Preprints of the 19th World Congress
The International Federation of Automatic Control
Cape Town, South Africa. August 24-29, 2014

Network Topology Reconfiguration for State Estimation Over Sensor Networks With Correlated Packet Drops

Alex S. Leong * **Daniel E. Quevedo** ** **Anders Ahlén** *** **Karl H. Johansson** ****

* *Department of Electrical and Electronic Engineering, University of Melbourne, Australia (e-mail: asleong@unimelb.edu.au).*
** *School of Electrical Engineering and Computer Science, University of Newcastle, Australia (e-mail: dquevedo@ieee.org)*
*** *Signals and Systems, Uppsala University, Uppsala, Sweden (e-mail: anders.ahlen@signal.uu.se)*
**** *ACCESS Linnaeus Centre, School of Electrial Engineering, Royal Institute of Technology, Stockholm, Sweden (e-mail: kallej@ee.kth.se)*

Abstract: This paper studies the problem of optimal network topology reconfiguration in sensor networks for state estimation. Multiple sensors make observations of a process, which are then transmitted, possibly via intermediate sensors, to a central gateway. Transmission over each link can experience packet drops. The time-varying wireless network environment is modelled by the notion of a network state as in Quevedo et al. (2013a). For each network state, different network configurations can be used, which govern the network topology and routing of packets. Switching between different configurations incurs a cost, in that unwanted links will need to be removed before the establishment of new links, leading to a transient time in which some links may not be available. The problem is to determine the optimal network configurations to use in each network state, in order to minimize an expected error covariance measure that takes into account the cost of reconfiguration. A simpler sub-optimal method which minimizes the upper bound of the expected error is also proposed, which in numerical simulations gives essentially identical results to the optimal method.

1. INTRODUCTION

Wireless sensor networks, which consist of a number of small and inexpensive sensors connected together via wireless links, have had many applications in recent years, for instance in environmental and industrial monitoring. The problem of estimation using wireless sensor networks has been an active research area, due to the unreliable nature of wireless links and the associated stability and performance issues.

The modelling of the wireless link as a packet dropping link, in which transmission of packets is assumed to be dropped or lost if the link is of poor quality, is common in the control literature. Kalman filtering for a single sensor over a packet dropping link has been considered in e.g. Sinopoli et al. (2004); Huang and Dey (2007); Epstein et al. (2008); Schenato (2008), to name a few. Extensions to multiple sensors include Liu and Goldsmith (2004); Gupta et al. (2009b), and to sensor networks with various different architectures such as Gupta et al. (2009a); Chiuso and Schenato (2011); Quevedo et al. (2012, 2013b). Kalman filtering over networks with tree structures include Shi (2009); Mo et al. (2011), which don't consider packet drops, and Quevedo et al. (2013a) which does. The work of Quevedo et al. (2013a) introduces the notion of a *network state*, which models time variations in the wireless environment, for example due to moving machines and robots in a factory. This network state process can be either a Markov chain or a semi-Markov process, which was also considered for a single sensor case in Censi (2011).

In Quevedo et al. (2013a) the network topology, i.e. which sensors communicate to each other and how packets are routed through the network, is assumed to be fixed even over different network states. In practice, if in a certain network state some links are of poor quality, e.g. a robot in a factory is blocking the line of sight between two sensors, then the sensors can possibly bypass these links by seeking different paths through the network, as is often done in networking by rerouting, see Bertsekas and Gallager (1992); Kurose and Ross (2012).

In this paper, we consider the problem of determining the optimal network topology configurations to use in each network state. The problem is complicated by the fact that network topology reconfigurations do not occur instantly, but may incur a cost, see Baskaran et al. (2007); Krasteva et al. (2011); Ramakrishnan et al. (2012) for examples of different cost functions. The cost of reconfiguration we consider in this paper is that in changing from one configuration to another, unwanted links will need to be removed before new links can be established (Baskaran et al. (2007)). This leads to a transient time where some links may not be available, leading to poor performance. We optimize an expected error covariance measure over the possible network configurations, taking into account this transient state when switching between different configurations. Computation of expected error covariances can be computationally demanding, so we also consider optimizing an upper bound to the expected error covariance.

