Automatica 52 (2015) 255–259

Contents lists available at ScienceDirect

Automatica

journal homepage: www.elsevier.com/locate/automatica

Technical communique

New results on stabilization of networked control systems with packet disordering[☆]

Andong Liu[a,1], Wen-an Zhang[a], Li Yu[a], Steven Liu[b], Michael Z.Q. Chen[c]

[a] Department of Automation, Zhejiang University of Technology, Zhejiang Provincial United Key Laboratory of Embedded Systems, Hangzhou 310023, PR China

[b] Department of Electrical and Computer Engineering, University of Kaiserslautern, Kaiserslautern 67663, Germany

[c] Department of Mechanical Engineering, The University of Hong Kong, Hong Kong

ARTICLE INFO

Article history:
Received 24 March 2014
Received in revised form
5 August 2014
Accepted 23 November 2014
Available online 31 December 2014

Keywords:
Networked control systems
Network-induced delays
Packet disordering
Packet dropouts

ABSTRACT

In this paper, the stabilization problem is studied for a class of networked control systems (NCSs) with delays, packet disordering and packet dropouts. A new packet reordering method is presented to deal with packet disordering and to choose the newest control input. A relationship between the reordered packet over two consecutive sampling intervals is given for the NCS with both time delays and packet dropouts. A sufficient condition for the NCS to be exponentially stable is presented by using the average dwell-time method. Finally, an illustrative example is given to demonstrate the effectiveness of the proposed method.

1. Introduction

Network-induced delay and packet dropout are two main problems in networked control system (NCS), and have attracted much research interest, see for example Gao and Chen (2007), Garcia and Antsaklis (2013), Xiong and Lam (2007), Yang and Han (2013), Zhang, Gao, and Kaynak (2013) and references therein. Compared with constant delays, the time-varying one is more difficult to treat, especially, when the delay is larger than one sampling period (long delay). Since the delay may be larger than one sampling period, more than one control signals may arrive at the actuator during one sampling interval. Moreover, the transmission of data packets may not necessarily follow a "first send first arrive" principle (Zhao, Kim, Liu, & Rees, 2012). This means that the newest control signal may arrive at the actuator before the older one, this is the so-called packet disordering problem Liu, Yu, and Zhang (2011a,b). The modeling and stability analysis of NCSs with packet disordering has attracted much research attention, see for example Cloosterman et al. (2010); Cloosterman, van de Wouw, Heemels, and Nijmeijer (2009), Li, Zhang, Yu, and Cai (2011), Zhang and Yu (2008, 2009). Due to limited network transmission capacity, packet dropouts are usually inevitable. Some results on NCSs concerning both delay and packet dropout issues were presented in García-Rivera and Barreiro (2007), Wang, Shen, Shu, and Wei (2012), Xia, Liu, Fu, and Rees (2009) and Yang, Liu, Shi, Thomas, and Basin (2014); Yang, Shi, Liu, and Gao (2011). It should be pointed out that the packet disordering problem is not considered in the aforementioned results for NCSs in the presence of delays and packet dropouts. In Zhang and Han (2012), H_∞ filtering problem has been investigated for NCSs with delays, packet disordering and packet dropouts. However, the explicit expression for how to choose the newest data signal was not given in Zhang and Han (2012).

On the other hand, due to the time-varying delays and packet dropouts, the number of available control signals at the actuator vary over different sampling intervals. Therefore, the NCS is naturally a switched system with a group of subsystems describing various system dynamics on different sampling intervals (Hetel, Daafouz, & Lung, 2008; Lin & Antsaklis, 2005; Wang, Liu, Wang, Rees, & Zhao, 2010). In response to the above discussion, we investigate the stability analysis problem for a class of NCSs with delays and packet dropouts, and focus on solving the packet disordering problem and the switched dynamic caused by long

☆ This work was supported by the National Natural Science Foundation of China under Grants 61273117, 61374053, and 61403344. The material in this paper was not presented at any conference. This paper was recommended for publication in revised form by Associate Editor Jie Chen under the direction of Editor André Tits.

E-mail addresses: lad-andrew@hotmail.com (A. Liu), wazhg@hotmail.com (W.-a. Zhang), lyu@zjut.edu.cn (L. Yu), sliu@eit.uni-kl.de (S. Liu), mzqchen@gmail.com (M.Z.Q. Chen).

