Dynamic Controller Allocation for Control over Erasure Channels

Daniel E. Quevedo * **Karl H. Johansson** ** **Anders Ahlén** *** **Isabel Jurado** ****

* *School of Electrical Engineering & Computer Science, The University of Newcastle, NSW 2308, Australia.* dquevedo@ieee.org
** *ACCESS Linnaeus Centre, School of Electrical Engineering, Royal Institute of Technology, Stockholm, Sweden.* kallej@ee.kth.se
*** *Department of Engineering Sciences, Signals and Systems, Uppsala University, Uppsala, Sweden.* Anders.Ahlen@signal.uu.se
**** *Departamento de Ingeniería de Sistemas y Automática, Escuela Superior de Ingenieros, Universidad de Sevilla, Spain.* ijurado@cartuja.us.es

Abstract: Wireless sensor-actuator networks offer flexibility in the design of networked control systems. One novel element when using networks with multiple nodes is that the role of individual nodes does not need to be fixed. In particular, there is no need to pre-allocate which nodes assume controller functions and which ones merely relay data. We present a flexible architecture for networked control using multiple nodes connected in series over analog erasure channels. The control architecture adapts to changes in network conditions, by allowing the role played by individual nodes to depend upon transmission outcomes.

Keywords: Wireless sensor-actuator networks; Packet dropouts.

1. INTRODUCTION

In a Networked Control System (NCS), sensors, controller and actuator links are not transparent, but are affected by bit-rate limitations, packet dropouts and/or delays, leading to performance degradation; see, e.g., papers in the special issues (Antsaklis and Baillieul, 2004, 2007; Franceschetti et al., 2008; Chen et al., 2011). In principle, communication artifacts can be alleviated by transmitting at high bit-rates and with high transmission powers (Pantazis and Vergados, 2007; Park et al., 2008; Quevedo et al., 2010; Cardoso de Castro et al., 2012). However, if network nodes are wireless and battery-operated, then energy efficiency is paramount. This makes the design of NCSs often a challenging task.

An interesting aspect which we believe has not been explored sufficiently is that of architectural freedom in the design of NCSs. When compared to traditional hard-wired control loops, NCSs offer architectural flexibility and additional degrees of freedom. In particular, there is no need to pre-assign in a static fashion which nodes carry out control calculations, and which nodes merely relay data. Intuitively, and in relation to the packet dropout issue, the roles of individual nodes should depend upon the information available, thus, upon transmission outcomes and the availability of acknowledgments.

⋆ The research was supported in part under Australian Research Council's Discovery Projects funding scheme (project number DP0988601), by the VINNOVA project WiComPI, project Dnr2009-02963, by MCyT (Grant DPI2010-19154), and by the European Commission FeedNetBack Project (Grant agreement 223866).

As background to our current work, Goodwin et al. (2008) studied performance of three static NCS architectures By adopting an additive signal-to-noise ratio constrained channel model. The results in that paper suggest that, in the absence of coding, placing the controller at the actuator node will give better performance than placing it at the sensor node. It is worth noting that Silva and Pulgar (2011) showed that the channel model in (Goodwin et al., 2008) can be used to describe erasure channels where dropouts are independent and identically distributed (i.i.d.). Viewed from that perspective, it was implicitly assumed in Goodwin et al. (2008) that communication acknowledgments are not available at the transmitter. Robinson and Kumar (2008) studied NCSs with stochastic packet dropouts using optimal control techniques. *Inter-alia*, the work showed that optimal control performance can be achieved if all nodes aggregate their entire history of received data and relay it to the controller which is located at the actuator node. Depending upon the information available at each node, various optimal control problems can be examined, see (Robinson and Kumar, 2008) and also (Gupta et al., 2009) for a formulation where the controller is pre-allocated to a fixed node having perfect access to previous plant inputs. More recently, Pajic et al. (2011) studied a distributed control strategy wherein the network itself acts as a controller for a MIMO plant. Each node (including the actuator nodes) perform linear combinations of internal state variables of neighboring nodes. In the case of analog erasure channels with i.i.d. dropouts (without acknowledgments), in (Pajic et al., 2011) the resulting NCS is then cast, analyzed and designed as a jump linear system.

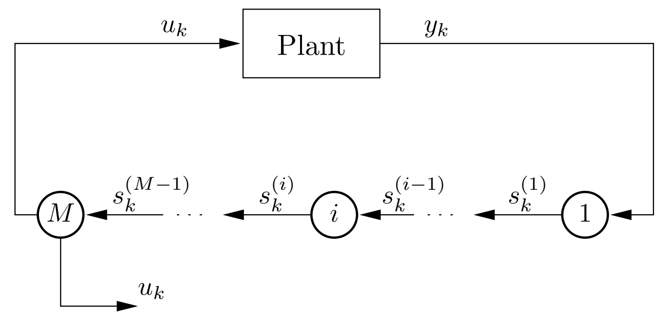

Fig. 1. Control over a line-graph with dropouts and unreliable acknowledgments of actuator values.

