2011 50th IEEE Conference on Decision and Control and
European Control Conference (CDC-ECC)
Orlando, FL, USA, December 12-15, 2011

Model-Based Compensation for Multi-Packet Transmission in Networked Control Systems

Yun-Bo Zhao, Jongrae Kim, Guang-Hong Yang and Guo-Ping Liu

***Abstract*— The sensing data is usually transmitted simultaneously from the sensor to the controller in conventional control systems. However, in networked control systems it is possible that a set of sensing data is transmitted via multiple separate data packets due to the multiple, geographically dispersed sensors. This scenario, referred to as "multi-packet transmission", brings to the system different delays for different parts of the sensing data. Within the packet-based control framework for networked control systems, a novel control structure is proposed. The negative effects of multi-packet transmission are effectively dealt with by first reconstructing the sensing data at the controller side and then compensating for the communication constraints using the packet-based control approach. Numerical examples illustrate the effectiveness of the proposed approach.**

I. Introduction

Networked control systems (NCSs) are those control systems that are closed via some form of communication networks. In most cases, the "communication networks" are referred to those that are not particularly designed for the control purpose, the most important one of which, is the Internet. By introducing the Internet into the control system, one can achieve the capability of remote and distributed control at a lower cost, with easy maintenance and flexible structure redesign. These advantages are obtained however at the cost of possible unreliable control performance since, unlike as assumed in conventional control systems, the Internet introduces to the NCSs imperfect communication links. These imperfect communication links may degrade the system performance of NCSs or even destabilize the system under certain conditions, thus constituting the core issue in the study of NCSs [1], [2].

In terms of the imperfect communication links, network-induced delay and data packet dropout have been the most studied in the literature to date. In this work we notice another issue seldom addressed before, that is, different parts of the sensing data at a single step may experience different delays or dropouts. Two reasons contribute to this issue. Firstly, the sensing data at a single step can be too much for the communication network to be encoded into a single data packet, resulting in the split of the data into several data packets for the transmission purpose. Secondly, the sensors in NCSs can be geographically dispersed as is often seen in today's large distributed systems, and this distributed structure of measurement forces the sensing data to be obtained and transmitted from multiple sensors. Both scenarios, resulting from either the limited size of data packet or the geographically dispersed sensors, are referred to equally as "multi-packet transmission" due to their similar consequence. This consequence of producing different communication characteristics for different parts of data, can potentially affect the system performance seriously, and yet have hardly been addressed in the literature to date [3]–[5].

Indeed, the only several available studies on multi-packet transmission have been inclined to first accept the negative effect incurred (without reasoning how this occurs), then design the controller and analyze the system performance from only the control engineering perspective [4], [5]. Our current work, on the contrary, is motivated by the fact that the system performance could be further improved by taking advantage of both advanced control algorithms and proper communication protocols. Specifically, this improvement is achieved by first reconstructing the sensing data at the controller side and then compensating for the communication constraints using the packet-based control approach. In this way we are able to show that the negative effects of multi-packet transmission can be effectively and actively compensated for. Furthermore, this co-design scheme is flexible to admit all the existing control algorithms to be used, making it a unified framework from the control engineering perspective.

II. NCSs with multi-packet transmission

The cause and the negative effects of multi-packet transmission are first discussed, followed by a brief review of related work in this area. The system setup studied in this paper is given at the end of this section.

A. *Multi-packet transmission in NCSs*

In the present work "multi-packet transmission" refers to the scenario where a set of data that is supposed to be transmitted simultaneously in conventional control systems, is however transmitted via multiple separate data packets in NCSs. These data can be, for example, the sensing data from the sensor side or the control data from the controller side. Generally speaking, two reasons contribute to the occurrence of this scenario.

1) The geographically dispersed control components, typically the distributed multiple sensors, enable the scenario where the sensing data has to be sampled and transmitted

This work was supported by EPSRC research grant EP/G036195/1, and in part by the National Natural Science Foundation of China under Grant 60934006.

Yun-Bo Zhao and Jongrae Kim are with Division of Biomedical Engineering, University of Glasgow, Glasgow, G12 8QQ, UK {Yun-Bo.Zhao,Jongrae.Kim}@glasgow.ac.uk

Guang-Hong Yang is with College of Information Science and Engineering, Northeastern University, Shenyang, Liaoning, 110004, China yangguanghong@ise.neu.edu.cn

Guo-Ping Liu is with Faculty of Advanced Technology, University of Glamorgan, Pontypridd, CF37 1DL, UK and also with CTGT Centre, Harbin Institute of Technology, 150001 Harbin, China {gpliu,drees}@glam.ac.uk

from multiple sensors located in geographically different places; See Fig. 1 for an illustrative structure. Note that this type of multi-packet transmission usually occurs only in the sensor-to-controller channel as we do not often use more than one controllers in a single control system.

