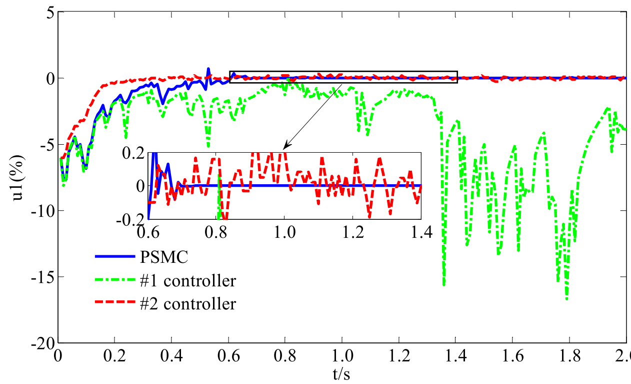



**FIGURE 7.** Evolution of control input $u_1(k)$ for different controllers.

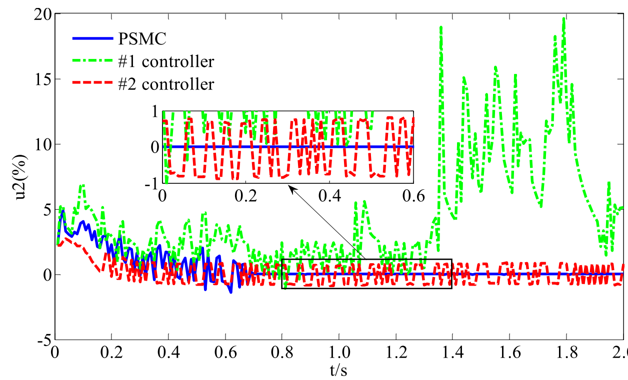



**FIGURE 8.** Evolution of control input $u_2(k)$ for different controllers.

those of two other controllers. One is the predictive controller proposed in [34], which is denoted as #1controller. And #1 controller can also be described by control law (23) with the same parameters, namely, $M = 9$, $N = 12$, $\boldsymbol{Q} = \boldsymbol{I}_{2N\times 2N}$ and $\boldsymbol{R} = 0_{2M\times 2M}$, and by linear sliding surface (10), with the same parameter $\boldsymbol{C}$ as calculated above. Meanwhile, the reference trajectory of #1 controller is set as in (15), with $q = 8$ and $\varepsilon = 0.5$. However, compared with the proposed controller, #1 controller does not have an input delay compensator, which means that only the first signal of $\bar{\boldsymbol{U}}(k) = [\bar{\boldsymbol{u}}(k), \bar{\boldsymbol{u}}(k+1), \ldots, \bar{\boldsymbol{u}}(k+M-1)]^{\mathrm{T}}$ is sent out as the controller output, i.e.,

$$\bar{\boldsymbol{u}}(k) = [\boldsymbol{I}_{m\times m}, \boldsymbol{0}, \ldots, \boldsymbol{0}]\bar{U}(k) \tag{63}$$

and when time delay or packet dropout occurs, the input hold strategy is used for compensation.

The other controller used for comparison is the one with the classic reaching law proposed in [40], denoted as #2controller, which is used to reflect the effectiveness of the proposed controller under chattering suppression. Compared with that for the proposed controller, only the reference trajectory has been changed to the classic reaching law (11). The other parameters, as well as the delay compensation strategy, remain the same as those for the proposed controller.

The response curves of the system states with the three different controllers are shown in Figs. 4-6. Notably, controllers with the proposed compensator (PSMC and #2controller) can drive the system states back to the equilibrium point. By contrast, #1controller cannot guarantee system-state convergence, and when the equivalent time delay increases, its performance quickly declines. When long-time delay and consecutive packet dropout exist, the input hold strategy is far from sufficient to provide suitable compensation for the control input, thus hindering the SMC signal from being switched in time when system states cross the sliding surface, resulting in high chattering or even system instability. In addition, compared with #2controller, the proposed controller stands out in terms of low chattering, and the influence of the time delay and packet dropout on the performance of the proposed controller is minimal.