The present work studies a single-loop NCS topology which uses a series connection of analog erasure channels. Thus, transmissions are affected by random packet dropouts. We focus on situations where the wireless nodes have only limited energy. This precludes communicating long data packets. Further, with exception of the actuator node, the nodes do not provide transmission acknowledgments. Instead of tackling optimal control formulations, we assume that the control policy consists of a pre-designed state feedback-gain, which, in the absence of network effects, would lead to the desired performance. Within this context, we present a flexible NCS architecture where the role played by individual nodes depends upon transmission outcomes. While all nodes perform (non-linear) Kalman filtering, with the algorithm proposed, transmission outcomes and their acknowledgments will determine, at each instant, whether the control input will be calculated at the actuator node, or closer to the sensor node. It turns out that, if individual dropout processes are i.i.d., then the controller location has a stationary distribution, which can be easily characterized.

The remainder of this work is organized as follows. Section 2 describes the NCS topology of interest. In Section 3 we present the dynamic NCS architecture. For the case of independent dropouts, Section 4 provides the distribution of the controller location. A numerical example is given in Section 5. Section 6 draws conclusions.

Notation: We write $\mathbb{N}_0$ for $\{0, 1, 2, \ldots\}$; $\mathbb{R}$ are the real numbers, whereas $\mathbb{R}_{\geq 0} \triangleq [0, \infty)$. If a matrix Q is positive definite, then we write $Q \succ 0$. We adopt the convention $\sum_{j=1}^{0} a_j = 0$, for all $a_0, a_1 \in \mathbb{R}$. To denote the probability of an event Ω, we write $\mathbf{Pr}\{\Omega\}$. A real random variable μ, which is zero-mean Gaussian with covariance Γ is denoted by $\mu \sim \mathcal{N}(0, \Gamma)$.

2. NCS SETUP

We consider MIMO LTI plant models of the form

$$\begin{aligned} x_{k+1} &= Ax_k + Bu_k + w_k \\ y_k &= Cx_k + v_k, \quad k \in \mathbb{N}_0 \end{aligned} \tag{1}$$

where $x_0 \sim \mathcal{N}(0, P_0)$, $P_0 \succ 0$. In (1), $u_k \in \mathbb{R}^m$ is the plant input, $x_k \in \mathbb{R}^n$ is the state, $y_k \in \mathbb{R}^p$ is the output, and $w_k \sim \mathcal{N}(0, Q)$, $Q \succ 0$ and $v_k \sim \mathcal{N}(0, R)$, $R \succ 0$, are driving noise and measurement noise, respectively. [1]

[1] In addition to measurement noise, v_k may also describe quantization effects, here modeled as Gaussian; see, e.g., (Quevedo et al., 2010).

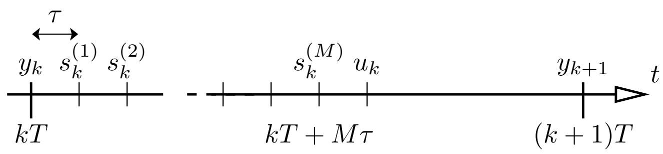

Fig. 2. Transmission Schedule; $t \in \mathbb{R}_{\geq 0}$ is actual time.

As foreshadowed in the introduction, we assume that a suitable feedback gain L has been pre-designed, such that if the the control inputs

$$u_k = Lx_k, \quad k \in \mathbb{N}_0 \tag{2}$$

were implemented at the plant, then satisfactory performance would be attained.

Sensor and actuator nodes are connected via a wireless network, characterised via a (directed) line-graph having M nodes, see Fig. 1. Transmissions are in sequential Round-Robin fashion $\{1, 2, \ldots, M, 1, 2, \ldots\}$ as depicted in Fig. 2. More precisely, the packet $s_k^{(i)}$ is transmitted from node i to node $i+1$ at times $kT + i\tau$, where T is the sampling period of (1) and $\tau \ll T/(M+1)$ refers to the times between transmissions of packets $s_k^{(i)}$. The plant input u_k is applied at time $kT + (M+1)\tau$. We, thus assume that in-network processing is much faster than the plant dynamics (1) and, as in, e.g., (Robinson and Kumar, 2008), neglect delays introduced by the network.

A distinguishing characteristic of the situation at hand is that (due to channel fading) the network introduces stochastic packet dropouts. To study the situation, we adopt an analog erasure channel model and introduce the binary success random processes

$$\gamma_k^{(i)} \in \{0, 1\}, \quad k \in \mathbb{N}_0,\ i \in \{1, 2, \ldots, M-1\},$$

where $\gamma_k^{(i)} = 1$ indicates that transmission of the packet $s_k^{(i)}$ from node i to node $i+1$ at time $kT + i\tau$, is successful, i.e., error-free; $\gamma_k^{(i)} = 0$ refers to a packet-dropout. Throughout this work we assume that the sensor node $i = 1$ has direct access to plant output measurements. For notational convenience, we write $\gamma_k^{(0)} = 1$, for all $k \in \mathbb{N}_0$.