1 Tel.: +86 571 85290531; fax: +86 571 85290567.

http://dx.doi.org/10.1016/j.automatica.2014.12.006
0005-1098/

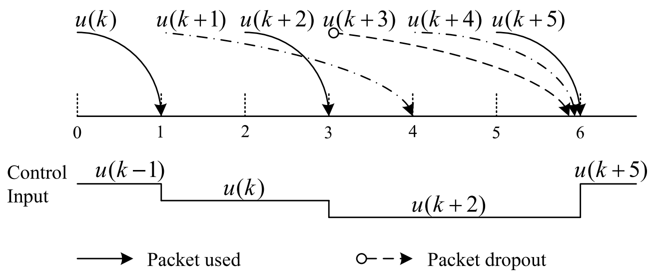

Fig. 1. Time diagram of signal transmitting in the NCS.

delay and packet dropout. The main contributions of the paper are as follows: (1) A new packet reordering method is proposed to deal with packet disordering and to choose the newest control signal. (2) A relationship between the reordered packet over two consecutive sampling intervals is given for the NCS with both network-induced delays and packet dropouts. (3) A sufficient condition for exponential stability of the NCS is derived by using the average dwell-time method. Finally, an example is given to demonstrate the effectiveness of the proposed method.

2. Modeling of the NCS

The plant in the NCS is described by the following continuous-time linear system model

$$\dot{x}(t) = A_p x(t) + B_p u(t) \tag{1}$$

where $x(t) \in R^n$ is the system state, $u(t) \in R^m$ is the control input, A_p and B_p are constant matrices with appropriate dimensions. The sensor is time-driven with sampling period T, the controller is event-driven, the actuator is time-driven and has a receiving buffer containing the most recent data packet from the controller, time delay τ_k is assumed to be bounded by $0 \le \tau_k \le dT$, where d is a known finite integer. Since the actuator is time-driven, one has $\tau_k \in \mathcal{N}_1 = \{0,\ 1,\ 2, \ldots, d\}\,T$.

For simplicity, we first consider only the delay issue. Since the network-induced delay may be larger than one sampling period, more than one control signals may arrive at the actuator during one sampling interval, but only one control signal is adopted by the actuator. Then, the problem is how to choose the newest control signal. Moreover, packet disordering problem arises when the newest packet arrives at the actuator before the older ones. A time diagram of the signal transmission is illustrated in Fig. 1, in which it is assumed that $d = 3$, and at most four control signals may arrive at the actuator during one sampling interval. Furthermore, control signal $u(k+2)$ arrives at the actuator earlier than $u(k+1)$. Thus, $u(k+2)$ is adopted at time $(k+3)T$. This phenomenon is called packet disordering. In order to use the newest control signal, $u(k+1)$ will be discarded.

By the aforementioned analysis, it can be seen that the adopted control signal may take values in $\{u(k-d+1), \ldots, u(k-1), u(k)\}$ at sampling instant kT, which will result in $d+1$ different dynamics of the system. Moreover, during the sampling interval $[kT,\ (k+1)T)$, the system dynamics are actually determined by $\{\tau_{k-d+1}, \ldots, \tau_{k-1},\ \tau_k\}$. Therefore, we use a vector $\tau(k) = [\tau_{k-d+1}, \ldots, \tau_{k-1}, \tau_k]$ to represent the control signal that is applied at the actuator, and define a vector-valued function $f:\ \tau(k) \to \sigma(k)$ to map $\tau(k)$ into a scalar $\sigma(k) \in \mathcal{N}_2 = \{0,\ 1, \ldots, d\}$. The expression of $\sigma(k)$ is given in detail as follows

$$\sigma(k) = \begin{cases} 0, & \tau_k = 0, \tau_{k-j} \in \mathcal{N}_1, \quad j = 1, 2, \ldots, d-1 \\ 1, & \tau_k \ge T, \tau_{k-1} \le T, \tau_{k-j} \in \mathcal{N}_1, \\ & j = 2, \ldots, d-1 \\ 2, & \tau_{k-i} \ge (i+1)T,\ i = 0, 1, \tau_{k-2} \le 2T, \\ & \tau_{k-j} \in \mathcal{N}_1, \quad j = 3, \ldots, d-1 \\ \vdots & \vdots \\ d, & \tau_{k-i} \ge (i+1)T, \quad i = 0, 1, \ldots, d-1 \end{cases} \tag{2}$$

and then the adopted control signal at time kT is $u(k - \sigma(k))$. Eq. (2) presents an explicit logic expression for how to choose the newest control signal and eliminating the impact of packet disordering. Also, $\sigma(k)$ can be given concisely as follows

$$\sigma(k) = \min\{i | \tau_{k-i} - iT \le 0,\ i = 0, 1, \ldots, d\}. \tag{3}$$

Then, we have the following proposition.