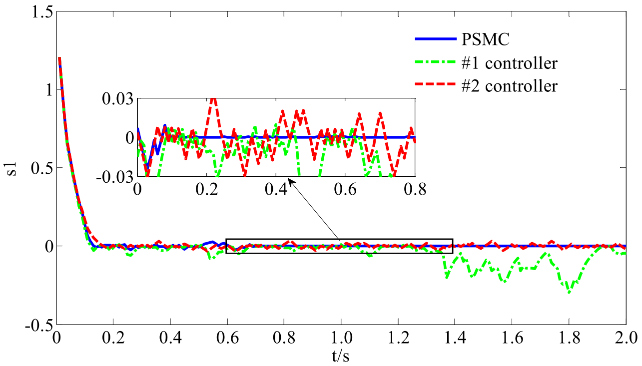



**FIGURE 9.** Evolution of sliding function $s_1(k)$ for different controllers.

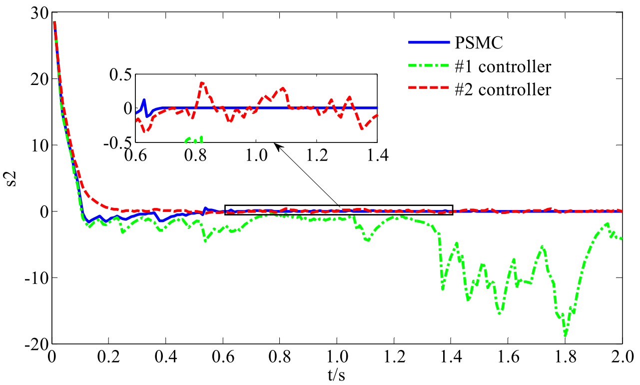



**FIGURE 10.** Evolution of sliding function $s_2(k)$ for different controllers.

Figs. 7 and 8 show the evolutions of the control input $\boldsymbol{u}_1(k)$ and $\boldsymbol{u}_2(k)$. Comparing Figs. 7 and 3, we can see that at approximately 1.4 s, the equivalent time delay increased to 5 and immediately resulted in a dramatic deterioration of the performance of #1controller. Fig. 8 shows that the chattering of #2controller is much fiercer than that of the proposed controller; this fierce chattering may damage the actuator. In this way, the superiority of the proposed controller is evident. Figs. 9 and 10 show the evolutions of sliding variables $\boldsymbol{s}_1(k)$ and $\boldsymbol{s}_2(k)$, which lead to the same conclusions as those obtained from the results given by Figs. 4-6.

## V. CONCLUSION

This paper focuses on the SMC problem of networked control system with a long-time delay and consecutive packet dropout in both controller-actuator link and sensor-controller link. First, a new modeling method is proposed, in which the logic ZOH is adopted to model the long-time delay and consecutive packet dropout in a unified model described by one Markov chain. In addition, since the transition probability matrix is not a full matrix, the difficulty of obtaining the parameters is reduced. Second, a chattering-free reaching law, which is a modification of the original one to make it suitable for a multiple-input system, is proposed, and the simulation results indicate that if this reaching law is used as the reference trajectory, it can effectively reduce chattering. Third, a predictive sliding mode controller equipped with a delay compensator is proposed. The combination of the predictive sliding mode controller and logic-ZOH-based compensator makes completely overcoming the influence of time delay and packet dropout possible by using the predicted control signal sequence of the previous time instant as the delay compensation. The proposed controller has been proved theoretically to be able to guarantee the reachability of the sliding mode dynamics and the chattering-free convergence of system states despite the existence of long-time delay and packet dropout, and simulation results also provide intuitive support for the effectiveness of the proposed method. However, the analysis was carried out under only the condition that the model established is precise enough and the external disturbance is small enough to be neglected. In our future work, we will concentrate on providing stability analysis for a system with noticeable modeling errors and external disturbances. Moreover, nonlinear systems will be considered in our future research, and we will seek to test the proposed method on a practical system so that the theory proposed can be further completed.

## REFERENCES

[1] X. Ge, F. Yang, and Q.-L. Han, "Distributed networked control systems: A brief overview," *Inf. Sci.*, vol. 380, pp. 117–131, Feb. 2015.