To save energy, in our formulation the wireless nodes $i \in \{1, 2, \ldots, M-1\}$ do not provide acknowledgments of receipt of the packets. However, the actuator node, M, will in general have less stringent energy constraints. At time $kT + (M+1)\tau$, it broadcasts (in parallel) the control value applied, namely u_k, back to the wireless nodes $i \in \{1, \ldots, M-1\}$, see Fig. 1. This acknowledgment-like signal is unreliable and affected by dropouts with associated success processes

$$\delta_k^{(i)} \in \{0, 1\}, \quad k \in \mathbb{N}_0,\ i \in \{1, 2, \ldots, M-1\}.$$

More precisely, if u_k is successfully received at node i, then we set $\delta_k^{(i)} = 1$; see also (Garone et al., 2010; Imer et al., 2006) for studies on the importance of acknowledgments in closed loop control. We assume that the actuator node has perfect knowledge of plant inputs, and thus, write $\delta_k^{(M)} = 1$, $\forall k \in \mathbb{N}_0$.

Due to packet dropouts, plant output measurements are not always available at the actuator node. On the other hand, the sensor node will, in general, not have perfect information of previous plant inputs. This makes the implementation of (2) via estimated state feedback a challenging task. The main purpose of the present work is to investigate which nodes of the network should use their local state estimates to implement the control law (2). We foresee that our approach will lead to a dynamic assignment of the role played by the individual network nodes. Which tasks are carried out by each node at each time instant, will depend upon transmission outcomes.

3. A FLEXIBLE NCS ARCHITECTURE

To keep communications and thereby energy use low, the packets transmitted by each node i have only two fields, namely, output measurements and tentative plant inputs (if available):

$$s_k^{(i)} = \big(y_k, u_k^{(i)}\big). \tag{3}$$

Plant outputs are transmitted in order to pass on information on the plant state to the nodes $\{i+1, i+2, \dots, M\}$, see Fig. 1. On the other hand, $u_k^{(i)}$ in (3) is the plant input which is applied at the plant provided the packet $s_k^{(i)}$ is delivered at the actuator node. If $s_k^{(i)}$ is lost, then following the algorithm described in Section 3.2, the plant input will be provided by one of the later nodes $\ell > i$, which thereby takes on the controller role at time k. For further reference, we will refer to the node which calculates the plant input at time k as $c_k \in \{1, 2, \dots, M\}$.

To implement the control law (2) over the network using packets of the form (3), we will adopt a certainty equivalence formulation wherein tentative plant inputs satisfy:

$$u_k^{(i)} = L\hat{x}_k^{(j)}, \quad \text{for some } j \in \{1, 2, \dots, i\},$$

where $\hat{x}_k^{(j)}$ is the plant state estimate computed at node j, by using local information. Thus, the plant input is given by

$$u_k = L\hat{x}_k^{(c_k)}. \tag{4}$$

Intuitively, good control performance will be achieved if the estimate used in (4) is accurate. Clearly, due to the multi-hop nature of the network, nodes which are closer to the sensor will have access to more output measurements, see Fig. 1. On the other hand, one can expect that nodes which are physically located closer to the actuator node will on average receive more plant input acknowledgments, thus, have better knowledge of plant inputs.

Remark 1. Of course, the above simple transmission and control strategy will in general not be optimal. In particular, nodes do not transmit local state estimates and the control law does not depend upon network parameters, e.g., dropout probabilities. The purpose of the present work is to develop a simple and practical method, which uses an existing control law for implementation over an unreliable network and only requires little communication.

3.1 State Estimation

While only the node c_k, will provide the plant input at instant k, in the present formulation all nodes compute local state estimates, $\hat{x}_k^{(i)}$, by using the data received from the preceding node. This serves as safeguard for instances when the loop is broken due to dropouts. In the sequel, we will focus on situations where acknowledgments of plant inputs are "quite reliable". Thus, the state estimates are simply calculated as per

$$\hat{x}_k^{(i)} = A\hat{x}_{k-1}^{(i)} + B\hat{u}_{k-1}^{(i)} + K_k^{(i)}\big(y_k - C(A\hat{x}_{k-1}^{(i)} + B\hat{u}_{k-1}^{(i)})\big),$$

where $K_k^{(i)}$ is the Kalman filter gain for a system with intermittent observations

$$\begin{aligned} K_k^{(i)} &= \Gamma_k^{(i)} P_k^{(i)} C^T (CP_k^{(i)} C^T + R)^{-1} \\ P_{k+1}^{(i)} &= A(I - K_k^{(i)} C) P_k^{(i)} A^T + Q. \end{aligned} \tag{5}$$

and

$$\Gamma_k^{(i)} \triangleq \prod_{j \in \{0,1,\dots,i-1\}} \gamma_k^{(j)}$$

is equal to 1 if and only if y_k is available at node i at time $kT + (i-1)\tau$. In (5), $\hat{u}_{k-1}^{(i)}$ is a local plant input estimate. In particular, if $\delta_{k-1}^{(i)} = 1$, then $\hat{u}_{k-1}^{(i)} = u_{k-1}$, and (5) becomes a Kalman filter with intermittent observations; see, e.g., (Sinopoli et al., 2004; Huang and Dey, 2007; Censi, 2011). On the other hand, at instances where $\delta_{k-1}^{(i)} = 0$, node i uses $u_{k-1}^{(i)}$, the tentative plant input value transmitted in the second field of the previous packet $s_{k-1}^{(i)}$ (if non-empty), or otherwise sets $\hat{u}_{k-1}^{(i)} = L\hat{x}_{k-1}^{(i)}$, see Algorithm 1.