The paper is organized as follows. The system model is described in Section 2. The optimal network reconfiguration problem is stated in Section 3, with computational issues discussed in Section 3.3 and a suboptimal method given in Section 3.4. A numerical example is studied in detail in Section 4.

2. SYSTEM MODEL

The process is a discrete time linear system of the form

$$x(k+1) = Ax(k) + w(k)$$

where $x(k) \in \mathbb{R}^n$ and $w(k) \sim N(0, Q)$ is i.i.d. over time. The process is observed by M sensors, with measurements

$$y_m(k) = C_m x(k) + v_m(k), \quad m = 1, \ldots, M$$

where $y_m(k) \in \mathbb{R}^{l_m}$, and $v_m(k) \sim N(0, R_m)$ are i.i.d. over time. We assume that $\{w\}$ and $\{v_m\}, m = 1, \ldots, M$ are mutually independent. We make the standard assumption that (A, C) is detectable and $(A, Q^{1/2})$ is stabilizable, where $C \triangleq \text{col}(C_1, \ldots, C_M)$.

2.1 Sensor network model

We consider the case where the sensors are connected together to form a sensor network with a gateway/fusion center. The sensor network in general is assumed to have a mesh structure. Sensor measurements are to be transmitted to the gateway which runs a Kalman filter. The paths used in transmitting from the sensors to the gateway are usually computed using routing algorithms. We assume that the links which are utilized in the set of routes from the sensors to the gateway, which we denote as the set of *active links*, has a tree structure (i.e. has no cycles).[1] In this model, the set of active links can be described using a directed graph with nodes/vertices $\{S_0, S_1, \ldots, S_M\}$, where the root node S_0 denotes the gateway, and $S_m, m = 1, \ldots, M$ denote the sensors. See Fig. 1 for an example with nine nodes (eight sensors and a gateway). Each sensor aggregates its own measurement to the received

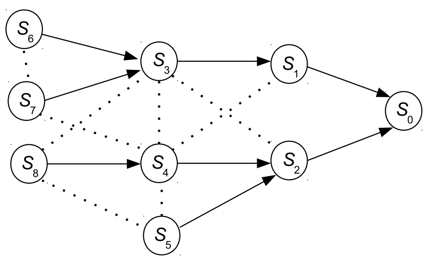

Fig. 1. Sensor network with nine nodes. The set of active links represented by arrows forms a tree, while the dotted lines represent inactive links.

packets from incoming nodes and transmits the resulting packet to a single destination node. We assume that transmissions can occur over a faster time scale than the process, as is typical in the industrial wireless sensor networks standard WirelessHART (HART Communication Foundation (2009a)), thus delays experienced in travelling through the network will be ignored. Following the notation of Quevedo et al. (2013a), we call the node that sensor S_m transmits to the *parent* of S_m, denoted by $\text{Par}(S_m)$. The outgoing link/edge from each of the nodes will be denoted as $\mathcal{E}_m = (S_m, \text{Par}(S_m)), m = 1, \ldots, M$. For a given tree, there is a unique path from each node S_m to the gateway S_0, denoted by $\text{Path}(S_m)$. We will call the set of edges and the set of nodes along this path by $\text{Edge}(\text{Path}(S_m))$ and $\text{Node}(\text{Path}(S_m))$ respectively.

[1] For instance, this will be the case when using shortest path type routing algorithms (Bertsekas and Gallager (1992)).

In this paper, due to changes in the wireless environment, the network topology formed by the set of active links can change over time because of reconfigurations of the network. As in Quevedo et al. (2013a), changes in the environment will be modelled by the notion of a *network state* process $\Xi(k) \in \mathbb{B} \triangleq \{1, 2, \ldots |\mathbb{B}|\}$, which is time-varying. As motivation for this idea, consider Fig. 2, which plots some fading channel measurements taken at a rolling mill at Sandvik in Sweden. Mobile machines cause infrequent but substantial variations in the expected channel gains.