[2] D. Almakhles, A. K. Swain, A. Nasiri, and N. Patel, "An adaptive two-level quantizer for networked control systems," *IEEE Trans. Control Syst. Technol.*, vol. 25, no. 3, pp. 1084–1091, May 2017.

[3] H. Yang, X. Guo, L. Dai, and Y. Xia, "Event-triggered predictive control for networked control systems with network-induced delays and packet dropouts," *Int. J. Robust Nonlinear Control*, vol. 28, pp. 1350–1365, Mar. 2018.

[4] A. Elahi and A. Alfi, "Finite-time $H_\infty$ control of uncertain networked control systems with randomly varying communication delays," *ISA Trans.*, vol. 69, pp. 65–88, Jul. 2017.

[5] Y. Song and J. Wang, "Delay- and packet-disordering-dependent $H_\infty$ output tracking control for networked control systems," *Asian J. Control*, vol. 17, no. 3, pp. 917–931, May 2015.

[6] W. Wu and Y. Zhang, "Event-triggered fault-tolerant control and scheduling codesign for nonlinear networked control systems with medium-access constraint and packet disordering," *Int. J. Robust Nonlinear Control*, vol. 28, pp. 1182–1198, Mar. 2018.

[7] M. Li and Y. Chen, "Robust tracking control of networked control systems with communication constraints and external disturbance," *IEEE Trans. Ind. Electron.*, vol. 64, no. 5, pp. 4037–4047, May 2017.

[8] D. Zhang, S. K. Nguang, and L. Yu, "Distributed control of large-scale networked control systems with communication constraints and topology switching," *IEEE Trans. Syst., Man, Cybern., Syst.*, vol. 47, no. 7, pp. 1746–1757, Jul. 2017.

[9] H. Yang, Y. Xu, and J. Zhang, "Event-driven control for networked control systems with quantization and Markov packet losses," *IEEE Trans. Cybern.*, vol. 47, no. 8, pp. 2235–2243, Aug. 2017.

[10] X. Liu, X. Yu, G. Ma, and H. Xi, "On sliding mode control for networked control systems with semi-Markovian switching and random sensor delays," *Inf. Sci.*, vols. 337–338, pp. 44–58, Apr. 2016.

[11] Z. Pang, G. Liu, D. Sun, and D. Zhou, "Data-based predictive control for networked nonlinear systems with packet dropout and measurement noise," *J. Syst. Sci. Complex.*, vol. 30, no. 5, pp. 1072–1083, Oct. 2017.

[12] X. Li and S. Sun, "$H_\infty$ control for networked stochastic non-linear systems with randomly occurring sensor saturations, multiple delays and packet dropouts," *IET Control Theory Appl.*, vol. 11, no. 17, pp. 2954–2963, 2017.

[13] D. Shah and A. Mehta, "Discrete-time sliding mode controller subject to real-time fractional delays and packet losses for networked control system," *Int. J. Control, Automat. Syst.*, vol. 15, no. 6, pp. 2690–2703, 2017.

[14] W. Wu, "Fault-tolerant control of uncertain non-linear networked control systems with time-varying delay, packet dropout and packet disordering," *IET Control Theory Appl.*, vol. 11, no. 7, pp. 973–984, 2017.