3.2 Algorithm for Dynamic Controller Placement

Algorithm 1 is run at every node i. Since we assume that acknowledgments from the actuator node are, in general, available, but transmissions of packets $s_k^{(i)}$ are less reliable, nodes closer to the sensor nodes can be expected to have better state estimates than nodes located further down the line. Therefore, preference is given to forward incoming tentative plant input values. [2]

The sensor node $i = 1$ uses as input

$$s_k^{(0)} = (y_k, \emptyset), \quad \gamma_k^{(0)} = 1. \tag{6}$$

If $\delta_{k-1}^{(1)} = 1$, then the sensor node calculates a tentative control value and transmits

$$s_k^{(1)} = (y_k, L\hat{x}_k^{(1)})$$

to node $i = 2$. Subsequent nodes then relay this packet to the actuator node. If the packet is dropped along the way, then the next node i where $\delta_{k-1}^{(i)} = 1$, calculates a tentative control value $u_k^{(i)} = L\hat{x}_k^{(i)}$ and transmits

$$s_k^{(i)} = (\emptyset, u_k^{(i)})$$

to node $i+1$, etc. On the other hand, if $\delta_{k-1}^{(1)} = 0$, then $s_k^{(0)}$ is relayed to subsequent nodes until arriving at some node i where u_{k-1} was successfully received. Control calculations are then carried out and the packet $s_k^{(i)}$ obtained is relayed towards the actuator node, etc. The actuator node, implements the value contained in the second field of $s_k^{(M)}$.

Remark 2. An advantage of allowing the control calculations to be located arbitrarily and in a time-varying fashion, is that it makes more difficult for someone to attack the NCS. The latter problem has been studied, for example, in Gupta et al. (2010); Smith (2011). □

[2] In contrast, under different assumptions, the previous work (Robinson and Kumar, 2008) concludes that control calculations should always be carried out at the actuator.

Algorithm 1 Dynamic Controller Placement

1: $k \leftarrow 0$, $\hat{x}_0^{(i)} \leftarrow 0$, $P_0^{(i)} \leftarrow P_0$, $j \leftarrow 0$
2: **while** $t \geq 0$ **do** ▷ $t \in \mathbb{R}_{\geq 0}$ is actual time
3:   **while** $t \leq kT + (i-1)\tau$ **do** ▷ wait-loop
4:     $j \leftarrow j + 1$
5:   **end while**
6:   $P_{k+1}^{(i)} \leftarrow AP_k^{(i)}A^T + Q$
7:   **if** $\gamma_k^{(i-1)} = 0$ **then** ▷ $s_k^{(i-1)}$ is dropped
8:     $u_k^{(i)} \leftarrow L\hat{x}_k^{(i)}$
9:     **if** $\delta_{k-1}^{(i)} = 1$ **then**
10:       $s_k^{(i)} \leftarrow \big(\emptyset, u_k^{(i)}\big)$ ▷ a tentative input
11:     **else**
12:       $s_k^{(i)} \leftarrow \big(\emptyset, \emptyset\big)$
13:     **end if**
14:   **end if**
15:   **if** $\gamma_k^{(i-1)} = 1$ **then** ▷ $s_k^{(i-1)}$ is received
16:     $(y, u) \leftarrow s_k^{(i-1)}$
17:     **if** $y \neq \emptyset$ **then** ▷ y_k is available
18:       $K_k^{(i)} \leftarrow P_k^{(i)}C^T(CP_k^{(i)}C^T + R)^{-1}$
19:       $\hat{x}_k^{(i)} \leftarrow \hat{x}_k^{(i)} + K_k^{(i)}\big(y - C\hat{x}_k^{(i)}\big)$
20:       $P_{k+1}^{(i)} \leftarrow A\big(I - K_k^{(i)}C\big)P_k^{(i)}A^T + Q$
21:     **end if**
22:     **if** $u \neq \emptyset$ **then**
23:       $u_k^{(i)} = u$
24:     **else**
25:       $u_k^{(i)} \leftarrow L\hat{x}_k^{(i)}$
26:     **end if**
27:     **if** $u = \emptyset \wedge \delta_{k-1}^{(i)} = 1$ **then**
28:       $s_k^{(i)} \leftarrow \big(y, u_k^{(i)}\big)$ ▷ a tentative input
29:     **else**
30:       $s_k^{(i)} \leftarrow (y, u)$ ▷ $s_k^{(i-1)}$ is forwarded
31:     **end if**
32:   **end if**
33:   **while** $t < kT + i\tau$ **do** ▷ wait-loop
34:     $j \leftarrow j + 1$
35:   **end while**
36:   **transmit** $s_k^{(i)}$
37:   **while** $t \leq kT + (M+1)\tau$ **do** ▷ wait-loop
38:     $j \leftarrow j + 1$
39:   **end while**
40:   **if** $\delta_k^{(i)} = 1$ **then** ▷ u_k is available
41:     $\hat{x}_{k+1}^{(i)} \leftarrow A\hat{x}_k^{(i)} + Bu_k$
42:   **else** ▷ the value in $s_k^{(i)}$ is used
43:     $\hat{x}_{k+1}^{(i)} \leftarrow A\hat{x}_k^{(i)} + Bu_k^{(i)}$
44:   **end if**
45:   $k \leftarrow k + 1$
46: **end while**

