

Stochastic stabilisation of wireless networked control systems with lossy multi-packet transmission

ISSN 1751-8644

Received on 27th April 2018

Revised 18th September 2018

Accepted on 6th December 2018

doi: 10.1049/iet-cta.2018.5255

www.ietdl.org

Yun-Bo Zhao¹ ✉, Tao Huang¹, Yu Kang², Xugang Xi³¹College of Information Engineering, Zhejiang University of Technology, Hangzhou 310023, People's Republic of China²Department of Automation, University of Science and Technology of China, Hefei 230027, People's Republic of China³School of Automation, Hangzhou Dianzi University, Hangzhou 310018, People's Republic of China

✉ E-mail: ybzhao@ieee.org

Abstract: The stochastic stabilisation of networked control systems is investigated with a special focus on the lossy multi-packet transmission in the wireless communication context. The resulting partially available system states due to multi-packet transmission are firstly reconstructed at the controller, and the sufficient conditions for stochastic stability are then given for the closed-loop system, which finally leads to a controller design method with explicit consideration of multi-packet transmission. The proposed theoretical results are verified by both numerical and TrueTime-based examples.

1 Introduction

Networked control systems (NCSs) are control systems in which the data is transmitted by some form of non-control-dedicated communication networks such as the internet. The data can be in the sensor-to-controller channel, the controller-to-actuator channel or both. The introduction of the communication network in NCSs brings benefits including, e.g. reduced implementation cost, easier maintenance, capability of remote operation, and so forth [1, 2], making NCSs the desired control framework for many practical fields such as the intelligent factory, internet of things, unmanned vehicles etc. [3–5]. These advantages are, unfortunately, obtained at the cost of possible performance degradation due to the introduction of the communication networks to control systems, since the communication network inevitably causes imperfect data transmission, by introducing the network constraints including data packet loss, network-induced delay, disorder of received packets etc. [6]. To overcome these difficulties, tremendous efforts have been made by the scientists and engineers in the field, and various control theories and techniques have been applied to NCSs, e.g. robust control theory [7–9], approaches based on Markov jump systems [10–12], predictive control [13–15], and joint design schemes between communication and control [16–18], to name a few.

Among all these available studies, one particular scenario has not been investigated sufficiently to date, i.e. the so-called ‘multi-packet transmission’, where the sensing data or the control data at each step is transmitted via multiple separate data packets [19–21]. This scenario can be caused by different reasons, as discussed in [22]. One reason can be due to the multiple, distributed sensors [23, 24], whose sensing data cannot be aggregated and integrated into a single data packet to transmit. Another reason can be due to either the sensing or control data, exceeding the size of the allowed effective load of the communication network, which is then forced to be divided into smaller portions. We acknowledge that the second cause of multi-packet transmission may be of more importance theoretically, since typical NCSs should use data networks, and data networks are supposed to allow rather large data packets. For example, the MAC frame of IEEE 802.11 allows a maximum load of 2312 bytes, which far exceeds the needed size of most NCS applications. On the other hand, with the increasing use of wireless communications in NCSs, more and more devices can now be connected in a distributed fashion using various wireless communication technologies, and due to the distributed structure of the system, sensors may have to send their sensing data

independently and separately, thus making the first cause of multi-packet transmission even worse. Therefore, more efforts have to be made to achieve a reliable control performance in the presence of multi-packet transmission, especially in the context of using wireless communications in NCSs.

To date, we have seen a large number of control algorithms solving a packet loss issue with the single-packet transmission. For example, the controller design problem was considered in [25] for systems with both bounded random packet loss and channel uncertainty; a non-cooperative linear quadratic game was used to design the optimal decentralised state-feedback controllers in [26] for a wireless sensor and actuator network with stochastic delays and packet losses; the output feedback guaranteed cost control issue was considered in [27] for NCSs with random packet dropouts and time delays; a novel state space model was proposed in [28] with the corresponding model predictive tracking control algorithm to deal with systems subject to packet loss and uncertainties. Also, a lot more other works, which are not listed here, have made great achievements in dealing with packet loss under single-packet transmission.

Despite these aforementioned achievements, most employed approaches for single-packet transmission are, however, not directly applicable to multi-packet transmission. Indeed, multi-packet transmission results in a unique feature in terms of packet loss, i.e. different parts of the data are transmitted independently and can thus face different network conditions and hence different loss probabilities. This unique feature fails most conventional studies for single-packet transmission and requires dedicated design and analysis for multi-packet transmission. Furthermore, as mentioned earlier multi-packet transmission can be more popular in the context of wireless communications while being wireless means packet loss can be more dominant than packet delay, yet conventional NCS studies more focus on the effect of time delays. Some pioneering works have been reported for multi-packet NCSs. To name a few, an observer-based networked predictive control approach was proposed for discrete-time NCSs with multi-packet transmission in [29] to compensate for multiple delays and packet dropouts; by modelling the packet loss process as a Markov chain the stochastic stability conditions were given in [30]; by modelling the packet loss process as an independent and identical Bernoulli process, the state feedback controller was designed in [31]; continuous-time NCS with lossy multi-packet transmissions was considered in [32], and so forth. These studies have explicitly considered the unique feature of multi-packet transmission, but most of them have more focused on the controller side while

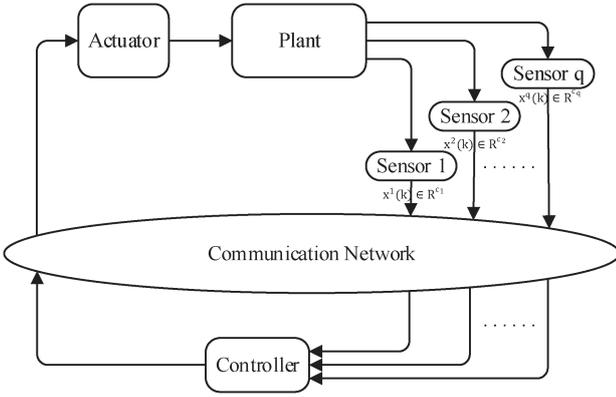


Fig. 1 Networked control systems with multi-packet transmission

ignoring the possibility of improving the system performance by actively compensating the missing partial data, which thus motivates the present work.

In this work, we investigate the design and analysis of NCSs with packet dropout incurred by multi-packet transmission, inspired by the general context of using wireless communications in NCSs. After carefully formulating the problem, sufficient conditions are proposed to guarantee the stochastic stability of the closed-loop NCS, with a corresponding controller design method. The approach adopts a particular active compensation scheme by taking better advantage of the available information at the controller side, making it distinct from most existing approaches.