[15] Y.-B. Zhao, G.-P. Liu, and D. Rees, "Stability and stabilisation of discrete-time networked control systems: A new time delay system approach," *IET Control Theory Appl.*, vol. 4, no. 9, pp. 1859–1866, 2010.
[16] Z. Wang, D. W. C. Ho, Y. Liu, and X. Liu, "Robust $H_\infty$ control for a class of nonlinear discrete time-delay stochastic systems with missing measurements," *Automatica*, vol. 45, no. 3, pp. 684–691, 2009.
[17] Y. Long and G.-H. Yang, "Robust fault detection for networked control systems with packet dropouts: An average dwell time method," *Optim. Control Appl. Methods*, vol. 36, no. 2, pp. 179–197, 2015.
[18] K. Lee and R. Bhattacharya, "Stability analysis of large-scale distributed networked control systems with random communication delays: A switched system approach," *Syst. Control Lett.*, vol. 85, pp. 77–83, Nov. 2015.
[19] A. H. Tahoun and H.-J. Fang, "Adaptive stabilisation of networked control systems tolerant to unknown actuator failures," *Int. J. Syst. Sci.*, vol. 42, no. 7, pp. 1155–1164, 2011.
[20] X. Liang, J. Xu, and H. Zhang, "Optimal control and stabilization for networked control systems with packet dropout and input delay," *IEEE Trans. Circuits Syst. II, Exp. Briefs*, vol. 64, no. 9, pp. 1087–1091, Sep. 2017.
[21] S. Wang, Y. Jianga, Y. Li, and D. Liu, "Fault detection and control co-design for discrete-time delayed fuzzy networked control systems subject to quantization and multiple packet dropouts," *Fuzzy Sets Syst.*, vol. 306, pp. 1–25, Jan. 2017.
[22] F. Yang, W. Wang, Y. Niu, and Y. Li, "Observer-based $H_\infty$ control for networked systems with consecutive packet delays and losses," *Int. J. Control Automat. Syst.*, vol. 8, no. 4, pp. 769–775, 2010.
[23] X. Li and S. Sun, "Robust $H_\infty$ control for networked systems with random packet dropouts and time delays," *Procedia Eng.*, vol. 29, pp. 4192–4197, Jan. 2012.
[24] H. Song, S.-C. Chen, and Y. Yam, "Sliding mode control for discrete-time systems with Markovian packet dropouts," *IEEE Trans. Cybern.*, vol. 47, no. 11, pp. 3669–3679, Nov. 2017.
[25] J. Dong and W. Kim, "Markov-chain-based output feedback control for stabilization of networked control systems with random time delays and packet losses," *Int. J. Control Automat. Syst.*, vol. 10, no. 5, pp. 1013–1022, 2012.
[26] L. Qiu, Q. Luo, F. Gong, S. Li, and B. Xu, "Stability and stabilization of networked control systems with random time delays and packet dropouts," *J. Franklin Inst.*, vol. 350, pp. 1886–1907, Sep. 2013.
[27] D. Wang, J. Wang, and W. Wang, "Output feedback control of networked control systems with packet dropouts in both channels," *Inf. Sci.*, vol. 221, pp. 544–554, Feb. 2013.
[28] C. Xiu and P. Guo, "Global terminal sliding mode control with the quick reaching law and its application," *IEEE Access*, vol. 6, pp. 49793–49800, Sep. 2018.
[29] A. Argha, L. Li, and S. W. Su, "$H_2$ based optimal sparse sliding mode control for networked control systems," *Int. J. Robust Nonlinear Control*, vol. 28, pp. 16–30, Jan. 2018.
[30] H. Li, H. Yang, F. Sun, and Y. Xia, "Sliding-mode predictive control of networked control systems under a multiple-packet transmission policy," *IEEE Trans. Ind. Electron.*, vol. 61, no. 11, pp. 6234–6243, Nov. 2014.
[31] W.-C. Yu, G.-J. Wang, and C.-C. Chang, "Discrete sliding mode control with forgetting dynamic sliding surface," *Mechatronics*, vol. 14, pp. 737–755, Sep. 2004.
[32] L. Wu, Y. Gao, J. Liu, and H. Li, "Event-triggered sliding mode control of stochastic systems via output feedback," *Automatica*, vol. 82, pp. 79–92, Aug. 2017.
[33] Y. Gao, W. Luo, J. Liu, and L. Wu, "Integral sliding mode control design for nonlinear stochastic systems under imperfect quantization," *Sci. China Inf. Sci.*, vol. 60, no. 12, 2017, Art. no. 120206.
[34] A. Nizar, B. Mansour, H. Nouri, and A. Said, "A new sliding function for discrete predictive sliding mode control of time delay systems," *Int. J. Automat. Comput.*, vol. 10, no. 4, pp. 288–295, Aug. 2013.
[35] B. Lian, Q. Zhang, and J. Li, "Sliding mode control for non-linear networked control systems subject to packet disordering via prediction method," *IET Control Theory Appl.*, vol. 11, no. 17, pp. 3079–3088, 2017.
[36] L. Xua and Q. Wang, "Sliding mode predictive control for networked control systems," *Energy Procedia*, vol. 13, pp. 4576–4583, Jan. 2011.
[37] Y. Xia, Z. Zhu, C. Li, H. Yang, and Q. Zhu, "Robust adaptive sliding mode control for uncertain discrete-time systems with time delay," *J. Franklin Inst.*, vol. 347, pp. 339–357, Feb. 2010.
[38] Y. Zhang, S. Xie, L. Zhang, and L. Ren, "Robust sliding mode predictive control of uncertain networked control system with random time delay," *Discrete Dyn. Nature Soc.*, vol. 2018, Jul. 2018, Art. no. 6959250. doi: 10.1155/2018/6959250.
[39] H. Du, X. Yu, M. Z. Q. Chen, and S. Li, "Chattering-free discrete-time sliding mode control," *Automatica*, vol. 68, no. 3, pp. 87–91, Jun. 2016.
[40] W. Gao, Y. Wang, and A. Homaifa, "Discrete-time variable structure control systems," *IEEE Trans. Ind. Electron.*, vol. 42, no. 2, pp. 117–122, Apr. 1995.
[41] J.-N. Li, H. Su, Y. Zhang, Z.-G. Wu, and J. Chu, "Chattering free sliding mode control for uncertain discrete time-delay singular systems," *Asian J. Control*, vol. 15, no. 1, pp. 260–269, 2013.
[42] G. Golo and Č. Milosavljević, "Robust discrete-time chattering free sliding mode control," *Syst. Control Lett.*, vol. 41, no. 1, pp. 19–28, 2000.
[43] L. Ren, S. Xie, L. Wang, Z. Miao, and J. Peng, "Adaptive sliding mode control for uncertain aero-engine distributed control system," (Chinese), *J. Aerosp. Power*, vol. 32, no. 8, pp. 2032–2040, 2017.