2) The data packet size of the communication network is so small that the set of data required for transmission at one time have to be divided and packed into several data packets. This type of multi-packet transmission can occur in either the sensor-to-controller channel or the controller-to-actuator channel, corresponding to the split of the sensing data or the control data, respectively.

From a practical viewpoint, although the insufficiently large data packet possibly contributes to the occurrence of multi-packet transmission, in the case of the Internet being used in NCSs (which is more and more popular nowadays), the data packet size is usually large enough to avoid the occurrence of this scenario. For example, in the commonly used Ethernet IEEE 802.3 protocol, a frame contains 368 bits of useful data, while an 8-bit data can encode $2^8 = 256$ different control signals which is ample for most control applications [6]. Due to this reason, this paper will consider only multi-packet transmission resulting from the geographically dispersed sensors and, consequently, only the sensing data may be affected by multi-packet transmission. The typical structure is illustrated in Fig. 1.

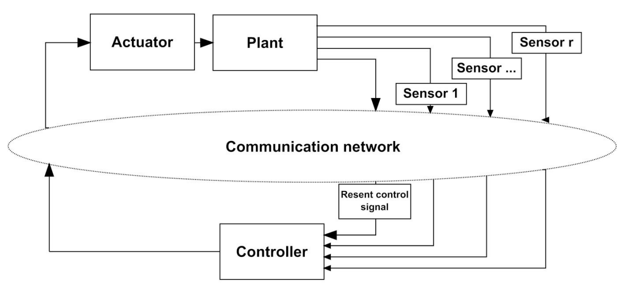

Fig. 1. Multi-packet transmission due to geographically dispersed sensors.

The existence of multi-packet transmission introduces multiple delays to the system. These delays can not be simply integrated into a single one since they are related to different parts of the concerned data. This fact poses great difficulties and conservativeness for the theoretical analysis as well as the practical implementation. Indeed, on the one hand, the multiple "partial" delays complicate the system model; on the other hand, using conventional control methods, the sensing data can be used only when all parts of it are received, which artificially increases the delay to the system and thus can significantly degrades the system performance.

B. Related work and preliminaries

As stated earlier, the main consequence of multi-packet transmission is the introduction of multiple delays to the control system. Given only this consequence, related works can be found in, for example, [7]–[10]. However, the models used there are essentially different from the situation considered here, as in all the aforementioned models, the data itself is integrated and multiple delays occur on the whole but not part of the data. For example, in [10] the following closed-loop model for continuous-time linear system with state feedback is proposed

$$\dot{x}(t) = Ax(t) + BKx(t - d_s(t) - d_a(t)) \quad (1)$$

where both the delays $d_s(t)$ and $d_a(t)$ are affecting the whole state $x(t)$. Although multiple delays are considered in (1), this model is obviously not suitable for NCSs with multi-packet transmission where, for different parts of the data, different delays apply. In fact, suppose there are r sensors in Fig. 1 and all the states are measurable, the system state at the controller side at time t, $x_{d(t)}(t)$, should read

$$x_{d(t)}(t) \triangleq [y_1^T(t - d_1(t)) \ \ldots \ y_r^T(t - d_r(t))]^T \quad (2)$$

where $y_i(t - d_i(t)), i = 1, 2, \ldots, r$ contains the state information obtained from the ith sensor and the delays $d_i(t), i = 1, \ldots, r$ are independent with each other. This implies that the closed-loop system with state feedback in the presence of multi-packet transmission can be written as follows,

$$\dot{x}(t) = Ax(t) + BKx_{d(t)}(t) \quad (3)$$

which is essentially different from the model in (1).

The model in (3) coincides with the models used in [3]–[5] where multi-packet transmission is explicitly considered. In [5], through some mathematical transformations the closed-loop system in (3) can be expressed in the form of multiple delays to the whole state with artificially added terms to the system matrices. Although this modeling approach simplifies the problem, the underlying philosophy is however to despondently accept the existence of multi-packet transmission and try only to design the control strategy from the control engineering perspective. The same philosophy can be found in [3], [4]. It is observed that this separation of control and communication results in considerable conservativeness, which thus motivates this work on the appropriate combination of control and communication to derive a superior co-design strategy. The proposed model-based compensation scheme for multi-packet transmission in NCSs consists of two parts, i.e., the reconstruction of the sensing data and the compensation for the negative effects brought by multi-packet transmission within the packet-based control framework. They are discussed in the next two sections, respectively.

