Control Vector Selection with Delay Estimation in Wireless Networked Control Systems

Keisuke Nakashima*, Takahiro Matsuda*, Masaaki Nagahara†, and Tetsuya Takine*
*Department of Information and Communications Technology, Graduate School of Engineering, Osaka University
Suita, Osaka 565–0871, Japan
Email: k-nakashima@post.comm.eng.osaka-u.ac.jp, {matsuda, takine}@comm.eng.osaka-u.ac.jp
†Institute of Environmental Science and Technology, The University of Kitakyushu
Kitakyushu, Fukuoka 808–0135, Japan
Email: nagahara@kitakyu-u.ac.jp

***Abstract*—In a sampled-data wireless networked control system, *bursty* packet losses and random delays could cause unstable behavior of controlled objects. In order to compensate these network-induced effects, we extend a *packetized predictive control* method, where multiple future control vectors are generated from estimated states of controlled objects and they are packed into a single packet transmitted over the wireless network. In the extended method, more control vectors are generated than in the conventional method, so that the controller should select appropriate control vectors among a set of those control vectors so as to be packed into a single packet. We thus consider a control vector selection scheme, where control vectors in each packet are selected based on the estimated average round-trip delays on the wireless network.**

I. Introduction

Wireless networked control systems (WNCSs) are control systems whose components are connected through wireless networks. In this paper, we study a *sampled-data WNCS*, where continuous-time controlled objects are controlled by one discrete-time controller through a CSMA/CA-based wireless network. In the WNCS, controlled objects could be unstable due to *bursty* packet losses and random delays occurred on the wireless network [1]. In order to compensate bursty packet losses, a *packetized predictive control* (PPC) method has been proposed in [2], where future control vectors are generated in the receding horizon manner and they are packed into a single packet for transmission. Note here that this method assumes that delays are negligible or constant. In order to compensate random delays, on the other hand, packet-based control (PBC) framework has been proposed in [3], where control vectors are generated based on the estimated states at multiple time instants within each sampling period, and they are packed into a single packet for transmission.

In order to compensate both bursty losses and random delays, we consider an extended PPC method by combining the original PPC method with the above-mentioned PBC framework. In a naive implementation of the extended PPC method, however, many control vectors are generated, so that they cannot be packed into a single packet. Therefore, the controller has to select appropriate control vectors among a set of those control vectors so as to be packed into a single packet. In this paper, we propose a control vector selection scheme for the extended PPC method, where the controller estimates the average of round-trip delays and selects appropriate control vectors based on the estimated average delays.

II. Control Vector Selection in PPC

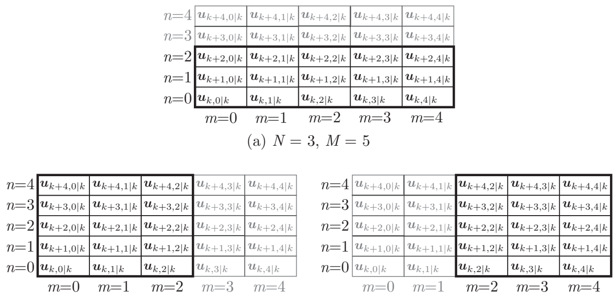
A. Packetized Predictive Control Method

Let $\boldsymbol{x}(t)$ $(t \geq 0)$ denote a state vector of a controlled object at continuous time t. With matrices $\boldsymbol{A}$, $\boldsymbol{B}$, and $\boldsymbol{C}$, we model a sampled-data WNCS with sampling periods of length d [4]:

$$\boldsymbol{x}_{k+1} = \boldsymbol{A}\boldsymbol{x}_k + \boldsymbol{B}\boldsymbol{u}_k + \boldsymbol{w}_k, \quad \boldsymbol{y}_k = \boldsymbol{C}\boldsymbol{x}_k + \boldsymbol{v}_k, \quad (1)$$

for $k \in \mathcal{Z}^+ = \{0, 1, \ldots\}$, where $\boldsymbol{x}_k = \boldsymbol{x}(kd)$, $\boldsymbol{y}_k$, and $\boldsymbol{u}_k$ denote a state vector, an observation vector, and a control vector, respectively, at time kd, and $\boldsymbol{w}_k$ and $\boldsymbol{v}_k$ denote noise vectors at time kd. In what follows, we refer to a set of a controlled object, sensors/actuators, and the transmitter/receiver of packets as a *controlled object unit*, and a set of the controller, the state estimator, and the transmitter/receiver of packets as the *controller unit*.