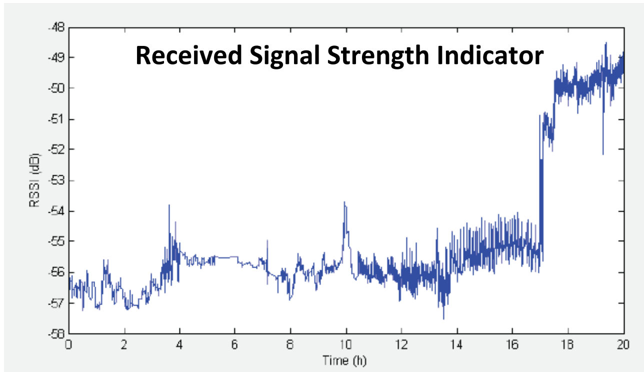

Fig. 2. Channel measurements taken at a rolling mill

We will assume in this paper that $\{\Xi\}$ is a discrete-time semi-Markov process (Ross (1996)), to model situations where network state transitions do not necessarily have to occur at every time instant k. The times of transitions between different network states is denoted by

$$\mathbb{K} \triangleq \{k_l\}, \text{with } k_0 = 0, \text{ and } k_0 < k_1 < k_2 \ldots \text{ all integers.}$$

The *holding times* (the amount of time spent in a particular state between transitions) are defined as

$$\Delta_l \triangleq k_{l+1} - k_l.$$

We assume that the holding times have finite support, thus

$$\Delta_l \leq \Delta_{max}, \quad \forall l$$

By the semi-Markov property, we have

$$\begin{aligned} &\mathbb{P}(\Xi(k_{l+1}) = j, \Delta_l = \delta | \Xi(k_0), \ldots, \Xi(k_{l-1}), \Xi(k_l) = i, \\ &\quad k_0, \ldots, k_l) \\ &= \mathbb{P}(\Xi(k_{l+1}) = j | \Xi(k_l) = i) \mathbb{P}(\Delta_l = \delta | \Xi(k_l) = i) \\ &= q_{ij} \psi_i(\delta), \quad \forall (k_l, \delta, i, j) \in \mathbb{K} \times \mathbb{N} \times \mathbb{B} \times \mathbb{B} \end{aligned}$$

where $q_{ij} \triangleq \mathbb{P}(\Xi(k_{l+1}) = j | \Xi(k_l) = i)$ are the transition probabilities of the embedded Markov chain, and

$$\psi_i(\delta) \triangleq \mathbb{P}(\Delta_l = \delta | \Xi(k_l) = i) \quad (1)$$

are the conditional probabilities of the holding times.

The *network configuration* $\pi(k)$ at time k determines which nodes each sensor will receive from and forward to, i.e. the set of routes from the sensors to the gateway. The set of all possible configurations is denoted by $\Pi = \{1, 2, \ldots, |\Pi|\}$. Depending on channel link conditions, some configurations may not be feasible in certain network states. We call $\Pi_j \subseteq \Pi$ the set of possible configurations when the network state is equal to j. Thus $(\Xi(k), \pi(k)) \in \mathbb{B} \times \Pi_j$ for $\Xi(k) = j$. We assume that the set of all possible configurations have been precomputed and are known at the gateway. This could be the case in some industrial applications where the number of sensors is relatively small. For instance, WirelessHART provides redundancy by maintaining multiple sets of routes which the network can switch to at different time instances (HART Communication Foundation (2009b)). [2]

Define the random variables $\gamma_m(k), m = 1, \ldots, M$ by

$$\gamma_m(k) = \begin{cases} 1, & \text{transmission via link } \mathcal{E}_m \text{ at time } k \text{ is successful} \\ 0, & \text{otherwise} \end{cases}$$

and the corresponding link success probabilities by

$$\phi_{m|(j,p)} \triangleq \mathbb{P}(\gamma_m(k) = 1 | \Xi(k) = j, \pi(k) = p)$$

We will assume in this paper that, *conditioned on a network state*, $\{\gamma_m\}$ are i.i.d. Bernoulli processes, with $\{\gamma_m\}$ independent of $\{\gamma_n\}$ for $m \neq n$. Note that in this model there is temporal correlation, in that the packet reception probabilies can be different in different network states. The special case of i.i.d. Bernoulli processes (no temporal correlation) is often considered in the literature, see e.g. Sinopoli et al. (2004); Liu and Goldsmith (2004); Epstein et al. (2008); Schenato (2008). Markovian packet losses as studied in e.g. Huang and Dey (2007); You et al. (2011) can also be regarded as a special case of this model, see Quevedo et al. (2013a) for details.