In view of the fact that NCSs are practically implemented in a digital environment due to the use of computers and communication networks, it might be more suitable to consider the system model in discrete-time,

$$x(k+1) = Ax(k) + Bu(k) \quad (4a)$$

where $x \in \mathbb{R}^n$, $u \in \mathbb{R}^m$, $A \in \mathbb{R}^{n\times n}$, $B \in \mathbb{R}^{n\times m}$, and the calculation of the control signal $u(k)$ is based on the available sensing data for the controller at time k, denoted by

$$x_{\tau_{sc,k}}(k) \triangleq [y_1^T(k - \tau_{sc,k}^1) \ \ldots \ y_r^T(k - \tau_{sc,k}^r)]^T \quad (4b)$$

where similar to (2) $y_i(k-\tau^i_{sc,k}), i=1,2,\ldots,r$ denotes the state information obtained by the controller at time k from the ith sensor, $\tau^i_{sc,k}$ is the delay of $y_i(k-\tau^i_{sc,k})$ and $\tau_{sc,k} \triangleq [\tau^1_{sc,k} \ \ldots \ \tau^r_{sc,k}]^T$. Denote the dimension of $y_i(k-\tau^i_{sc,k})$ by n_i it is then held that $\sum_{i=1}^r n_i = n$.

Remark 1: Note that by appropriately reordering the system states $x(t)$ and adjusting the corresponding system matrices A, the following relationship is always held

$$\begin{aligned} x(k) &= [y_1^T(k) \ \ldots \ y_i^T(k) \ \ldots \ y_r^T(k)]^T \\ &= [x_1(k) \ \ldots \ x_j(k) \ \ldots \ x_n(k)]^T \end{aligned} \tag{5}$$

i.e., the components of the system state have the same order as the sensors, implying $y_1(k) = [x_1(k) \ \ldots \ x_{n_1}(k)]^T$, $y_2(k) = [x_{n_1+1}(k) \ \ldots \ x_{n_1+n_2}(k)]^T, \ldots$. Therefore, the system model in (4b) is proposed without the loss of generality. In what follows we will assume that the condition in (5) holds for the system model in (4) .

III. Sensing Data Reconstruction

In our proposed compensation scheme for multi-packet transmission, the sensing data is first reconstructed at the controller side. In what follows the reason why this process is necessary and the detailed reconstruction method are discussed.

A. Sensing data reconstruction: Why?

Suppose at time k at the controller side, we have the delayed sensing data $x_{\tau_{sc,k}}(k)$, the detailed structure of which is shown in (4b). $x_{\tau_{sc,k}}(k)$ can not be used directly to derive the control signal, as this is not the typical sensing data required for conventional control algorithms. In fact, in most existing control algorithms without the optimization on the multi-packet transmission issue, the control signal can not be calculated unless the sensing data from all the sensors are successfully received. Under our current notations, this is equivalently saying that the control signal at time k is calculated based on the artificially delayed state $x(k-\tau^{max}_{sc,k})$ where

$$\tau^{max}_{sc,k} \triangleq \max\{\tau^i_{sc,k} : i=1,\ldots,r\} \tag{6}$$

Comparing $x(k-\tau^{max}_{sc,k})$ with $x_{\tau_{sc,k}}(k)$ it is readily seen that in conventional control algorithms most of the available updated sensing data, $y_i(k-\tau^i_{sc,k})$ where $\tau^i_{sc,k} < \tau^{max}_{sc,k}$, are simply wasted. In addition, if any part of the data is lost during transmission, all the other data packets sent at that time will have to be discarded since in this case no "whole" sensing data can be constructed at the controller side. Such a straightforward observation tells the conservativeness of conventional control algorithms.

In this work we try to take advantage of all the available sensing data $x_{\tau_{sc,k}}(k)$, rather than the artificially delayed one, $x(k-\tau^{max}_{sc,k})$. In fact, our goal is to derive from the available sensing data the predicted, most updated sensing data, $\bar{x}(k-\tau^{min}_{sc,k})$ with

$$\tau^{min}_{sc,k} \triangleq \min\{\tau^i_{sc,k} : i=1,\ldots,r\} \tag{7}$$

where the symbol $^-$ in $\bar{x}(k-\tau^{min}_{sc,k})$ is used to indicate the fact that $\bar{x}(k-\tau^{min}_{sc,k})$ contains the predicted data but not fully the measured one. For simplicity of notations, let $\tau^*_{sc,k} \triangleq \tau^{min}_{sc,k}$ and thus $\bar{x}(k-\tau^*_{sc,k}) \triangleq \bar{x}(k-\tau^{min}_{sc,k})$.