We define $\boldsymbol{u}_{\ell|k}$ $(\ell \geq k)$ as a future control vector at time ℓd. Note that $\boldsymbol{u}_{\ell|k}$ is generated based on the estimated state $\hat{\boldsymbol{x}}_{\ell|k}$ of $\boldsymbol{x}_l$ from $\boldsymbol{y}_k$ and information available at time kd. Let $\mathcal{N}_0 = \{0, 1, \ldots, N_0 - 1\}$, where N_0 denotes the total number of future control vectors generated at each time. In the PPC method [2], $\hat{\boldsymbol{x}}_{k|k}$ is estimated by (1), and $\boldsymbol{u}_{k+n|k}$'s $(n \in \mathcal{N}_0)$ are obtained by solving an optimization problem to minimize a finite-horizon cost function online. After that, they are packed into a single packet and transmitted through the wireless network. When a controlled object unit receives the control packet successfully, it stores all control vectors and inputs them into its controlled object in order whenever the subsequent control packets are lost consecutively.



(a) $N = 3$, $M = 5$

(b) $N = 5$, $M = 3$

(c) $N = 5$, $M = 3$, $m^* = 3$

Fig. 1. Examples of a control vector selection for $NM = 15$ and $N_0M_0 = 25$.

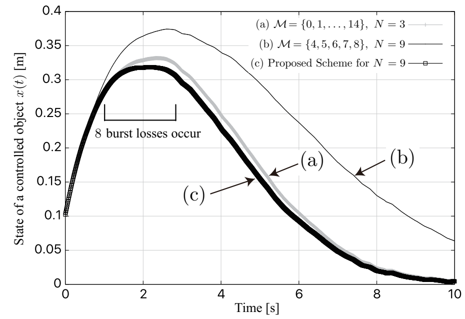


Fig. 2. The transient response for three control selection patterns, where $x(t) = 0$ is a target.

B. The Extended PPC

Even in the sampled-data WNCS without packet losses, random delays may cause unstable behavior of controlled objects due to the *inconsistency problem* [5]. We focus on a pair of the controller unit and a controlled object unit. Let Δ_k ($k \in \mathcal{Z}^+$) denote the time interval from the instant when an observation vector $\boldsymbol{y}_k$ is generated to the instant when the controlled object unit receives the corresponding control packet. We call Δ_k the round-trip delay and assume $\Delta_k < d$.

Let $\mathcal{M}_0 = \{0, 1, \ldots, M_0 - 1\}$ and $\delta = d/M_0$, where M_0 denotes the number of slots within one sampling period. When the controller unit receives $\boldsymbol{y}_k$, it generates a set $\hat{\mathcal{X}}_k = \{\hat{\boldsymbol{x}}_{k,m|k} \mid m \in \mathcal{M}_0\}$ of estimated states $\hat{\boldsymbol{x}}_{k,m|k}$ of $\boldsymbol{x}(kd + m\delta)$. Next, from $\hat{\mathcal{X}}_k$, the controller generates a set $\mathcal{P}_k = \{\boldsymbol{u}_{k+n,m|k} \mid n \in \mathcal{N}_0, m \in \mathcal{M}_0\}$ of future control vectors $\boldsymbol{u}_{k+n,m|k}$ to be used at time $kd + m\delta$. $\mathcal{P}_k$ is then packed into a single packet pkt_k and transmitted. When the controlled object unit receives pkt_k, it measures the round-trip delay Δ_k and stores a set $\boldsymbol{U}_{k,m_k|k} = \{\boldsymbol{u}_{k+n,m_k|k} \mid n \in \mathcal{N}_0\}$ of appropriate control vectors, where $m_k = \lceil \Delta_k/\delta \rceil \in \mathcal{M}_0$. If $m_k \notin \mathcal{M}_0$, a set of zero vectors is stored instead. $\boldsymbol{U}_{k,m_k|k}$ is used for the predictive control. For example, if the next control packet pkt_{k+1} is not received, $\boldsymbol{u}_{k+1,m_k|k}$ is inputted to the controlled object at time $(k+1)d + m_k\delta$.