2.2 Kalman filter at Gateway

Define the random variables $\theta_m(k), m = 1, \ldots, M$ by

$$\theta_m(k) = \begin{cases} 1, & \text{transmission via Path}(S_m) \text{ at time } k \text{ is successful} \\ 0, & \text{otherwise} \end{cases}$$

Due to the fact that the set of active links forms a tree, we have

$$\theta_m(k) = \prod_{\mathcal{E}_i \in \text{Edge}(\text{Path}(S_m))} \gamma_i(k)$$

and $\mathbb{P}(\theta_m(k) = 1 | \Xi(k) = j, \pi(k) = p) = \prod_{\mathcal{E}_i \in \text{Edge}(\text{Path}(S_m))} \phi_{i|(j,p)}$.

Let $\theta(k) \triangleq [\, \theta_1(k) \ \ldots \ \theta_M(k) \,]^T$,

$$y(k) \triangleq \begin{bmatrix} \theta_1(k) y_1(k) \\ \vdots \\ \theta_M(k) y_M(k) \end{bmatrix}, \quad C(k) \triangleq \begin{bmatrix} \theta_1(k) C_1 \\ \vdots \\ \theta_M(k) C_M \end{bmatrix}$$

The Kalman filtering equations can then be written as:

$$\begin{aligned} \hat{x}(k+1|k) &= A\hat{x}(k|k-1) + K(k)(y(k) - C(k)\hat{x}(k|k-1)) \\ P(k+1|k) &= AP(k|k-1)A^T + Q - K(k)C(k)P(k|k-1)A^T \end{aligned} \tag{2}$$

where $R \triangleq \text{diag}(R_1, \ldots, R_M)$ and

$K(k) \triangleq AP(k|k-1)C(k)^T \big(C(k)P(k|k-1)C(k)^T + R\big)^{-1}$.

In the sequel, we will use the shorthand $P(k) \triangleq P(k|k-1)$.

3. NETWORK RECONFIGURATION

As stated in the previous section, the network states model changes in the wireless environment, with changes in network states occurring at the random transition times $k_0, k_1, k_2, \ldots$. Due to changes in the wireless environment, the packet reception probabilities of existing links can change, and there could even be a complete loss of connectivity in some links. The purpose of the present work is to illustrate how to compensate for changes in the wireless environment through network reconfiguration. At each new transition time instant k_l when the network state changes, we wish to choose a new configuration $\pi(k_l)$ in order to minimize an expected error covariance performance measure. However, network configuration changes are not immediate and there is often a cost involved in reconfiguring the network, as explained below.

3.1 Reconfiguration Issues

In what follows, we will use a similar cost of reconfiguration as in Baskaran et al. (2007), where in changing from one configuration to another, unwanted links will need to be removed before the establishment of new links. We will refer to this as a *transient state*. Thus there is a transient time or *reconfiguration time* $T \geq 0$ where some links will not be available, resulting in poor transitory performance of the Kalman filter.

Example 1. Consider for example the network configuations shown in Fig. 3 (see Section 4 for the full set of network configurations). In reconfiguring from network configuration 2 to network configuration 3, the links from sensor 3 to sensor 2, and from sensor 4 to sensor 2, will first need to be removed, leading to the transient state shown in Fig. 3 where sensors 3 and 4 do not have connectivity to the rest of the network for some time T. Similarly, reconfiguring from network configuration 3 to configuration 2 will also lead to the same transient state. □

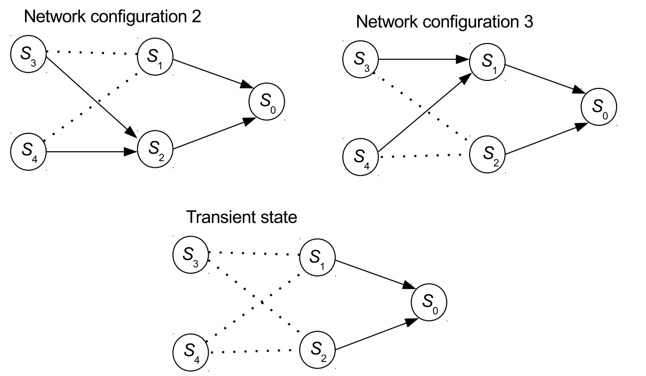

Fig. 3. Transient state when reconfiguring between two network configurations, see Example 1.