Fig. 1 shows that the control signal applied to the plant is also sent to the controller. This seemingly redundant mechanism has its functions. In fact, the time-varying delay in the controller-to-actuator channel is unknown to the controller in NCSs and therefore the control signal actually applied to the plant is also unknown to the controller in the packet-based compensation scheme discussed in the next section. The controller can only know these actually applied control signals by sending them back. This transmission will be through another link, different from those for the sensing data. In this paper we assume that the delays for these resent control signals are less than those of any of the sensing data at the same time, i.e.,

$$\tau^u_{sc,k} \le \tau^*_{sc,k}, \forall k \tag{8}$$

where $\tau^u_{sc,k}$ is the delay of the resent control signal at time k. This assumption makes sense in practice because these control signals are sent back earlier than the corresponding sensing data, as the former can be sent back as soon as they are applied to the plant while the latter have to be after the control signals being applied and then sampled.

B. Sensing data reconstruction: How?

Partition the system matrices A and B in (4a) as $r \times r$ and $r \times 1$ block matrices respectively, as follows:

$$A = \begin{pmatrix} A^{11} & A^{12} & \ldots & A^{1r} \\ A^{21} & A^{22} & \ldots & A^{2r} \\ \vdots & \vdots & \ddots & \vdots \\ A^{r1} & A^{r2} & \ldots & A^{rr} \end{pmatrix}, \ B = \begin{pmatrix} B^1 \\ B^2 \\ \ldots \\ B^r \end{pmatrix} \tag{9}$$

where the block in row i and column j of A in its block matrix form, A^{ij}, has the size of $n_i \times n_j$ and the ith block of B in its block matrix form, B^i, has the size of $n_i \times m$. This partition enables us to write the system in (4) in block matrix form with regards to the system states obtained from different sensors (by using also (5)), the ith of which is

$$y_i(k+1) = \sum_{j=1}^r A^{ij} y_j(k) + B^i u(k) \tag{10}$$

Notice that at time k at the controller side, the whole state information at time $k-\tau^{max}_{sc,k}$ is available to the controller and thus the state information of sensor i one step ahead (if not available yet to the controller) can be estimated using (10),

$$\hat{y}_i(k-\tau^{max}_{sc,k}+1) = \sum_{j=1}^r A^{ij} y_j(k-\tau^{max}_{sc,k}) + B^i u(k-\tau^{max}_{sc,k}) \tag{11}$$

Ideally this process can be repeated step by step and the whole state information can then be estimated till $k-\tau^*_{sc,k}$, thus fulfilling our objective. However, the effect of multi-packet transmission makes that some parts of the data may not be available during this process. In fact, at this very time instant, $k - \tau_{sc,k}^{max} + 1$, the sensing data from at least one sensor will not available to the controller, due to the definition of $\tau_{sc,k}^{max}$ in (6).

The state information of sensor i at time $k - \tau_{sc,k}^{max} + l$, i.e., $y_i(k - \tau_{sc,k}^{max} + l), 1 \leq l \leq \tau_{sc,k}^{max} - \tau_{sc,k}^{*} - 1$ can be missing due to the following two different reasons and they are treated in our algorithm using a very similar strategy.

1) The delay of sensor i at time k is so large that $\tau_{sc,k}^{i} > \tau_{sc,k}^{max} - l$. This will prevent any newer sensing data from sensor i being received by the controller on or after $k - \tau_{sc,k}^{max} + l$. Notice that if the sensing data reconstruction is done step by step from the very beginning (11), the reconstructed one, $\hat{y}^i(k - \tau_{sc,k}^{max} + l)$ will always be available at the time of calculating the one step ahead sensing data. Therefore in this case we will use the reconstructed sensing data for sensor i on and after time $k - \tau_{sc,k}^{max} + l$.
2) The delay of sensor i at time k is small, ensuring that $\tau_{sc,k}^{i} \leq \tau_{sc,k}^{max} - l$, but a data packet dropout occurs. In this case, the sensing data from sensor i can be still available after $k - \tau_{sc,k}^{max} + l$ but not on this specific time instant. Therefore, the estimated one $\hat{y}^i(k - \tau_{sc,k}^{max} + l)$ is used at $k - \tau_{sc,k}^{max} + l$, but we will turn back to the real measured sensing data whenever available.