C. Control Vector Selection

In the extended PPC method, $\mathcal{P}_k$ could not be packed into a single packet pkt_k because the size of $\mathcal{P}_k$ is proportional to N_0M_0. In such a case, the controller has to select a set $\mathcal{P}^*_k = \{\boldsymbol{u}_{k+n,m|k} \mid n \in \mathcal{N}, m \in \mathcal{M}\} \subset \mathcal{P}_k$ of control vectors for single-packet transmission, where $\mathcal{N} = \{0, 1, \ldots, N-1\}$ ($N \leq N_0$), $\mathcal{M} \subseteq \mathcal{M}_0$, and $M \triangleq |\mathcal{M}|$. Fig. 1 (a) and (b) show examples of a control vector selection for $NM = 15$ and $N_0 = M_0 = 5$. The set of control vectors in Fig. 1 (a) is robust to random delays while the set of control vectors in Fig. 1 (b) is robust to bursty packet losses.

In order to achieve high stability of the controlled object, we propose a control vector selection scheme with delay estimation. In the proposed scheme, the controlled object unit transmits m_{k-1} by piggybacking on $\boldsymbol{y}_k$, and the controller unit computes the average $RTT_{\text{avg}}(k)$ of round-trip delays with m_{k-1}. Suppose that M is an odd number. The controller unit obtains $m^* = \lceil RTT_{\text{avg}}(k)/\delta \rceil$ and sets $\mathcal{M} = \{p, p+1, \ldots, q\}$, where p and q are determined by

- $p = 0$, $q = M - 1$, if $m^* < (M-1)/2$,
- $p = m^* - (M-1)/2$, $q = m^* + (M-1)/2$, if $(M-1)/2 \leq m^* \leq M_0 - 1 - (M-1)/2$,
- $p = M_0 - M$, $q = M_0 - 1$, if $m^* > M_0 - 1 - (M-1)/2$.

Fig. 1 (c) shows an example of a control vector selection by the proposed scheme for $N = 5$, $M = 3$, $m^* = 3$.

III. Performance Evaluation

In the simulation experiment, one controller unit and 20 controlled object units are connected through a CSMA/CA-based wireless network. We set $NM = 45$ and evaluate the extended PPC method for three selection patterns. Fig. 2 shows an example of the transient response when 8 consecutive control packet losses occur, where (a) and (b) correspond to the performance for fixed $\mathcal{M}$ and N, and (c) corresponds to the performance of the proposed scheme. This result shows the effectiveness of our proposed scheme because $|x(t)|$ in (c) is the smallest.

Acknowledgments

This research was supported in part by Strategic Information and Communications R&D Promotion Programme (SCOPE) of Ministry of Internal Affairs and Communications No. 172310003 and JSPS KAKENHI Grant No. JP16K00124, 15H02668, and 16KK0134.

References

[1] R. A. Gupta and M. Y. Chow, *IEEE Transactions on Industrial Electronics*, vol. 57, no. 7, pp. 2527–2535, Jul. 2010.

[2] M. Nagahara et al., *IEEE Transactions on Automatic Control*, vol. 59, no. 7, pp. 1899–1905, Jul. 2014.

[3] Y. Zhao et al., *IEEE Transactions on Control Systems Technology*, vol. 17, no. 4, pp. 859–865, Jul. 2009.

[4] Y. Yamamoto, *IEEE Transactions on Automatic Control*, vol. 39, no. 4, pp. 703–713, Apr. 1994.

[5] K. Nakashima et al., *Proc. IEEE PIMRC 2017*, Oct. 2017.