The value of T is dependent on the underlying communication technology. For instance, in IEEE 802.11 the time needed to reroute a wireless network could be on the order of seconds (Pham et al. (2007)). On the order hand, in WirelessHART which actively maintains multiple routes that can be switched at different time instances (HART Communication Foundation (2009b)), it might be more appropriate to take $T = 0$. See also Baskaran et al. (2007) and the references therein for the reconfiguration time of optical networks. Therefore, in choosing which new configuration to use, there is potentially a tradeoff between a configuration that gives good performance (after it is fully reconfigured) but requires many link changes, versus a configuration that has fewer link changes but slightly

[2] Another possible form of redundancy is by transmitting the same information along multiple paths. However, due to large overheads this is usually not implemented in practice.

poorer performance. We will state the problem formally in the next subsection.

3.2 Optimal network reconfiguration

At each transition instant $k_l \in \mathbb{K}$, we seek to find a network configuration $\pi(k_l)$ which is to be held until the next transition instant $k_{l+1} \in \mathbb{K}$, and which minimizes the expected state estimation covariance over this holding period. Here we will assume that the gateway has knowledge of the current error covariance $P(k_l)$, the old network configuration $\pi(k_{l-1})$, and the current network state $\Xi(k_l)$. For ease of exposition, we introduce the aggregated process

$$\mathcal{U}(k_l) \triangleq \big(P(k_l), \Xi(k_l), \pi(k_{l-1})\big), \quad k_l \in \mathbb{K}. \tag{3}$$

In terms of $\mathcal{U}(k_l)$, the new configuration $\pi(k_l) \in \Pi_j$ when $\Xi(k_l) = j$ is found via the following optimization:

$$\pi(k_l) = \arg\min_{\pi \in \Pi_j} \mathcal{V}(\mathcal{U}(k_l), \pi), \quad \text{where}$$
$$\mathcal{V}(\mathcal{U}(k_l), \pi) = \mathbb{E}\Big\{ \sum_{i=1}^{\Delta_l} \operatorname{tr} P(k_l + i) \Big| \mathcal{U}(k_l) \Big\}. \tag{4}$$

The quantity $\mathcal{V}(\mathcal{U}(k_l), \pi)$ amounts to the expected trace of the error covariance over the holding time Δ_l, when the configuration π is used. In (4), the expectation is taken over both the packet loss processes (which affect the Kalman filter recursions (2)) and the random holding times (using (1)).

Following the model in Section 2.1, the network state $\Xi(k_l)$ determines the distribution of the holding time and thereby the upper limit of the sum in (4), see (1); differences between the decision variable π and the previous configuration $\pi(k_l - 1)$ determine which links would be moved to a transient state. In particular, the terms

$$\mathbb{E}\left\{P(k_l + i) \,|\, \mathcal{U}(k_l)\right\} \tag{5}$$

are computed based on whether the network is still in the transient mode (if $i \leq T$) or has been fully reconfigured (if $i > T$), with the expectation over the discrete variables $\{\theta(k_l), \dots, \theta(k_l + i - 1)\}$.

Remark 1. Given the semi-Markov network model adopted, the reconfiguration strategy (4) yields that the process $\mathcal{U}(k_l)$ *at the transition instants* $k_l \in \mathbb{K}$ is Markovian. This opens the possibility of analyzing *estimator stability* (see Sinopoli et al. (2004)) by adapting the methods developed in Quevedo et al. (2013a). These results will be detailed in future work.

3.3 Computational Aspects

In principle, minimization of (4) can be carried out by checking the values of $\mathcal{V}(\mathcal{U}(k_l), \pi)$ for each of the different configurations $\pi \in \Pi_j$. However, computation of the expectations (5) involves considering the values of $P(k_l + i)$ for all possible combinations of $\{\theta(k_l), \dots \theta(k_l + i - 1)\}$, with the number of possibilities being $O(2^{Mi})$ in general. In particular, computing $\mathbb{E}\left\{P(k_l + \Delta_{\max}) \,|\, \mathcal{U}(k_l)\right\}$ will have a complexity of $O(2^{M\Delta_{\max}})$. Thus, for large holding times, minimization of (4) is computationally intensive.