Simply speaking, the underlying idea of our estimation process is, the real measured sensing data is used as much as possible and the estimated one is used only in the absence of the former. The general estimation equation can be written as follows, for $1 \leq l \leq \tau_{sc,k}^{max} - \tau_{sc,k}^{*} - 1$,

$$\hat{y}_i(k - \tau_{sc,k}^{max} + l + 1) = \sum_{j=1}^{r} A^{ij} \bar{y}_j(k - \tau_{sc,k}^{max} + l) + B^i u(k - \tau_{sc,k}^{max} + l) \quad (12)$$

where $\bar{y}_j(k - \tau_{sc,k}^{max} + l) = y_j(k - \tau_{sc,k}^{max} + l)$ if $y_j(k - \tau_{sc,k}^{max} + l)$ is available to the controller; $\bar{y}_j(k - \tau_{sc,k}^{max} + l) = \hat{y}_j(k - \tau_{sc,k}^{max} + l)$ otherwise. Notice that $u(k - \tau_{sc,k}^{max} + l), 1 \leq l \leq \tau_{sc,k}^{max} - \tau_{sc,k}^{*} - 1$ is always available due to the assumption in (8),

Repeat this process for all parts of the sensing data, we can have an estimation of the whole sensing data at time $k - \tau_{sc,k}^{*}$, as follows,

$$\bar{x}(k - \tau_{sc,k}^{*}) = [\bar{y}_1^T(k - \tau_{sc,k}^{*}) \ \ldots \ \bar{y}_r^T(k - \tau_{sc,k}^{*})]^T \quad (13)$$

where similarly, $\bar{y}_j(k - \tau_{sc,k}^{*}) = y_j(k - \tau_{sc,k}^{*})$ if $y_j(k - \tau_{sc,k}^{*})$ is available to the controller; $\bar{y}_j(k - \tau_{sc,k}^{*}) = \hat{y}_j(k - \tau_{sc,k}^{*})$ otherwise.

IV. Packet-based compensation

With the sensing data being reconstructed at the controller side, we are now able to fit the problem into the packet-based control framework for NCSs to actively and effectively compensate for the negative effects brought by the communication constraints in NCSs, including especially those by multi-packet transmission.

A. Packet-based compensation

Packet-based control for NCSs is already a mature control strategy [6], [11], [12]. In this section this control strategy is modified to account for multi-packet transmission in NCSs, while the detailed design process of general packet-based control framework is ignored.

The key point of the previously reported packet-based control approach is to realize that the number of control signals that one data packet can contain in NCSs is usually much larger than the upper bound of the delay (data packet dropout as well). Indeed, denote the effective load of the data packet being used in NCSs by B_p and the data size required for encoding a single step of the control signal by B_c. The number of control signals that one data packet can contain can then be obtained as

$$N = \lfloor \frac{B_p}{B_c} \rfloor \quad (14)$$

where $\lfloor \frac{B_p}{B_c} \rfloor = \max\{\varsigma : \varsigma \in \mathbb{N}, \varsigma \leq \frac{B_p}{B_c}\}$. The following condition typically holds for most NCSs.

Assumption 1 (Delay bound): The sum of the network-induced delay and consecutive data packet dropout in the controller-to-actuator channel is upper bounded by N, i.e.,

$$\bar{\tau}_{ca} \triangleq \max_{k \geq 1}\{\tau_{ca,k} + \bar{\chi}_{ca}\} < N \quad (15)$$

where $\tau_{ca,k}$ and $\bar{\chi}_{ca}$ represent the network-induced delay and the upper bound of consecutive data packet dropout in the controller-to-actuator channel, respectively.

In the current system setting, the successful implementation of the packet-based control approach also requires the following assumption.

Assumption 2 (Time synchronization): The control components in NCSs including the sensor, the controller and the actuator are time synchronized and data packets are sent with time stamps to notify when they were sent.