4. DYNAMIC CONTROLLER LOCATION

With Algorithm 1, which of the nodes calculates the plant input u_k, depends upon the transmission outcomes. For further reference, we shall denote the set of nodes which calculate a tentative control input (see lines 10 and 28 of Algorithm 1) via

$$\mathcal{C}_k \subset \{1, 2, \dots, M\}.$$

Since the packets $s_k^{(i)}$ are communicated sequentially, see Fig. 2, the controller node at time k is given by

$$c_k = \max(\mathcal{C}_k), \tag{7}$$

see (4). Clearly, $\mathcal{C}_k$ and c_k are determined by the realizations of the random variables

$$\big\{\delta_{k-1}^{(1)}, \delta_{k-1}^{(2)}, \dots, \delta_{k-1}^{(M-1)}, \gamma_k^{(1)}, \gamma_k^{(2)}, \dots, \gamma_k^{(M-1)}\big\}.$$

To further elucidate the situation, we introduce the processes $\big\{\mu_k^{(i)}\big\}_{k\in\mathbb{N}_0}$, $i \in \{0, 1, \dots, M\}$, where

$$\mu_k^{(i)} = \begin{cases} 0 & \text{if the second field of } s_k^{(i)} \text{ is empty} \\ 1 & \text{otherwise.} \end{cases} \tag{8}$$

The following assumption simplifies the analysis by supposing that transmission processes are i.i.d. It does, however, take into account the fact that radio connectivity from the actuator node to the other nodes will be distance dependent; see, e.g., (Goldsmith, 2005).

Assumption 3. The link transmission processes are i.i.d., with a common success probability $p \in [0, 1]$:

$$\mathbf{Pr}\big\{\gamma_k^{(i)} = 1\big\} = p, \quad \forall i \in \{1, 2, \dots, M-1\}. \tag{9}$$

The acknowledgment success processes are i.i.d., with

$$\mathbf{Pr}\big\{\delta_k^{(i)} = 1\big\} = q_i, \quad \forall i \in \{1, 2, \dots, M-1\}, \tag{10}$$

for given success probabilities $q_1, q_2, \dots, q_{M-1} \in [0, 1]$. □

Note that Assumption 3 does not imply that the processes $\big\{\mu_k^{(i)}\big\}_{k\in\mathbb{N}_0}$ for different nodes i are independent. However, the assumption made does imply stationarity, as evidenced by Lemma 4 given below.

Lemma 4. Suppose that Assumption 3 holds. Then

$$\mathbf{Pr}\big\{\mu_k^{(i)} = 1\big\} = q_i + \sum_{j=1}^{i-1} p^j q_{i-j} \prod_{\ell=0}^{j-1}(1 - q_{i-\ell}) \tag{11}$$

$$\mathbf{Pr}\{i \in \mathcal{C}_k\} = q_i\big(1 - p\mathbf{Pr}\big\{\mu_k^{(i-1)} = 1\big\}\big) \tag{12}$$

$$\mathbf{Pr}\{c_k = i\} = p^{M-i}\mathbf{Pr}\{i \in \mathcal{C}_k\}, \tag{13}$$

for all $k \in \mathbb{N}_0$ and $i \in \{1, 2, \dots, M\}$, and where $q_M = 1$.

Proof. Clearly, if Assumption 3 holds, then

$$\mathbf{Pr}\{\mu_k^{(1)} = 1\} = \mathbf{Pr}\{\delta_{k-1}^{(1)} = 1\} = q_1.$$