3.4 Suboptimal network reconfiguration

To address the computational issues outlined above, we propose to adopt a suboptimal approach wherein, using $\mathcal{U}(k_l)$ defined as in (3), the new configuration $\pi(k_l) \in \Pi_j$ is obtained via

$$\pi(k_l) = \arg\min_{\pi \in \Pi_j} \mathcal{W}(\mathcal{U}(k_l), \pi), \quad \text{where}$$
$$\mathcal{W}(\mathcal{U}(k_l), \pi) = \sum_{\delta=1}^{\Delta_{\max}} \sum_{i=1}^{\delta} \operatorname{tr} Y(k_l + i) \mathbb{P}\{\Delta_l = \delta \,|\, \Xi(k_l) = j\}, \tag{6}$$

Here, the sequence $\{Y(k_l + 1), Y(k_l + 2), \dots\}$ is given by:

$$Y(k+1) = AY(k)A^T + Q$$
$$- \mathbb{E}[AY(k)C(k)^T(C(k)Y(k)C(k)^T + R)^{-1}C(k)Y(k)A^T] \tag{7}$$

for $k \geq k_l$, with initial condition $Y(k_l) = P(k_l)$, and where the expectation is with respect to the random matrix $C(k)$. Again, (7) is computed taking into account whether the network is still in the transient mode or has been fully reconfigured.

The following result, linking the optimal and the sub-optimal approaches is easy to show:

Lemma 1. The sequence $Y(k)$ is an upper bound to $\mathbb{E}\{P(k)|\mathcal{U}(k_l)\}$ for $k \geq k_l$.

Proof Define

$$g_k(X) = AXA^T + Q - \mathbb{E}[AX\tilde{C}_k^T(\tilde{C}_k X \tilde{C}_k^T + R)^{-1}\tilde{C}_k X A^T]$$

where $\tilde{C}_k$ is a random matrix having the same distribution as $C(k)$. Lemma 1 is proved by using the fact that $g_k(.)$ is concave in X, and induction. The concavity of $g_k(.)$ is shown by using similar techniques as in Sinopoli et al. (2004); Dey et al. (2009). The details are omitted for brevity. ■

Upper bounding sequences of the form (7) are much easier to compute than the expected error covariance when the holding times are large. Furthermore, the bounds generally seem to be quite tight, see, e.g., Leong and Quevedo (2013). In the next section we will look at a numerical example, where we will see that the new configurations obtained using the suboptimal method are essentially identical to the configurations obtained using the optimal scheme.

4. SIMULATION STUDY

We consider a simple example with four sensor nodes. The set of all network configurations are shown in Fig. 4. There are two network states, with network configurations 1 and 2 possible in network state 1 (so that $\Pi_1 = \{1, 2\}$), and network configurations 1 and 3 possible when in network state 2 (so that $\Pi_2 = \{1, 3\}$). The reconfiguration time is taken to be fixed at $T = 1$. The packet reception probabilities for the links in each of the network configurations are:

$$\phi_{1|(1,1)} = 0.5, \phi_{2|(1,1)} = 0.5, \phi_{3|(1,1)} = 0.2, \phi_{4|(1,1)} = 0.5$$
$$\phi_{1|(1,2)} = 0.5, \phi_{2|(1,2)} = 0.5, \phi_{3|(1,2)} = 0.8, \phi_{4|(1,2)} = 0.5$$
$$\phi_{1|(2,1)} = 0.5, \phi_{2|(2,1)} = 0.5, \phi_{3|(2,1)} = 0.5, \phi_{4|(2,1)} = 0.2$$
$$\phi_{1|(2,3)} = 0.5, \phi_{2|(2,3)} = 0.5, \phi_{3|(2,3)} = 0.5, \phi_{4|(2,3)} = 0.8 \tag{8}$$

Network state 1 could correspond to the case where there is a robot between sensor nodes 1 and 3, giving a low probability of packet reception of 0.2 for the direct link (from sensor 3 to sensor 1) in network configuration 1, while in network configuration 2 sensor 3 will instead transmit to sensor 2 with a higher packet reception probability of 0.8. Similarly network state 2 will correspond to the case where the robot is now situated between sensors 2 and 4.