With Assumptions 1 and 2, we are able to send a sequence of forward control signals (or "forward control sequence" (FCS)) simultaneously over the network instead of one at a time as typically done in conventional control systems, with the length of the FCS being $\bar{\tau}_{ca} + 1$. That is, at time k at the controller side, instead of calculating and sending only current control signal $u(k)$, the FCS $U(k|k - \tau_{sc,k}^{*})$ calculated based on the reconstructed sensing data $\bar{x}(k - \tau_{sc,k}^{*})$, is packed into one data packet and sent to the actuator,

$$U(k|k - \tau_{sc,k}^{*}) \triangleq [u(k|k - \tau_{sc,k}^{*}) \ \ldots \ u(k + \bar{\tau}_{ca}|k - \tau_{sc,k}^{*})] \quad (16)$$

Upon receiving $U(k|k - \tau_{sc,k}^{*})$, the actuator is then able to select from it the appropriate control signal to actively compensate for current communication constraints in NCSs (via a specially designed control component called "control action selector" (CAS)). For example, if the delay in the control-to-actuator channel for $U(k|k - \tau_{sc,k}^{*})$ is $\tau_{ca,k}$, the actuator may thus choose $u(k + \tau_{ca,k}|k - \tau_{sc,k}^{*})$ at time $k + \tau_{ca,k}$ and apply it to the plant to compensate for the communication constraints. Notice here that all the time instants are based on

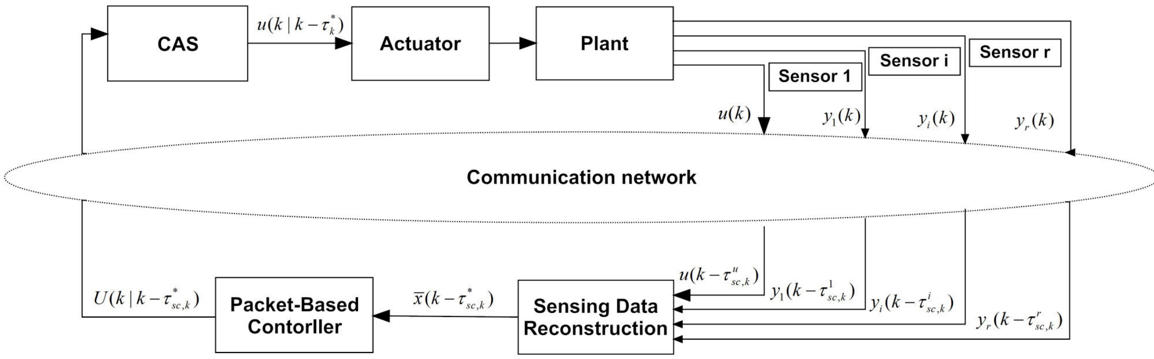

Fig. 2. Model-based compensation for NCSs with multi-packet transmission.

the controller side. If we use the time at the plant side and denote the delay for the chosen control signal by $\tau_{ca,k}^*$, the control law can then be written as follows,

$$u(k) = u(k|k - \tau_k^*) \tag{17}$$

where $\tau_k^* \triangleq \tau_{sc,k}^* + \tau_{ca,k}^*$ and $u(k|k-\tau_k^*)$ is selected from FCS $U(k-\tau_{ca,k}^*|k-\tau_k^*)$.

The general block diagram of the packet-based compensation scheme for NCSs with multi-packet transmission is illustrated in Fig. 2. It is clear that the control law in (17) provides an active compensation mechanism for the communication constraints in NCSs including the negative effects brought by multi-packet transmission and thus potentially leads to a better system performance than that using conventional control approaches. The reader of interest is referred to [6], [11], [12] for further details of the general framework of packet-based control for NCSs.

B. Packet-based controller: A model-based solution

The packet-based compensation scheme and the corresponding control law in (17) give the general control strategy for NCSs with multi-packet transmission. Under this framework, the specific controller can be designed separately, enabling this control strategy to be a unified one. As an example a model predictive control (MPC) based controller is designed in this paper to validate the effectiveness of the proposed approach, but any other control algorithms that can give better system performance can also be applied.

In a typical MPC implementation, the predictive controller determines a sequence of forward control signals at each control interval that optimize future open-loop plant behavior and only the first control input is applied to the plant. This sequence of forward control signals can be regarded as the FCS in the packet-based control framework without any modification, thus making the MPC based method a natural selection for our framework. However, different from classic MPC method where only the first control signal is applied to the plant, in our current framework any forward control signal from the FCS can be selected and then applied to the plant, in order to actively compensate for the communication constrains in NCSs.