The remainder of the paper is organised as follows. Section 2 formulates the problem. The stability conditions and controller design method are given in Section 3. Both numerical and TrueTime-based examples are provided in Section 4 to illustrate the effectiveness of the proposed method. Finally, the paper is concluded in Section 5.

2 Problem formulation

The considered NCS with lossy multi-packet transmission is first formulated, followed by the closed-loop system description with a state reconstruction method to deal with the multi-packet transmission.

2.1 NCSs with lossy multi-packet transmission

The considered system setup is depicted in Fig. 1, where the data transmission in the controller-to-actuator channel is perfect while the sensor-to-controller channel suffers the so-called ‘multi-packet transmission’, i.e. q independent sensors sample the plant individually and then send the sensing data via the communication channel independently.

The plant is described as follows:

$$\tilde{x}(k+1) = \tilde{A}\tilde{x}(k) + \tilde{B}u(k), \quad (1)$$

where $\tilde{x}(k) \in \mathbb{R}^n$, $u(k) \in \mathbb{R}^m$, $\tilde{A} \in \mathbb{R}^{n \times n}$, and $\tilde{B} \in \mathbb{R}^{n \times m}$ are the state vector, the control input, and the system and input matrices, respectively.

In our system setting as depicted in Fig. 1, multiple sensors are present and each sensor may correspond to several components of the system states, but the system state is not necessarily organised corresponding to the order of the sensors. We realise later that it can help the design and analysis if the data from each sensor can be readily extracted from the system state, i.e. the following new state vector $x(k)$ obtained by organising the states contributed by each sensor sequentially is preferred rather than the original state $\tilde{x}(k)$

$$\begin{aligned} x(k) &= [(x^1(k))^T, (x^2(k))^T, \dots, (x^q(k))^T]^T \\ &= [x_1(k) \ x_2(k) \ \dots \ x_n(k)]^T, \end{aligned} \quad (2)$$

where $x^i(k)$ is the c_i -dimensional sensing data contributed by sensor i , $\sum_{i=1}^q c_i = n$, and $x_i(k)$ is the i th element of $x(k)$.

Notice that the n elements in $\tilde{x}(k)$ is in fact a permutation of the elements in $x(k)$. Suppose the i th element in $\tilde{x}(k)$ is the p_i element in $x(k)$, then $\tilde{x}(k)$ can also be rewritten in terms of $x_i(k)$, as follows:

$$\tilde{x}(k) = [x_{p_1}(k) \ x_{p_2}(k) \ \dots \ x_{p_n}(k)]^T. \quad (3)$$

From (2) and (3), it is clear that a finite exchange sequence of the elements in $\tilde{x}(k)$ must exist, which reorders $\tilde{x}(k)$ to form $x(k)$. Denote such a sequence by $\mathcal{P} = \{[i_1, i_2]\}$, where $[i_1, i_2]$ means at the i th step exchange the i_1 th and i_2 th elements in $\tilde{x}(k)$. Notice that every exchange operation can be recorded by a swapping matrix, i.e. after exchange operation $[i_1, i_2]$ on $\tilde{x}(k)$ the new state is $S_{i_1, i_2}\tilde{x}(k)$, where the swapping matrix S_{i_1, i_2} is obtained by swapping the i_1 th and i_2 th columns of the n -dimensional identity matrix.

It is then obtained that

$$x(k) = S\tilde{x}(k), \quad (4)$$

where $S = \prod_{[i_1, i_2] \in \mathcal{P}} S_{i_1, i_2}$.

It is clear that the linear transformation in (4) does not change any system behaviours such as stability and robustness, and hence in what follows we may safely assume that the states have already been reordered according to the sequence of the sensor without loss of generality.

The plant considered is of the following form:

$$x(k+1) = Ax(k) + Bu(k). \quad (5)$$

Remark 1: The key technique to obtain the finite exchange sequence \mathcal{P} is in fact increasingly sorting the sequence $p_i, i = 1, 2, \dots, n$, which can be solved by almost any classic sorting algorithms. For example, using the bubble sorting algorithm, we may from p_1 sequentially exchange the positions of the two adjacent elements, making the smaller one ahead. Other algorithms work alike. From this discussion, we may also notice \mathcal{P} is not unique.

These sorting algorithms guarantee an invertible transformation from $\tilde{x}(k)$ to $x(k)$, which is necessary for our later discussion. If, on the other hand, what we are interested in is only to obtain $x(k)$ from $\tilde{x}(k)$, the following simple equation may work well:

$$x(k) = [I_n \ I_n] \prod_{i=1}^n S_{i, n+p_i} [\tilde{x}^T(k) \ 0_n]^T,$$

where I_n is the n -dimensional identity matrix, 0_n is the n -dimensional zero vector, and the $2n$ -dimensional swapping matrix $S_{i, j}$ is defined similarly.

As mentioned earlier, the data sent from each sensor to the controller can be lost. As assumed in [31], the successful transmission of the sensing data of sensor i , i.e. $x^i(k)$, can be modelled as an independent Bernoulli process, denoted by α_k^i , i.e.

$$\alpha_k^i = \begin{cases} 1 & x^i(k) \text{ is successfully transmitted,} \\ 0 & \text{otherwise,} \end{cases} \quad (6)$$

where

$$\mathbb{E}\{\alpha_k^i\} = \alpha_i, \quad (7a)$$

$$\mathbb{E}\{(\alpha_k^i - \alpha_i)^2\} = \sigma_i^2. \quad (7b)$$

Remark 2: By (6) the sensing data from each sensor can be lost. Suppose at time k at the controller side the sensing data from

sensor i experience τ_i consecutive data loss, then the system state at the controller side can be written as follows:

$$x(k) = [x^1(k - \tau_1) x^2(k - \tau_2) \dots x^q(k - \tau_q)].$$

The above equation explains the key challenge caused by multi-packet transmission in NCSs: the system state at the controller side is ‘partially lost’, as different parts of the state at the controller side (corresponding to different sensors) may have different delays, i.e. generally $\tau_i \neq \tau_j$, $i \neq j$, $i, j = 1, 2, \dots, q$.