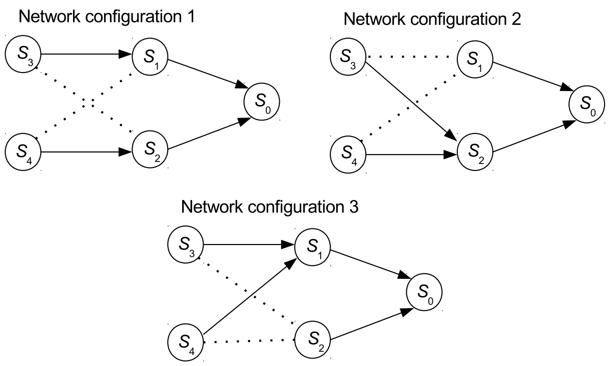

Fig. 4. Network configurations for example of Section 4

The holding times have the following distribution:

$$\mathbb{P}(\Delta_l = 1|\Xi(k_l) = 1) = \mathbb{P}(\Delta_l = 1|\Xi(k_l) = 2) = 0.1$$
$$\mathbb{P}(\Delta_l = 2|\Xi(k_l) = 1) = \mathbb{P}(\Delta_l = 2|\Xi(k_l) = 2) = 0.1$$
$$\mathbb{P}(\Delta_l = 3|\Xi(k_l) = 1) = \mathbb{P}(\Delta_l = 3|\Xi(k_l) = 2) = 0.1$$
$$\mathbb{P}(\Delta_l = 4|\Xi(k_l) = 1) = \mathbb{P}(\Delta_l = 4|\Xi(k_l) = 2) = 0.7$$

The transition probabilities for the embedded Markov chain $\{\Xi(k_l)\}$, $k_l \in \mathbb{K}$ are

$$\mathbb{P}(\Xi(k_{l+1}) = 1|\Xi(k_l) = 1) = q_{11} = 0.5$$
$$\mathbb{P}(\Xi(k_{l+1}) = 2|\Xi(k_l) = 1) = q_{12} = 0.5$$
$$\mathbb{P}(\Xi(k_{l+1}) = 1|\Xi(k_l) = 2) = q_{21} = 0.5$$
$$\mathbb{P}(\Xi(k_{l+1}) = 2|\Xi(k_l) = 2) = q_{22} = 0.5$$

We consider a system with parameters

$$A = \begin{bmatrix} 1.1 & 0.2 \\ 0.2 & 0.8 \end{bmatrix}, \quad Q = \begin{bmatrix} 0.2 & 0 \\ 0 & 0.2 \end{bmatrix},$$

$C_1 = C_2 = C_3 = C_4 = [1\ 1]$, $R_1 = R_2 = 10$, $R_3 = R_4 = 0.1$. The differences in the sensor measurement noise covariances correspond to the situation where the process to be observed is located much closer to sensors 3 and 4, than to sensors 1 and 2.

In Fig. 5 we plot the simulated values of $\mathrm{tr}\, P(k+1)$ obtained by solving the network reconfiguration problem (4). We also include the case where only network configuration 1 is used when in both network states 1 and 2, which can be regarded as the case of no reconfiguration. We see that there are times where the optimal reconfiguration has error covariance that is either larger or smaller than the case of no reconfiguration, illustrating the tradeoff mentioned at the end of Section 3.1. From Monte Carlo simulations, the trace of the average error covariance is around 1.56, whereas the case of no reconfiguration is around 1.87, which amounts to a performance gain of about 20%.

Fig. 6 illustrates the corresponding network states $\Xi(k)$ and Fig. 7 the corresponding network configurations $\pi(k)$ used at each time instance. For this example, the network configurations obtained using the suboptimal method of Section 3.4 by solving problem (6) is identical to Fig. 7. It seems that for the packet reception probabilities given in (8), network configuration 1 will not be chosen. However, different behaviour can be observed by modifying these probabilites. In Fig. 8 we plot the simulation run of the network configurations used at each time instance, with the same network states as in Fig. 6, but now with $\phi_{3|(1,1)} = 0.45$ and $\phi_{4|(2,1)} = 0.45$, so that the probability of packet reception in these two links (for these network states) is