It is easy to see from lines 12 and 30 of Algorithm 1 that

$$\begin{aligned} i \in \mathcal{C}_k \iff & \delta_{k-1}^{(i)} = 1 \\ & \wedge \big(\gamma_k^{(i-1)} = 0 \vee \big(\gamma_k^{(i-1)} = 1 \wedge \mu_k^{(i-1)} = 0\big)\big) \\ \iff & \delta_{k-1}^{(i)} = 1 \wedge \big(\gamma_k^{(i-1)} = 0 \vee \mu_k^{(i-1)} = 0\big) \end{aligned} \tag{14}$$

and that, similarly

$$\begin{aligned}\mu_k^{(i)} = 0 &\iff \big(\gamma_k^{(i-1)} = 0 \wedge \delta_{k-1}^{(i)} = 0\big)\\ &\quad \vee \big(\gamma_k^{(i-1)} = 1 \wedge \mu_k^{(i-1)} = 0 \wedge \delta_{k-1}^{(i)} = 0\big)\\ &\iff \delta_{k-1}^{(i)} = 0 \\ &\quad \wedge \big(\gamma_k^{(i-1)} = 0 \vee \big(\gamma_k^{(i-1)} = 1 \wedge \mu_k^{(i-1)} = 0\big)\big)\\ &\iff \delta_{k-1}^{(i)} = 0 \wedge \big(\gamma_k^{(i-1)} = 0 \vee \mu_k^{(i-1)} = 0\big)\end{aligned} \tag{15}$$

for all $i \in \{1, 2, \dots, M\}$. Expression (15) provides the recursion

$$\begin{aligned}\mathbf{Pr}\{\mu_k^{(i)} = 1\} &= 1 - \mathbf{Pr}\{\delta_{k-1}^{(i)} = 0\}\\ &\quad \times \mathbf{Pr}\{\mu_k^{(i-1)} = 0 \vee \gamma_k^{(i-1)} = 0\}\\ &= 1 - (1-q_i)\big(1 - \mathbf{Pr}\{\mu_k^{(i-1)} = 1 \wedge \gamma_k^{(i-1)} = 1\}\big)\\ &= 1 - (1-q_i)\big(1 - p\mathbf{Pr}\{\mu_k^{(i-1)} = 1\}\big)\\ &= q_i + p(1-q_i)\mathbf{Pr}\{\mu_k^{(i-1)} = 1\},\end{aligned}$$

which has the explicit solution (11). On the other hand, (14) gives

$$\begin{aligned}&\mathbf{Pr}\{i \in \mathcal{C}_k\}\\ &\quad = \mathbf{Pr}\{\delta_{k-1}^{(i)} = 1\}\big(1 - \mathbf{Pr}\{\gamma_k^{(i-1)} = 1\}\mathbf{Pr}\{\mu_k^{(i-1)} = 1\}\big),\end{aligned}$$

thus establishing (12). By (7) the distribution of c_k can be determined from

$$\begin{aligned}\mathbf{Pr}\{c_k = i\} &= \mathbf{Pr}\{\max(\mathcal{C}_k) = i\}\\ &= \mathbf{Pr}\{i \in \mathcal{C}_k \wedge \gamma_k^i = \gamma_k^{i+1} = \dots = \gamma^{M-1} = 1\}\end{aligned}$$

for $i \in \{1, \dots, M-1\}$, whereas for the actuator node, we have $\mathbf{Pr}\{c_k = M\} = \mathbf{Pr}\{M \in \mathcal{C}_k\}$. This proves (13). $\square$

The above result characterizes the distributions of $\mu_k^{(i)}$, of $\mathcal{C}_k$, and of the controller node location c_k (provided Assumption 3 holds). These distributions depend upon the communication success probabilities.

Example 5. Suppose that Assumption 3 holds and that acknowledgments are always available, i.e.,

$$q_i = 1, \quad \forall i \in \{1, \dots, M\}.$$

Expression (11) then provides that

$$\mathbf{Pr}\{\mu_k^{(i)} = 1\} = 1, \quad \forall i \in \{1, \dots, M\}.$$

Since, by (6), $\mathbf{Pr}\{\mu_k^{(0)} = 1\} = 0$, Lemma 4 gives that

$$\mathbf{Pr}\{i \in \mathcal{C}_k\} = \begin{cases} 1 & \text{if } i = 1 \\ 1-p & \text{if } i \in \{2, \dots, M\}\end{cases}$$

and the controller location sequence has the following geometric-like distribution

$$\mathbf{Pr}\{c_k = i\} = \begin{cases} p^{M-1} & \text{if } i = 1 \\ (1-p)p^{M-i} & \text{if } i \in \{2, 3, \dots, M\}.\end{cases} \tag{16}$$

On the other hand, if the actuator does not broadcast the plant input values at all, in which case

$$q_i = 0 \quad \forall i \neq M,$$

then our result gives that

$$\begin{aligned}\mathbf{Pr}\{\mu_k^{(i)} = 1\} &= 0,\\ \mathbf{Pr}\{c_k = M\} = \mathbf{Pr}\{M \in \mathcal{C}_k\} &= 1,\end{aligned}$$

for all $i \in \{1, \dots, M-1\}$, $k \in \mathbb{N}_0$ and the controller is collocated with the actuator (with probability one). This essentially corresponding to the conclusions made by Robinson and Kumar (2008); Goodwin et al. (2008) for alternative NCS setups without acknowledgments of plant inputs. $\square$

Table 1. Set $\mathcal{C}_k$ (with controller node location c_k in bold-face) for $M = 3$, see Example 6