Remark 3: In our problem setting, the sensor-to-controller channel contains multiple lossy data links while the data transmission in the controller-to-actuator channel is perfect. This seemingly strange system structure is proposed to model the scenario of using wireless communications in NCSs. Indeed, with wireless communication in Fig. 1, the multiple sensors share the same wireless channel and have to compete with each other to send their data to the controller, but the controller can regard the wireless channel as private to it and thus the control data can be sent directly to the actuator without competition. From the control system perspective this system setting thus produces a lossy sensor-to-controller channel, but a perfect controller-to-actuator channel, given that the performance of the wireless channel itself is guaranteed (or at least with a very low loss rate thus ignorable safely). The use of wireless communication also explains why we do not consider a delay in our system setting: the considered system is supposed to be physically located in a relatively small area due to the use of wireless communication, and in this case, data packet dropout is more dominant than delay [33].

2.2 Closed-loop system with state reconstruction

Conventional methods of dealing with partial packet loss due to multi-packet transmission (as detailed in Remark 2), either by using the last available whole packet or replacing the unknown parts by 0, are clearly conservative [33–35]. In this work, a system state reconstruction method as proposed in [22] is used to reconstruct the partial lost information.

Taking consideration of the q sensors, the system matrix A and input matrix B in (5) can be divided into $q \times q$ and $q \times 1$ block matrices, as follows:

$$A = \begin{bmatrix} A^{11} & A^{12} & \dots & A^{1q} \\ A^{21} & A^{22} & \dots & A^{2q} \\ \vdots & \vdots & \ddots & \vdots \\ A^{q1} & A^{q2} & \dots & A^{qq} \end{bmatrix}, \quad B = \begin{bmatrix} B^1 \\ B^2 \\ \vdots \\ B^q \end{bmatrix}, \quad (8)$$

where $A^{ij} \in \mathbb{R}^{c_i \times c_j}$, $i = 1, 2, \dots, q$, $j = 1, 2, \dots, q$, and $B^i \in \mathbb{R}^{c_i \times m}$. Then

$$x^i(k+1) = \sum_{j=1}^q A^{ij} x^j(k) + B^i u(k), \quad (9)$$

where $i = 1, 2, \dots, q$.

Due to the independent packet loss of each sensor, usually not all the states $x^j(k)$, $j = 1, 2, \dots, q$, are available to the controller. To deal with this packet loss, we use the following technique to reconstruct the state of sensor i at time $k+1$ using (9), denoted by

$$\hat{x}^i(k+1) = \sum_{j=1}^q A^{ij} \hat{x}^j(k) + B^i u(k), \quad (10)$$

where $\hat{x}^i(k) = \alpha_k^i x^i(k) + (1 - \alpha_k^i) \hat{x}^i(k)$, i.e. the estimated partial state is only used in the absence of the actual partial state.

Let $\bar{x}^T(k) = [(\hat{x}^1(k))^T, (\hat{x}^2(k))^T, \dots, (\hat{x}^q(k))^T]$, we obtain

$$\bar{x}(k) = \Theta_k x(k) + (I - \Theta_k) \hat{x}(k), \quad (11)$$

where

$$\Theta_k = \begin{bmatrix} \alpha_k^1 I^{c_1 \times c_1} & & & \\ & \alpha_k^2 I^{c_2 \times c_2} & & \\ & & \ddots & \\ & & & \alpha_k^q I^{c_q \times c_q} \end{bmatrix}. \quad (12)$$

It follows that (10) can be written in a compact form

$$\hat{x}(k+1) = A \bar{x}(k) + B u(k). \quad (13)$$

A state feedback controller can now be designed as $u(k) = K \bar{x}(k)$ or alternatively

$$u(k) = K \Theta_k x(k) + K(I - \Theta_k) \hat{x}(k), \quad (14)$$

where $K \in \mathbb{R}^{m \times n}$ is the feedback gain matrix.

By defining $\eta^T(k) = [x^T(k), \hat{x}^T(k)]$, from (5), (13) and (14) the closed-loop NCS is obtained as

$$\eta(k+1) = \Phi_k \eta(k), \quad (15)$$

where

$$\Phi_k = \begin{bmatrix} A + BK\Theta_k & BK(I - \Theta_k) \\ (A + BK)\Theta_k & (A + BK)(I - \Theta_k) \end{bmatrix}. \quad (16)$$

3 Closed-loop stability and controller design

In this section, new sufficient conditions for the stochastic stability of the closed-loop system are given, and the controller in (14) is designed.

3.1 Closed-loop stability

The following stochastic stability definition is introduced.

Definition 1: The closed-loop system in (15) is said to be stochastic stability if for any initial value $\eta(0)$ it holds that

$$\mathbb{E} \left\{ \sum_{k=0}^{\infty} \|\eta(k)\|^2 \right\} < \infty. \quad (17)$$

Theorem 1: For given feedback gain K , the closed-loop system in (15) is stochastic stability if there exist scalars $\gamma_1, \gamma_2 > 0$ and positive definite matrices $P > 0$ and $Q > 0$ such that

$$(BK)^T P B K < \gamma_1 I, \quad (18a)$$

$$(A + BK)^T Q (A + BK) < \gamma_2 I, \quad (18b)$$

$$M_1^T P M_1 + M_2^T Q M_2 + M_3^T M_3 + M_4^T M_4 - \begin{bmatrix} P & 0 \\ 0 & Q \end{bmatrix} < 0, \quad (18c)$$

where

$$M_1 = [A + BK\Xi \quad BK(I - \Xi)], \quad (19a)$$

$$M_2 = [(A + BK)\Xi \quad (A + BK)(I - \Xi)], \quad (19b)$$

$$M_3 = \left[\sqrt{\gamma_1} \sum_{i=1}^q \sigma_i E^i \quad -\sqrt{\gamma_1} \sum_{i=1}^q \sigma_i E^i \right], \quad (19c)$$

$$M_4 = \left[\sqrt{\gamma_2} \sum_{i=1}^q \sigma_i E^i \quad -\sqrt{\gamma_2} \sum_{i=1}^q \sigma_i E^i \right], \quad (19d)$$

with α_i and σ_i being defined in (7a) and (7b), respectively,

$$\mathbb{E}\{\Theta_k\} = \Xi = \text{diag}\{\alpha_1 I^{c_1 \times c_1}, \dots, \alpha_q I^{c_q \times c_q}\}$$

and

$$E^i = \text{diag}\{0, \dots, 0, I_i^{c_i \times c_i}, 0, \dots, 0\}.$$

Proof: We prove the theorem by the following three steps. Firstly, we define the following Lyapunov functional:

$$V(k) = x(k)^T P x(k) + \hat{x}(k)^T Q \hat{x}(k). \quad (20)$$

From (16), we can obtain that

$$\begin{aligned} & \mathbb{E}\{V(k+1)|x(k), \hat{x}(k)\} - V(k) \\ &= \mathbb{E}\{x(k+1)^T P x(k+1) + \hat{x}(k+1)^T Q \hat{x}(k+1) \\ & \quad |x(k), \hat{x}(k)\} - V(k) \\ &= \mathbb{E}\{[(A + BK\Theta_k)x(k) + BK(I - \Theta_k)\hat{x}(k)]^T P \\ & \quad \times [(A + BK\Theta_k)x(k) + BK(I - \Theta_k)\hat{x}(k)] + [(A \\ & \quad + BK)\Theta_k x(k) + (A + BK)(I - \Theta_k)\hat{x}(k)]^T Q [(A \\ & \quad + BK)\Theta_k x(k) + (A + BK)(I - \Theta_k)\hat{x}(k)]\} - x(k)^T \\ & \quad \times P x(k) - \hat{x}(k)^T Q \hat{x}(k) \\ &= \mathbb{E}\{[(A + BK\Xi)x(k) + BK(\Theta_k - \Xi)x(k) + BK(I \\ & \quad - \Xi)\hat{x}(k) + BK(\Xi - \Theta_k)\hat{x}(k)]^T P [(A + BK\Xi)x(k) \\ & \quad + BK(\Theta_k - \Xi)x(k) + BK(I - \Xi)\hat{x}(k) + BK(\Xi \\ & \quad - \Theta_k)\hat{x}(k)] + [(A + BK)\Xi x(k) + (A + BK)(\Theta_k \\ & \quad - \Xi)x(k) + (A + BK)(I - \Xi)\hat{x}(k) + (A + BK)(\Xi \\ & \quad - \Theta_k)\hat{x}(k)]^T Q [(A + BK)\Xi x(k) + (A + BK)(\Theta_k \\ & \quad - \Xi)x(k) + (A + BK)(I - \Xi)\hat{x}(k) + (A + BK)(\Xi \\ & \quad - \Theta_k)\hat{x}(k)]\} - x(k)^T P x(k) - \hat{x}(k)^T Q \hat{x}(k). \end{aligned} \quad (21)$$

Secondly, we consider the expression obtained above item by item.

Noting that $\mathbb{E}\{\Theta_k - \Xi\} = 0$, we have

$$\mathbb{E}\{V(k+1)|x(k), \hat{x}(k)\} - V(k) = V_1(k) + \mathbb{E}\{V_2(k)\} - V_3(k),$$

where

$$V_1(k) = \eta(k)^T \Lambda_1 \eta(k),$$

$$V_3(k) = \eta(k)^T \Lambda_3 \eta(k),$$

with $\Lambda_3 = \begin{bmatrix} P & 0 \\ 0 & Q \end{bmatrix}$ and $\Lambda_1 = M_1^T P M_1 + M_2^T Q M_2$. M_1 and M_2 are given by (19a) and (19b), respectively.

Define

$$\mathbb{E}\{V_2(k)\} = \mathbb{E}\{V_{21}(k) + V_{22}(k)\},$$

where

$$\mathbb{E}\{V_{21}(k)\} = \mathbb{E}\{[BK(\Theta_k - \Xi)x(k) + BK(\Xi - \Theta_k)\hat{x}(k)]^T \\ \times P[BK(\Theta_k - \Xi)x(k) + BK(\Xi - \Theta_k)\hat{x}(k)]\}, \quad (22)$$

$$\mathbb{E}\{V_{22}(k)\} = E\{[(A + BK)(\Theta_k - \Xi)x(k) + (A + BK)(\Xi \\ - \Theta_k)\hat{x}(k)]^T Q [(A + BK)(\Theta_k - \Xi)x(k) + (A \\ + BK)(\Xi - \Theta_k)\hat{x}(k)]\}. \quad (23)$$

Let $y(k) = x(k) - \hat{x}(k)$. From (12) and the fact that the random variables $\alpha_k^1, \dots, \alpha_k^q$ are independent of each other, it holds that

$$\begin{aligned} \mathbb{E}\{V_{21}(k)\} &= \sum_{i=1}^q y(k)^T \sigma_i^2 E^i (BK)^T P (BK) E^i y(k) \\ &\leq \gamma_1 \sum_{i=1}^q (\sigma_i E^i y(k))^T (\sigma_i E^i y(k)) \\ &= \gamma_1 \sum_{i=1}^q (\sigma_i E^i y(k))^T (\sigma_i E^i y(k)) \\ & \quad + \gamma_1 \sum_{i=1, j=1, i \neq j}^q (\sigma_i E^i y(k))^T (\sigma_j E^j y(k)) \\ &= \gamma_1 \left(\sum_{i=1}^q \sigma_i E^i y(k) \right)^T \left(\sum_{i=1}^q \sigma_i E^i y(k) \right), \end{aligned} \quad (24)$$

where $\gamma_1 = \lambda_{\max}((BK)^T P BK)$ and $\lambda_{\max}((BK)^T P BK) > 0$ denotes the maximal eigenvalue of $(BK)^T P BK$. It follows that

$$\mathbb{E}\{V_{21}(k)\} \leq \eta(k)^T \Lambda_{21} \eta(k), \quad (25)$$

where $\Lambda_{21} = M_3^T M_3$ with M_3 being given by (19c).

Similarly, the following inequality holds:

$$\mathbb{E}\{V_{22}(k)\} \leq \eta(k)^T \Lambda_{22} \eta(k), \quad (26)$$

where $\Lambda_{22} = M_4^T M_4$ with M_4 being defined in (19d) and $\gamma_2 = \lambda_{\max}((A + BK)^T Q (A + BK))$, $\lambda_{\max}((A + BK)^T Q (A + BK)) > 0$ denotes the maximal eigenvalue of $(A + BK)^T Q (A + BK)$.

Lastly, we consider back to the Lyapunov functional.

From above, we obtain

$$\mathbb{E}\{V(k+1)|x(k), \hat{x}(k)\} - V(k) \leq \eta(k)^T \Lambda \eta(k), \quad (27)$$

where

$$\Lambda = \Lambda_1 + \Lambda_{21} + \Lambda_{22} - \Lambda_3.$$

It follows from (18c) that $\Lambda < 0$, and hence

$$\begin{aligned} & \mathbb{E}\{V(k+1)|x(k), \hat{x}(k)\} - V(k) \\ & \leq \eta(k)^T \Lambda \eta(k) \leq -\beta \eta(k)^T \eta(k), \end{aligned} \quad (28)$$

where $\beta = \lambda_{\min}(-\Lambda)$, $\lambda_{\min}(-\Lambda)$ denotes the maximal eigenvalue of $-\Lambda$.