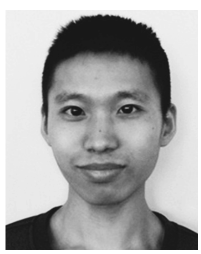
**YU ZHANG** was born in Taiyuan, Shanxi, China, in 1990. He received the B.S. degree in aeronautical engineering from Air Force Engineering University, Xi'an, Shaanxi, China, in 2013, where he is currently pursuing the Ph.D. degree in aerospace engineering with Aeronautics and Astronautics Engineering Institute.

His main research interests include aero-engine control system, networked control systems, sliding mode control, and robust control.

**SHOUSHENG XIE** was born in Yunan, Guangdong, China, in 1959. He received the B.S. and M.S. degrees in aerospace engineering from Air Force Engineering University, Xi'an, Shaanxi, China, in 1985, and the Ph.D. degree in aerospace engineering from Northwestern Polytechnical University, Xi'an, in 1998.

He is currently a Professor with the Department of Aeronautical Engineering, Air Force Engineering University. He is the author of seven books, more than 100 articles, and holds 14 patents. His research interests include the aircraft propulsion system integrated control, networked control systems, distributed control, and fault diagnosis.

**LITONG REN** was born in Dezhou, Shandong, China, in 1987. He received the B.S. and M.S. degrees in aeronautical engineering from Air Force Engineering University, Xi'an, Shaanxi, China, in 2013, and the Ph.D. degree in aerospace engineering from Air Force Engineering University, Xi'an, in 2017, where he is currently a Lecturer with the Department of Aeronautical Engineering. His research interests include aerospace control theory and control system fault detection.



**LEDI ZHANG** was born in Changchun, Jilin, China, in 1990. He received the B.S. and M.S. degrees in aeronautical engineering from the Aviation University of Air Force, Changchun, in 2015, and the Ph.D. degree in aerospace engineering from Air Force Engineering University, Xi'an, Shaanxi, China, in 2019.

He is currently a Lecturer with the Department of Aeronautical Engineering, Air Force Engineering University. His research interests include distributed control system, observer design, and fault-tolerant control.