Different from classic MPC method, within our framework the objective function, $J_{k,\tau_{sc,k}^*}$ at time k, is a function of the predicted states and control signals based on the reconstructed sensing data $\bar{x}(k-\tau_{sc,k}^*)$, as follows,

$$\begin{aligned} J_{k,\tau_{sc,k}^*} =& \bar{X}^T(k|k-\tau_{sc,k}^*)Q\bar{X}(k|k-\tau_{sc,k}^*) \\ &+ \bar{U}^T(k|k-\tau_{sc,k}^*)R\bar{U}(k|k-\tau_{sc,k}^*) \end{aligned} \tag{18}$$

where $\bar{X}(k|k-\tau_{sc,k}^*) = [x(k+1|k-\tau_{sc,k}^*) \cdots x(k+N_p|k-\tau_{sc,k}^*)]^T$ is the predictive state trajectory, $\bar{U}(k|k-\tau_{sc,k}^*) = [u(k-\tau_{sc,k}^*|k-\tau_{sc,k}^*) \cdots u(k+N_u-1|k-\tau_{sc,k}^*)]^T$ is the predicted control signals, Q and R are constant weighting matrixes, N_p and N_u are the prediction horizon and the control horizon respectively, and for the successful implementation of the proposed approach we require $N_u > \bar{\tau}_{ca}$ and normally we have $N_p \geq N_u$.

Due to the page limit the optimal FCS is simply presented as follows without the detailed deduction process. The reader of interest is referred to [6] where a similar process was conducted.

$$U(k|k-\tau_{sc,k}^*) = K_{\tau_{sc,k}^*}\bar{x}(k-\tau_{sc,k}^*) \tag{19}$$

where $K_{\tau_{sc,k}^*} = -M_{\tau_{sc,k}^*}(F_{\tau_{sc,k}^*}^T Q F_{\tau_{sc,k}^*} + R)^{-1} F_{\tau_{sc,k}^*}^T Q E_{\tau_{sc,k}^*}$ with $E_{\tau_{sc,k}^*} = [(A^{\tau_{sc,k}^*+1})^T \cdots (A^{\tau_{sc,k}^*+N_p})^T]^T$, $F_{\tau_{sc,k}^*}$ is a block lower triangular matrix with its non-null elements defined by $(F_{\tau_{sc,k}^*})_{ij} = A^{\tau_{sc,k}^*+i-j}B$, $j-i \leq \tau_{sc,k}^*$ and $M_{\tau_{sc,k}^*} = [0_{m(\bar{\tau}_{ca}+1)\times m\tau_{sc,k}^*} \; I_{m(\bar{\tau}_{ca}+1)\times m(\bar{\tau}_{ca}+1)} \; 0_{m(\bar{\tau}_{ca}+1)\times m(N_u-\bar{\tau}_{ca})}]$.

C. The model-based compensation scheme

Up to now we are able to organize the whole algorithm as follows, the block diagram of which is illustrated in Fig. 2.

Algorithm 1 (Model-based compensation scheme):

1. The multiple sensors sample the plant dynamics and send them to the controller independently; The control signal applied to the plant is also sent to the controller;

2. The sensing data is reconstructed at the controller side using (12) and (13), based on the sensing data from different sensors with different delays (4b) and the resent control signal $u(k-\tau_{sc,k}^u)$;

3. The FCS is calculated using (19) and sent to the actuator within one data packet;

4. The CAS selects the appropriate control signal to compensate for the communication constraints by (17) and and the actuator applies it to the plant.

V. Numerical Example

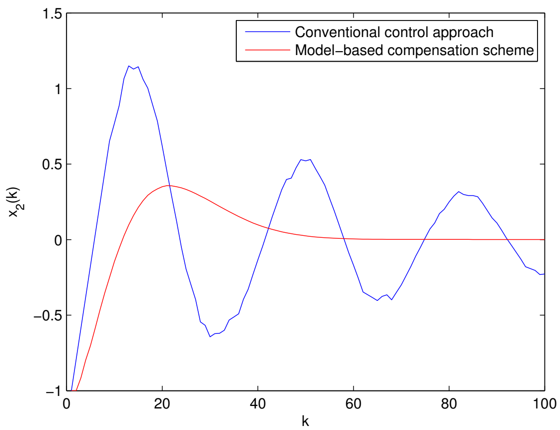

Fig. 3. State trajectories using the LQR controller and the model-based compensation scheme, respectively. Showing only the second state.

Consider a second order system in discrete-time borrowed from [6], which is open-loop unstable with the system matrices being

$$A = \begin{pmatrix} 0.98 & 0.1 \\ 0 & 1 \end{pmatrix}, B = \begin{pmatrix} 0.04 \\ 0.1 \end{pmatrix}.$$

In [6], a Linear Quadratic Optimal (LQR) controller is designed without consideration of the communication constraints, which yields the time-invariant feedback gain $K_{LQR} = [0.7044\ 1.3611]$. This controller will be used here to illustrate the effectiveness of the proposed model-based compensation scheme.