From (28), for any $t > 0$, we have

$$\mathbb{E}\{V(\eta(k))\} - \{V(\eta(0))\} \leq -\beta \sum_{k=0}^t \mathbb{E}\{\eta(k)^T \eta(k)\} \quad (29)$$

and for any $m > 0$

$$\begin{aligned} \sum_{k=0}^m \mathbb{E}\{\eta(k)^T \eta(k)\} &\leq \frac{1}{\beta} (\{V(\eta(0))\} - \mathbb{E}\{V(\eta(k))\}) \\ &\leq \frac{1}{\beta} \{V(\eta(0))\} < \infty, \end{aligned} \quad (30)$$

which completes the proof by Definition 1. \square

3.2 Controller design

Suppose $\varepsilon_i, i = 1, 2, \dots, m$ are the non-zero singular values of B . Then by singular value decomposition there exist orthogonal matrices $U \in \mathbb{R}^{m \times n}$ and $V \in \mathbb{R}^{m \times m}$ such that

$$B = U \begin{bmatrix} \Sigma \\ 0 \end{bmatrix} V^T, \quad (31)$$

- §1. Reorder the system states using (4) to obtain the transformed system description as in (5).
 §2. q sensors sample the plant and send them to the controller independently via the communication network.
 §3. At the controller side.

- §3.1 Reconstruct the system state using (13).
 §3.2 Calculate the control action using (14) based on the reconstructed states.
 §3.3 Send the control signal to the actuator.

- §4. Apply the control action to the plant.

Fig. 2 Algorithm 1

where $\Sigma = \text{diag}\{\varepsilon_1, \varepsilon_2, \dots, \varepsilon_m\}$.

We have the following lemma from [37].

Lemma 1: Given B in (31). For matrix P of the following structure:

$$P = U \begin{bmatrix} P_1 & 0 \\ 0 & P_2 \end{bmatrix} U^T, \quad (32)$$

there exists matrix $Z \in \mathbb{R}^{m \times m}$ such that $PB = BZ$ and $Z = V\Sigma^{-1}P_1\Sigma V^T$, where $P_1 \in \mathbb{R}^{m \times m}$ and $P_2 \in \mathbb{R}^{(n-m) \times (n-m)}$.

Theorem 2: The closed-loop system in (15) with the controller in (14) is stochastic stable, if there exists scalars $\gamma_1, \gamma_2 > 0$ and a positive definite matrix $P \in \mathbb{R}^{n \times n}$ with the structure as in (32) and a matrix M with an appropriate dimension such that (see (33a))

$$\begin{bmatrix} -\gamma_1 I & * \\ BM & -P \end{bmatrix} < 0, \quad (33b)$$

$$\begin{bmatrix} -\gamma_2 I & * \\ PA + BMC & -P \end{bmatrix} < 0. \quad (33c)$$

Furthermore, the control gain is given by

$$K = V\Sigma^{-1}P_1^{-1}\Sigma V^T M. \quad (34)$$

Proof: Firstly, we rewrite (18c) as follows by using the Schur complement:

$$\begin{bmatrix} -P & * & * & * & * & * \\ 0 & -Q & * & * & * & * \\ A + BK\Xi & BK(I - \Xi) & -P^{-1} & * & * & * \\ (A + BK)\Xi & (A + BK)(I - \Xi) & 0 & -Q^{-1} & * & * \\ \sqrt{\gamma_1} \sum_{i=1}^q \sigma_i E^i & -\sqrt{\gamma_1} \sum_{i=1}^q \sigma_i E^i & 0 & 0 & -I & * \\ \sqrt{\gamma_2} \sum_{i=1}^q \sigma_i E^i & -\sqrt{\gamma_2} \sum_{i=1}^q \sigma_i E^i & 0 & 0 & 0 & -I \end{bmatrix} < 0. \quad (35)$$

Then pre- and post-multiplying (35) with $\text{diag}\{I, I, P, Q, I, I\}$, we have

$$\begin{bmatrix} -P & * & * & * & * & * \\ 0 & -Q & * & * & * & * \\ PA + PBK\Xi & PBK(I - \Xi) & -P & * & * & * \\ (QA + QBK)\Xi & (QA + QBK)(I - \Xi) & 0 & -Q & * & * \\ \sqrt{\gamma_1} \sum_{i=1}^q \sigma_i E^i & -\sqrt{\gamma_1} \sum_{i=1}^q \sigma_i E^i & 0 & 0 & -I & * \\ \sqrt{\gamma_2} \sum_{i=1}^q \sigma_i E^i & -\sqrt{\gamma_2} \sum_{i=1}^q \sigma_i E^i & 0 & 0 & 0 & -I \end{bmatrix} < 0. \quad (36)$$

Similarly the constraint conditions (18a) and (18b) in Theorem 1 are transformed into (33b) and (33c), respectively.

According to Lemma 1, there exists a non-singular matrix P having the structure (32) such that $PB = BP_1$. Let $M = P_1K$ and $Q = P$. Thus (36) is transformed into (33a) and we obtain the control gain matrix as in (34) by using the feasp solver available in the MATLAB LMI toolbox to solve the constraint conditions. This completes the proof. \square

3.3 Algorithm

Now, the proposed algorithm (see Fig. 2) for closed-loop NCS can be organised as follows.

Remark 4: In practical applications noises, disturbances and model uncertainties are always part of the system and may severely affect the system performance. In the present work, we work on the nominal system since what matters to us the most is how to deal with a multi-packet transmission in NCSs but not others. It is also well known that most controllers do guarantee certain stability margins and are thus immune to noises and disturbances to a certain extent. Furthermore, the real-time system state reconstruction is done at every step, which can also reduce the negative effects of noises and disturbances, recalling the receding horizon concept.

Remark 5: In the present work, we more focus on the control scheme design and closed-loop stability analysis but not any further optimal indices of the control system. However, such a performance index may readily be included within the H_2 , H_∞ or other control frameworks, and will be our future research directions.