$\delta_{k-1}^{(1)}$	$\gamma_k^{(1)}$	$\delta_{k-1}^{(2)}$	$\gamma_k^{(2)}$	$\mathcal{C}_k$, c_k	**Pr**
1	1	any	1	$\{\mathbf{1}\}$	q_1p^2
1	1	any	0	$\{1, \mathbf{3}\}$	$q_1p(1-p)$
1	0	1	1	$\{1, \mathbf{2}\}$	$q_1(1-p)q_2p$
1	0	1	0	$\{1, 2, \mathbf{3}\}$	$q_1(1-p)q_2(1-p)$
1	0	0	any	$\{1, \mathbf{3}\}$	$q_1(1-p)(1-q_2)$
0	any	1	1	$\{\mathbf{2}\}$	$(1-q_1)q_2p$
0	any	1	0	$\{2, \mathbf{3}\}$	$(1-q_1)q_2(1-p)$
0	any	0	any	$\{\mathbf{3}\}$	$(1-q_1)(1-q_2)$

Example 6. Consider an NCS as in Fig. 1 with $M = 3$ nodes and suppose that Assumption 3 holds. In this case, Lemma 4 establishes that

$$\mathbf{Pr}\{c_k = i\} = \begin{cases} q_1p^2 & \text{if } i = 1 \\ pq_2(1-pq_1) & \text{if } i = 2 \\ 1 - pq_2 - p^2q_1 + p^2q_1q_2 & \text{if } i = 3.\end{cases} \tag{17}$$

Note that, since M is small, the result (17) can alternatively be obtained by examining the probabilities of all possible transmission outcomes, as illustrated in Table 1. Of course, for a large number of nodes, such a procedure is non-practical and use of Lemma 4 is preferable. $\square$

5. SIMULATION STUDY

We consider a network with $M = 10$ nodes. Dropout and acknowledgment processes are as per Assumption 3 with $p \in \{0.95, 0.97, 0.99\}$ and

$$q_1 = q_2 = q_3 = q_4 = 0.9, \; q_5 = q_6 = q_7 = q_8 = q_9 = 0.98.$$

Figures 3 to 5 illustrate histograms of c_k, obtained by running the algorithm for 1000 steps with dropout probabilities as indicated. Note the different scales on the y-axis. Fig. 3 shows that, with the algorithm proposed, for smaller link transmission success probabilities p, control calculations are in general carried out at the actuator node. On the other hand, if links are more reliable, then the controller will be placed at the sensor node at most time steps, see Fig. 5.

It is interesting to note that, in both cases considered, the histograms obtained approximate those which would have been obtained if one had chosen $q_i = 1$, for all $i \in \{1, 2, \dots, M\}$, see (16).

6. CONCLUSIONS

We have presented a flexible architecture for the implementation of an estimated state feedback control law over analog erasure channels. The algorithm provided allows the role played by individual nodes to depend on transmission outcomes. In particular, the controller location depends upon the availability of past plant input values and transmission outcomes. Future work may include performance and stability analysis of the resulting NCS in the presence of correlated dropout processes. Also an extension of the ideas presented to the control of multiple-loops would be of interest.

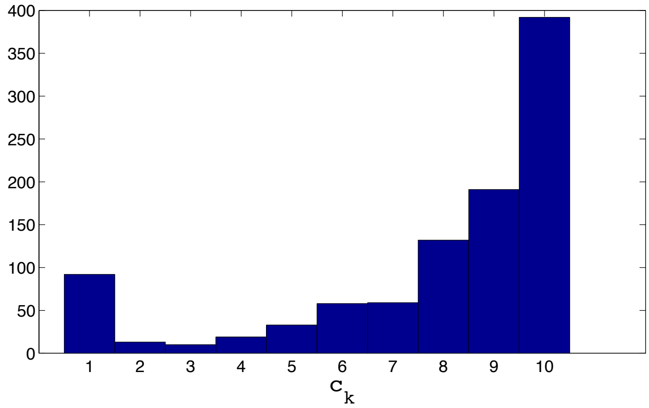

Fig. 3. Histogram of the controller location c_k for a network with $p = 0.95$.

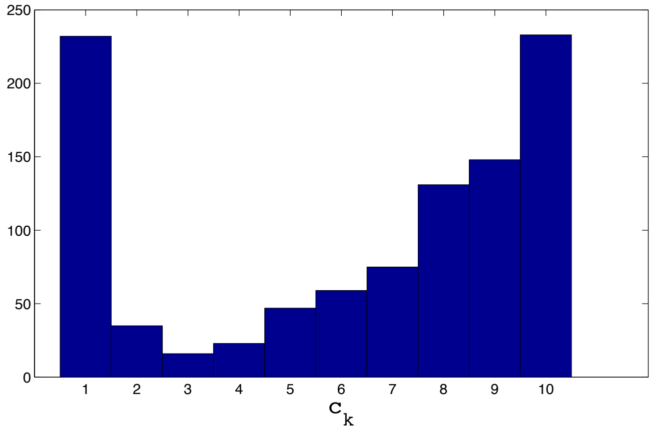

Fig. 4. Histogram of the controller location c_k for a network with $p = 0.97$.