We assume that the two states are sampled and transmitted by two independent sensors, meaning that at any time instant k, the whole state $x(k)$ has to be constructed from data received from two sensors, i.e., $x(k) = [y_1(k)\ y_2(k)]$, where $y_1(k) = x_1(k)$, $y_2(k) = x_2(k)$ are from different sensors. The sensing data from different sensors experience different delays, with the upper bound being 4 and 8 time steps respectively. In the model-based compensation scheme, the sensing data is reconstructed at the controller first, and then the packet-based controller is designed using (19) with $N_u = 15, N_p = 20$ and Q and R being identity matrices with appropriate dimensions. The initial state of the system is set as $x_0 = [-1\ -1]^T$ and the upper bound of delay and consecutive dropouts in the controller-to-actuator channel is $\bar{\tau}_{ca} = 3$.

The comparison of the state trajectories using the model-based compensation scheme and the LQR controller is shown in Fig. 3. It is seen that the model-based compensation scheme gives rise to better system performance than the LQR controller. This is mainly due to the different upper bounds of the delays in the sensor-to-controller channel for the two methods: The LQR controller uses an upper bound of 8 time steps (the maximum of all partial delays) since the control signal is not calculated until sensing data from both sensors are received, while the model-based compensation scheme successfully reduces the upper bound to 4 time steps (the minimum of all partial delays). This comparison illustrates the effectiveness of the proposed model-based compensation scheme.

VI. Conclusions

Multi-packet transmission is a distinct feature of NCSs and potentially degrades the system performance significantly. By reconstructing the sensing data and then compensating for the communication constraints using the packet-based control approach, the negative effects brought by multi-packet transmission are effectively eliminated. Future work will concentrate on the theoretical evaluation of this control strategy and its extension to continuous-time and nonlinear systems.

References

[1] L.-S. Hu, T. Bai, P. Shi, and Z. Wu, "Sampled-data control of networked linear control systems," *Automatica*, vol. 43, no. 5, pp. 903–911, 2007.

[2] S. Graham, G. Baliga, and P. R. Kumar, "Abstractions, architecture, mechanisms, and a middleware for networked control," *IEEE Trans. Autom. Control*, vol. 54, no. 7, pp. 1490–1503, 2009.

[3] M. S. Branicky, V. Liberatore, and S. M. Phillips, "Networked control system co-simulation for co-design," in *Proc. 2003 American Control Conference*, vol. 4, 2003, pp. 3341– 3346.

[4] D.-L. Wen and G.-H. Yang, "State feedback control of continuous-time networked control systems in multiple-packet transmission," in *Proc. 2009 Chinese Control and Decision Conference (CCDC 2009)*, 2009, pp. 582–587.

[5] X.-L. Zhu and G.-H. Yang, "State feedback controller design of networked control systems with multiple-packet transmission," *Int. J. Control*, vol. 82, no. 1, pp. 86–94, 2009.

[6] Y.-B. Zhao, G.-P. Liu, and D. Rees, "Design of a packet-based control framework for networked control systems," *IEEE Trans. Control Syst. Technol.*, vol. 17, no. 4, pp. 859–865, 2009.

[7] F.-L. Lian, J. Moyne, and D. Tilbury, "Modelling and optimal controller design of networked control systems with multiple delays," *Int. J. Control*, vol. 76, no. 6, pp. 591 – 606, 2003.

[8] T. Ensari and S. Arik, "Global stability analysis of neural networks with multiple time varying delays," *IEEE Trans. Autom. Control*, vol. 50, no. 11, pp. 1781–1785, 2005.

[9] M. Basin and J. Rodriguez-Gonzalez, "Optimal control for linear systems with multiple time delays in control input," *IEEE Trans. Autom. Control*, vol. 51, no. 1, pp. 91–97, 2006.

[10] J. Lam, H. Gao, and C. Wang, "Stability analysis for continuous systems with two additional time-varying delay components," *Syst. Control Lett.*, vol. 56, no. 1, pp. 16–24, 2007.

[11] Y.-B. Zhao, G.-P. Liu, and D. Rees, "Modeling and stabilization of continuous-time packet-based networked control systems," *IEEE Trans. Syst. Man Cybern. Part B-Cybern.*, vol. 39, no. 6, pp. 1646–1652, 2009.

[12] ——, "Packet-based deadband control for Internet-based networked control systems," *IEEE Trans. Control Syst. Technol.*, vol. 18, no. 5, pp. 1057–1067, 2010.