4 Simulation examples

Two examples are presented to illustrate the effectiveness of the proposed method in this study. One is totally numerical, mainly to show the effectiveness in a well-defined system setting, and the other is based on the widely used TrueTime toolbox, for the verification in a relatively more realistic setting with more uncertainties.

$$\begin{bmatrix} -P & * & * & * & * & * \\ 0 & -P & * & * & * & * \\ PA + BM\Xi & BM(I - \Xi) & -P & * & * & * \\ (PA + BM)\Xi & (PA + BM)(I - \Xi) & 0 & -P & * & * \\ \sqrt{\gamma_1} \sum_{i=1}^q \sigma_i E^i & -\sqrt{\gamma_1} \sum_{i=1}^q \sigma_i E^i & 0 & 0 & -I & * \\ \sqrt{\gamma_2} \sum_{i=1}^q \sigma_i E^i & -\sqrt{\gamma_2} \sum_{i=1}^q \sigma_i E^i & 0 & 0 & 0 & -I \end{bmatrix} < 0, \quad (33a)$$

4.1 Numerical example

Consider the following system, which adds a disturbance term $w(k)$ to the considered plant in (5):

$$x(k+1) = Ax(k) + Bu(k) + w(k), \quad (37)$$

where the system matrices are taken from [31]

$$A = \begin{bmatrix} 1.7 & 0.4 & 1.8 \\ -2 & -0.8 & -3.1 \\ -3.2 & -1.5 & -1.2 \end{bmatrix}, \quad B = \begin{bmatrix} 0.5 & 0 \\ 0 & 0.5 \\ 1 & 0.5 \end{bmatrix},$$

and $w(k)$ is a Gauss white noise with the variance being 0.01.

In our system setting, each of the three system states is sampled and sent by one of the three sensors, and the three sensors are independent of each other. The data packet sent from each sensor has the same probability of 0.25 to be lost, and the initial system state is set as $x(0) = [-5, 4, 5]^T$.

Using Theorem 2, we obtain the following controller gain:

$$K = \begin{bmatrix} -0.4673 & 0.0169 & -1.9479 \\ 5.0939 & 2.2630 & 5.8885 \end{bmatrix},$$

With conventional state feedback, the control signal has to be designed using the last received whole system state, where the controller gain can be designed as follows [31]:

$$K_C = \begin{bmatrix} -1.0990 & 0.4548 & -0.5830 \\ 1.8160 & 0.7814 & 1.6750 \end{bmatrix}.$$

The state trajectories and control signals with both controllers are shown in Figs. 3 and 4. It is seen that the controller designed by our approach leads to a stable trajectory, while the conventional control method causes instability of the system.

To further illustrate the effectiveness of our proposed method, with the above system structure, we perform extensive simulations over five parameter sets containing different packet loss probabilities of the sensors. The results are summarised in Table 1. It is clearly seen that our approach generally leads to a more stable closed-loop system, even in the statistical sense.

4.2 TrueTime-based example

A TrueTime-based example is illustrated. TrueTime [38] provides a MATLAB/Simulink-based toolbox to simulate most existing communication networks with the support of a wide range of communication protocols. One may combine the TrueTime toolbox with the original support of MATLAB/Simulink for dynamic systems to simulate NCSs in a more realistic way. This capability makes the toolbox very popular in the NCSs community.

In the simulation, the plant in (5) with the following system matrices borrowed from [39] is considered

$$A = \begin{bmatrix} 0.9850 & -0.0348 & -0.0248 \\ 0.0050 & 0.9999 & -0.0001 \\ 0.0001 & 0.0050 & 1.0000 \end{bmatrix}, \quad B = \begin{bmatrix} 0.0050 \\ 0.0001 \\ 0.0002 \end{bmatrix}, \quad (38)$$

where the three states are sampled by three independent sensors, respectively. The initial system state is set as $x(0) = [-1, -1, -1]^T$.

The wireless network is implemented using the TrueTime 2.0 toolbox, whose system diagram is shown in Fig. 5. We use the 802.11b (wireless local area network) protocol with the data rate being 80,000 bits/s and the minimum frame size being 20 bits. The packet loss probability is set as 0.2.

Using Theorem 2, we obtain the controller gain as follows:

$$K = [-1.0322 \quad -0.3377 \quad 0.3454].$$

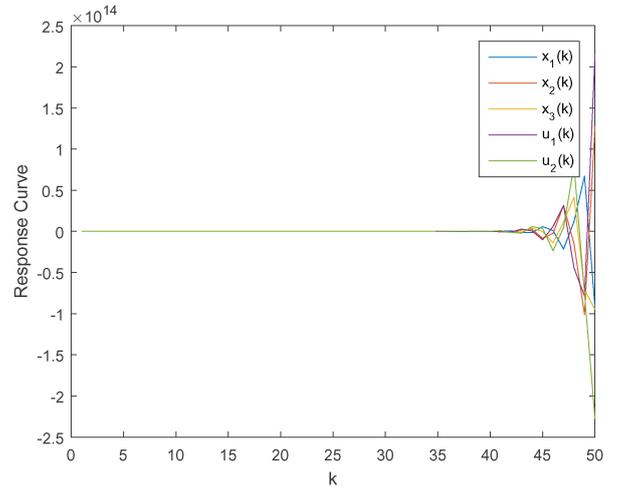


Fig. 3 System responses using control method from [31]

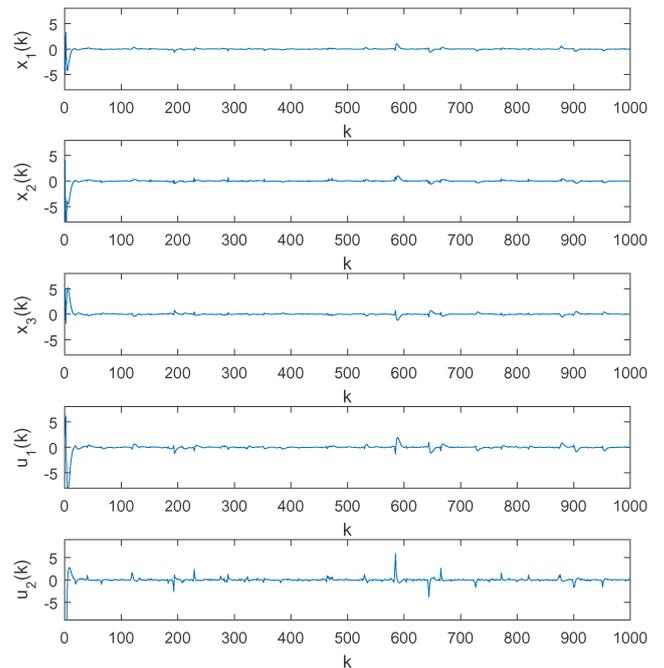


Fig. 4 System responses using our proposed approach

Table 1 Performance comparison using a large number of simulations. 50 simulations are performed for each parameter set

Dropout probabilities $[\alpha_1 \alpha_2 \alpha_3]$	Percentage of resulting stable trajectories	
	Our approach, %	Conventional method, %
[0.1 0.1 0.1]	100	84
[0.2 0.2 0.2]	100	64
[0.25 0.25 0.25]	100	42
[0.3 0.3 0.3]	100	28
[0.35 0.35 0.35]	98	14

With the predictive-based control method as proposed in [40], the controller gain is designed as follows [40]:

$$K_{PC} = [1.8441 \quad 2.2036 \quad 0.8488].$$

The state trajectories using our approach and the method in [40] are compared in Fig. 6, which clearly shows that our proposed approach fast stabilises the system while the method in [40] may lead to dramatic fluctuations.