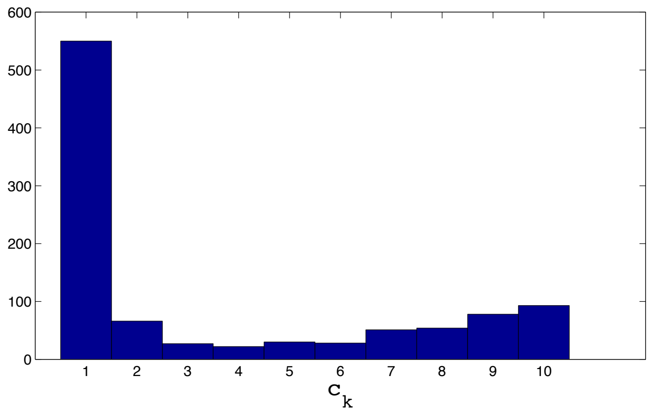

Fig. 5. Histogram of the controller location c_k for a network with $p = 0.99$.

REFERENCES

Antsaklis, P. and Baillieul, J. (2004). Guest editorial special issue on networked control systems. *IEEE Trans. Automat. Contr.*, 49(9), 1421–1423.

Antsaklis, P. and Baillieul, J. (2007). Special issue on technology of networked control systems. *Proc. IEEE*, 95(1), 5–8.

Cardoso de Castro, N., Canudas-de-Wit, C., and Garin, F. (2012). Energy-aware wireless networked control using radio-mode management. In *Proc. Amer. Contr. Conf.*

Censi, A. (2011). Kalman filtering with intermittent observations: Convergence for semi-Markov chains and an intrinsic performance measure. *IEEE Trans. Automat. Contr.*, 56(2), 376–381.

Chen, J., Johansson, K.H., Olariu, S., Paschalidis, I.C., and Stokmenovic (2011). Guest editorial special issue on wireless sensor and actuator networks. *IEEE Trans. Automat. Contr.*, 56(11), 2244–2246.

Franceschetti, M., Javidi, T., Kumar, P.R., Mitter, S., and Teneketzis, D. (2008). Guest editorial control and communications. *IEEE J. Select. Areas Commun.*, 26(4), 577–579.

Garone, E., Sinopoli, B., and Casavola, A. (2010). LQG control over lossy TCP-like networks with probabilistic packet acknowledgements. *Int. J. Syst. Contr. and Commun.*, 2(1/2/3), 55–81.

Goldsmith, A. (2005). *Wireless Communications*. Cambridge University Press.

Goodwin, G.C., Quevedo, D.E., and Silva, E.I. (2008). Architectures and coder design for networked control systems. *Automatica*, 44(1), 248–257.

Gupta, A., Langbort, C., and Başar, T. (2010). Optimal control in the presence of an intelligent jammer with limited actions. In *Proc. IEEE Conf. Decis. Contr.*, 1096–1101. Atlanta, GA.

Gupta, V., Dana, A.F., Hespanha, J.P., Murray, R.M., and Hassibi, B. (2009). Data transmission over networks for estimation and control. *IEEE Trans. Automat. Contr.*, 54(8), 1807–1819.

Huang, M. and Dey, S. (2007). Stability of Kalman filtering with Markovian packet losses. *Automatica*, 43(4), 598–607.

Imer, O.C., Yüksel, S., and Başar, T. (2006). Optimal control of LTI systems over unreliable communication links. *Automatica*, 42(9), 1429–1439.

Pajic, M., Sundaram, S., Pappas, G.J., and Mangharam, R. (2011). The wireless control network: A new approach for control over networks. *IEEE Trans. Automat. Contr.*, 56(10), 2305–2318.

Pantazis, N.A. and Vergados, D.D. (2007). A survey on power control issues in wireless sensor networks. *IEEE Commun. Surv. Tutorials*, 9(4), 86–107.

Park, P.G., Fischione, C., and Johansson, K.H. (2008). Experimental evaluation of power control algorithms for wireless sensor networks. In *Proc. IFAC World Congr.* Seoul, Korea.

Quevedo, D.E., Ahlén, A., and Østergaard, J. (2010). Energy efficient state estimation with wireless sensors through the use of predictive power control and coding. *IEEE Trans. Signal Processing*, 58(9), 4811–4823.

Robinson, C.L. and Kumar, P.R. (2008). Optimizing controller location in networked control systems with packet drops. *IEEE J. Select. Areas Commun.*, 26(4), 661–671.

Silva, E.I. and Pulgar, S.A. (2011). Control of LTI plants over erasure channels. *Automatica*, 47, 1729–1736.

Sinopoli, B., Schenato, L., Franceschetti, M., Poolla, K., Jordan, M.I., and Sastry, S.S. (2004). Kalman filtering with intermittent observations. *IEEE Trans. Automat. Contr.*, 49(9), 1453–1464.

Smith, R. (2011). A decoupled feedback structure for covertly appropriating networked control systems. In *Proc. IFAC World Congr.*, 90–95. Milan, Italy.