Proposition 1. *If $\sigma(k) = r$, then $\sigma(k+1) \le r + 1$.*

Proof. Let $\sigma(k+1) = \min\left\{j | \tau_{k+1-j} - jT \le 0,\ j = 0, 1, \ldots, d\right\}$. If $\sigma(k) = r$, it follows from (3) that $\tau_{k-r} - rT \le 0$ and $\tau_{k-i} - iT > 0$ for $i < r$. From the above analysis, it can be obtained that $\tau_{k+1-(r+1)} - (r+1)T < 0$, which infers that $\sigma(k+1) \le r+1$. This completes the proof. □

Remark 1. The packet disordering can be effectively eliminated by applying the mechanism given in (3). Assuming that the packets of time $(k-i-1)T$ and $(k-i)T$ are disordering, then one obtains $\tau_{k-i-1} \ge \tau_{k-i} + 2T$. Therefore, if $\tau_{k-i-1} - (i+1)T \le 0$, it must be $\tau_{k-i} - iT < 0$, which means that the actuator will use the newest packet of time $(k-i)T$ and discard packet of time $(k-i-1)T$.

Let θ_k be the number of consecutive packet dropout with $\theta_k \in [0,\ s]$, $s \le d$ at time kT, and $\tau_k = dT$ if the packet of time kT is lost. A time diagram of the signal transmitting with both time delay and packet dropout is illustrated in Fig. 1. In view of (2), it is difficult to handle packet dropout in a unified framework because $\sigma(k) \in \{0,\ 1, \ldots, d+s\}$. Inspired by Eq. (3), the following result can be derived with both time delays and packet dropouts

$$\sigma(k) = r + \theta_{k-r} \tag{4}$$

where $r = \min\{i | \tau_{k-i} - (i + \theta_{k-i})T \le 0,\ i = 0, 1, \ldots, d\}$.

Remark 2. If there is no packet dropout during sampling interval $[(k-d)T,\ kT]$, one has $\theta_{k-i} = 0$ for $i \in \{0,\ 1, \ldots, d\}$. Then (4) equals to (3). If $\theta_{k-r} \ne 0$, the packets of time $(k-r)T$ to $(k-r-\theta_{k-r})T$ are dropout and $\tau_{k-i} - (i+\theta_{k-i})T > 0$ for $0 \le i < r$, which means that the packet of time $(k-i)T$ for $0 \le i < r$ is not successfully delivered during interval $((k-r)T,\ kT]$. Therefore, Eq. (4) shows that the actuator always uses the newest packet during interval $[(k-d-s)T,\ kT]$. Similar to Proposition 1, the following proposition can be obtained.

Proposition 2. *If $\sigma(k) = r + \theta_{k-r}$, $0 \le \theta_{k-r} \le s$, then $\sigma(k+1) \le r + \theta_{k-r} + 1$ and $\sigma(k+1) \ne r+2, \ldots, r+\theta_{k-r}$.*

Proof. By using a similar method to the proof of Proposition 1, it can be easily obtained that $\sigma(k+1) \le r + \theta_{k-r} + 1$. Next, we will prove $\sigma(k+1) \ne r+2, \ldots, r+\theta_{k-r}$.

If the packet of time $(k-r)T$ is successfully delivered during interval $[(k-d)T,\ kT]$, it means $\theta_{k-r} = 0$. Thus, $\sigma(k+1) \le r+1$. If the packet of time $(k-r)T$ is dropout, then one has

$$r + \theta_{k-r} = r + 1 + \theta_{k-r-1} = \cdots = r + \theta_{k-r} + \theta_{k-r-\theta_{k-r}} \tag{5}$$

which implies that $\theta_{k-r-\theta_{k-r}} = 0$. If the packet at time $(k-r+1)T$ is dropout, the following result can be obtained by (5)

$$r - 1 + \theta_{k-r+1} = r + \theta_{k-r}. \tag{6}$$

It follows from (6) that $\sigma(k) = r - 1 + \theta_{k-r+1}$. However, it is inconsistent with condition $\sigma(k) = r + \theta_{k-r}$. Therefore, if the packet of time $(k-r)T$ is dropout, the packet at time $(k-r+1)T$ must be successfully delivered.

Let $\sigma(k+1) = \bar{j} + \theta_{k+1-\bar{j}}$. If $\tau_{k+1-j} - (j + \theta_{k+1-j})T > 0$ for $0 \le j \le r$, then $\bar{j} = r+1$ and $\sigma(k+1) = r + \theta_{k-r} + 1$. If there exists a j for $0 \le j \le r$ satisfied $\tau_{k+1-j} - (j + \theta_{k+1-j})T \le 0$, it follows from (5) and (6) that $\bar{j} + \theta_{k+1-\bar{j}} \le r$ because the packet at time $(k-r+1)T$ must be successfully delivered. Therefore, $\sigma(k+1) \le r$ and $\sigma(k+1) \ne r+1,\ r+2, \ldots, r+\theta_{k-r}$. The proof is thus completed. □