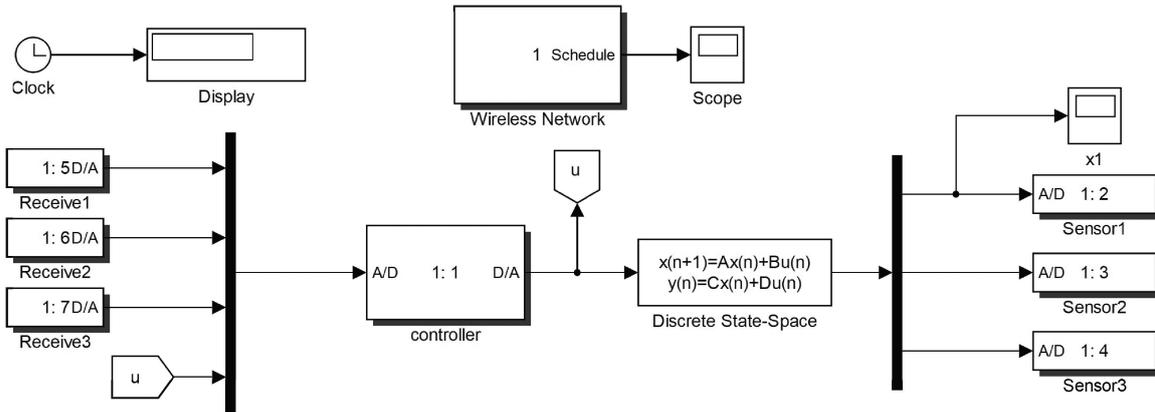


Fig. 5 TrueTime-based simulation diagram

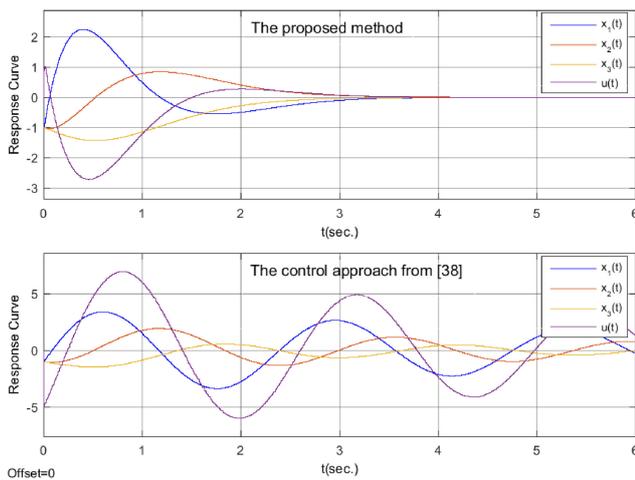


Fig. 6 State responses using both approaches in TrueTime

In the above two examples, our approach is shown to be superior to the conventional methods. This should be mainly due to the fact that our approach takes better use of the incomplete state information at the controller side.

5 Conclusion

Inspired by using wireless communications in networked control systems, multi-packet transmission is investigated within the stochastic stability framework. The sufficient conditions for ensuring the stochastic stability as well as the corresponding controller design method are given. The present work considers only packet dropout in order to simplify the problem setting, which may be extended in our future works. We are witnessing the technical integration of wireless communications and control systems, which have enabled many significant applications in the Industry 4.0 era. As one fundamental technical issue underlying such integration, we believe the efficient treatment of multi-packet transmission in NCSs will receive more attention in the future.

6 Acknowledgments

This work was supported in part by the National Natural Science Foundation of China under grant no. 61673350, and in part by the 1000-Talent Youth Program and the Zhejiang 1000-Talent Program.

7 References

- [1] Hu, W., Liu, G.P., Zhou, H.: 'Web-based 3-d control laboratory for remote real-time experimentation', *IEEE Trans. Ind. Electron.*, 2013, **60**, (10), pp. 4673–4682
- [2] Cloosterman, M.B.G., Hetel, L., Wouw, N.V.D., *et al.*: 'Controller synthesis for networked control systems', *Automatica*, 2010, **46**, (10), pp. 1584–1594
- [3] You, K.Y., Xie, L.H.: 'Survey of recent progress in networked control systems', *Acta Autom. Sinica*, 2013, **39**, (2), pp. 101–117
- [4] Ma, C.Q., Zhang, J.F.: 'On formability of linear continuous-time multi-agent systems', *J. Syst. Sci. Complex*, 2012, **25**, (1), pp. 13–29
- [5] Ma, C.Q., Zhang, J.F.: 'Consensus control for leader-following multi-agent systems with measurement noises', *J. Syst. Sci. Complex*, 2010, **23**, (1), pp. 35–49
- [6] Zhang, X.M., Han, Q.L., Yu, X.: 'Survey on recent advances in networked control systems', *IEEE Trans. Ind. Inf.*, 2016, **12**, (5), pp. 1740–1752
- [7] Shi, Y., Yu, B.: 'Robust mixed H_2/H_∞ control of networked control systems with random time delays in both forward and backward communication links', *Automatica*, 2011, **47**, (4), pp. 754–760
- [8] Wang, Z., Member, S., Yang, F., *et al.*: 'Robust H_∞ control for networked systems with random packet losses', *IEEE Trans. Syst. Man Cybern. B, Cybern.*, 2007, **37**, (4), pp. 916–924
- [9] Elahi, A., Alfi, A.: 'Finite-time H_∞ control of uncertain networked control systems with randomly varying communication delays', *ISA Trans.*, 2017, **69**, pp. 65–88
- [10] Zhang, X.M., Han, Q.L.: 'Network-based H_∞ filtering using a logic jumping-like trigger', *Automatica*, 2013, **49**, (5), pp. 1428–1435
- [11] Xiong, J., Lam, J.: 'Stabilization of linear systems over networks with bounded packet loss', *Automatica*, 2007, **43**, (1), pp. 80–87
- [12] Gao, H., Meng, X., Chen, T.: 'Stabilization of networked control systems with a new delay characterization', *IEEE Trans. Autom. Control*, 2008, **53**, (9), pp. 2142–2148
- [13] Liu, G.P.: 'Predictive control of networked multiagent systems via cloud computing', *IEEE Trans. Cybern.*, 2017, **47**, (8), pp. 1–8
- [14] Dinh, T.Q., Ahn, K.K., Marco, J.: 'A novel robust predictive control system over imperfect networks', *IEEE Trans. Ind. Electron.*, 2017, **64**, (2), pp. 1751–1761
- [15] Tan, C., Yin, X., Liu, G.P., *et al.*: 'Prediction-based approach to output consensus of heterogeneous multi-agent systems with delays', *IET Control Theory Appl.*, 2018, **12**, (1), pp. 20–28
- [16] Peters, E.G.W., Quevedo, D.E., Fu, M.: 'Controller and scheduler codesign for feedback control over IEEE 802.15.4 networks', *IEEE Trans. Control Syst. Technol.*, 2016, **24**, (6), pp. 2016–2030
- [17] Henriksson, E., Quevedo, D.E., Peters, E.G.W., *et al.*: 'Multiple-loop self-triggered model predictive control for network scheduling and control', *IEEE Trans. Control Syst. Technol.*, 2015, **23**, (6), pp. 2167–2181
- [18] Zhao, Y.B., Kim, J., Liu, G.P.: 'Error bounded sensing for packet-based networked control systems', *IEEE Trans. Ind. Electron.*, 2011, **58**, (3), pp. 1980–1989
- [19] Ijesjanin, M., Quevedo, D.E., Nesi, D.: 'Packetized MPC with dynamic scheduling constraints and bounded packet dropouts', *Automatica*, 2014, **50**, (3), pp. 784–797
- [20] Zhu, X.-L., Yang, G.-H.: 'State feedback controller design of networked control systems with multiple-packet transmission', *Int. J. Control*, 2009, **82**, (1), pp. 86–94
- [21] Wen, D.-L., Yang, G.-H.: 'State feedback control of continuous-time networked control systems in multiple-packet transmission'. Proc. Chinese Control and Decision Conf., Guilin, June 2009, pp. 582–587
- [22] Zhao, Y.B., Kim, J., Yang, G.H., *et al.*: 'Model-based compensation for multi-packet transmission in networked control systems'. Proc. IEEE Conf. on Decision and Control and European Control Conf., Orlando, December 2011, pp. 3136–3141
- [23] Wu, D., Wu, J., Chen, S.: 'Separation principle for networked control systems with multiple-packet transmission', *IET Control Theory Appl.*, 2011, **5**, (3), pp. 507–513
- [24] Yang, S.H., Liang, T.H., Wu, J.L.: 'Output feedback multiple-packet transmission networked control systems'. Proc. IEEE Int. Conf. on Networking, Sensing and Control, Evry, April 2013, pp. 615–620
- [25] Na, X., Zhan, Y., Xia, Y., *et al.*: 'Control of networked systems with packet loss and channel uncertainty', *IET Control Theory Appl.*, 2016, **10**, (17), pp. 2251–2259
- [26] Wang, Z., Wang, X., Liu, L.: 'Stochastic optimal linear control of wireless networked control systems with delays and packet losses', *IET Control Theory Appl.*, 2016, **10**, (7), pp. 742–751
- [27] Qiu, L., Yao, F., Xu, G., *et al.*: 'Output feedback guaranteed cost control for networked control systems with random packet dropouts and time delays in

- forward and feedback communication links', *IEEE Trans. Autom. Sci. Eng.*, 2016, **13**, (1), pp. 284–295
- [28] Lu, R., Xu, Y., Zhang, R.: 'A new design of model predictive tracking control for networked control system under random packet loss and uncertainties', *IEEE Trans. Ind. Electron.*, 2016, **63**, (11), pp. 6999–7007
- [29] Song, H., Liu, G.P., Yu, L.: 'Networked predictive control of uncertain systems with multiple feedback channels', *IEEE Trans. Ind. Electron.*, 2013, **60**, (11), pp. 5228–5238
- [30] Wu, J., Chen, T.: 'Design of networked control systems with packet dropouts', *IEEE Trans. Automat. Control*, 2007, **52**, (7), pp. 1314–1319
- [31] Hu, S., Yan, W.Y.: 'Stability of networked control systems under a multiple-packet transmission policy', *IEEE Trans. Autom. Control*, 2008, **53**, (7), pp. 1706–1711
- [32] Wu, D., Wu, J., Chen, S.: 'Robust H_∞ control for networked control systems with uncertainties and multiple-packet transmission', *IET Control Theory Appl.*, 2010, **4**, (5), pp. 701–709
- [33] Guo, G., Jin, H.: 'A switching system approach to sensor and actuator assignment for stabilisation via limited multi-packet transmitting channels', *Int. J. Control*, 2011, **84**, (1), pp. 78–93
- [34] Seiler, P.: 'Coordinated control of unmanned aerial vehicles'. PhD thesis, University of California at Berkeley, 2001
- [35] Zhu, X.-L., Yang, G.-H.: 'Stability analysis and state feedback control of networked control systems with multi-packet transmission'. Proc. American Control Conf., Washington, June 2008, pp. 3133–3138
- [36] Jianning, L., Hongye, S., Zhengguang, W., *et al.*: 'Stabilization of wireless networked control system with multi-packet transmission policy'. Proc. Chinese Control Conf., Hefei, July 2012, pp. 5770–5774
- [37] Sun, J., Chen, J.: 'Networked predictive control for systems with unknown or partially known delay', *IET Control Theory Appl.*, 2014, **8**, (18), pp. 2282–2288
- [38] Cervin, A., Henriksson, D., Ohlin, M.: '*TrueTime 2.0 beta 5-reference manual*' (Automatic Control Department, Lund University, Lund, Sweden, 2010)
- [39] Chen, C.-T.: '*Linear system theory and design*' (Oxford University Press, New York, USA, 1998, 3rd edn.)
- [40] Liu, G.-P.: 'Consensus and stability analysis of networked multiagent predictive control systems', *IEEE Trans. Cybern.*, 2017, **47**, (4), pp. 1114–1119