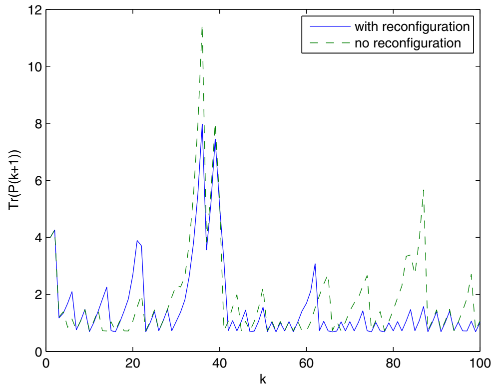

Fig. 5. Error covariances at different time instances

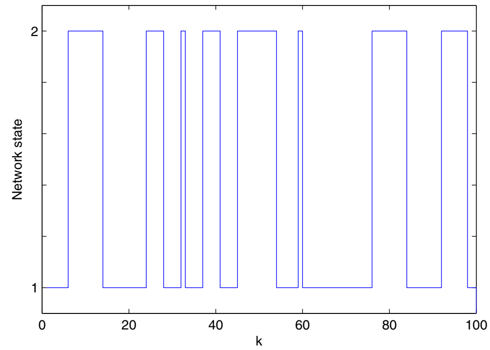

Fig. 6. Network states at different time instances

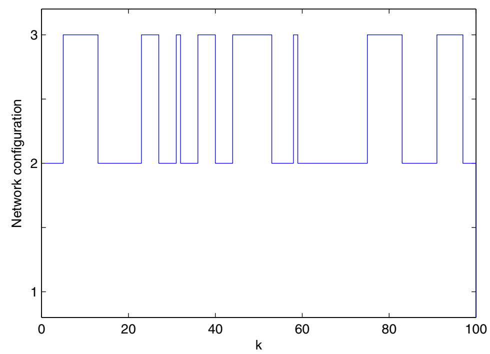

Fig. 7. Network configurations at different time instances using both proposed methods, given success probabilities as in (8).

increased. Simlarly, in Fig. 9 we plot the network configurations used when $\phi_{3|(1,1)} = 0.6$ and $\phi_{4|(2,1)} = 0.6$. Again, the network configurations obtained using the suboptimal method of Section 3.4 are identical to Fig. 8 and Fig. 9. We see that as $\phi_{3|(1,1)}$ and $\phi_{4|(2,1)}$ are increased, the network is less likely to reconfigure. The cost of reconfiguration which causes links to be lost in the transient state leads to the network not changing its topology.

5. CONCLUSION

We have presented a network topology reconfiguration method for state estimation in sensor networks. Network reconfigura-

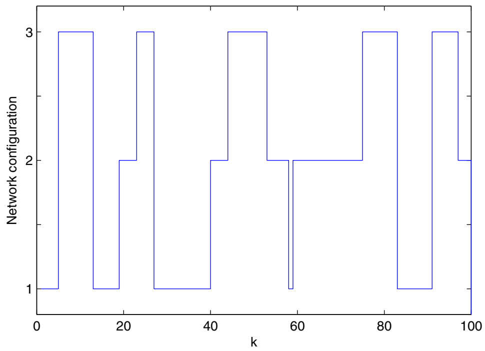

Fig. 8. Network configurations at different time instances: $\phi_{3|(1,1)} = 0.45$ and $\phi_{4|(2,1)} = 0.45$

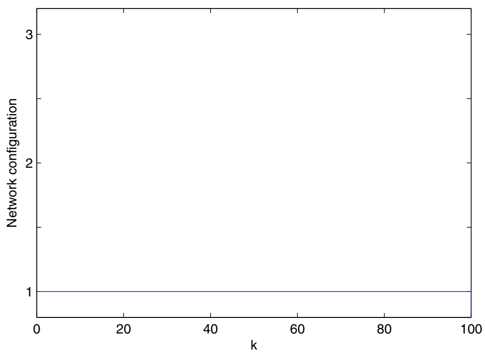

Fig. 9. Network configurations at different time instances: $\phi_{3|(1,1)} = 0.6$ and $\phi_{4|(2,1)} = 0.6$

tions are triggered when the wireless environment, modelled by the notion of a network state, changes. The optimization of an expected error performance measure which takes into account the cost of reconfiguration has been studied, and a less computationally intensive suboptimal method proposed. Future work will include the derivation of stability conditions for the estimation scheme with reconfigurations, and consideration of more general fading channel distributions.

REFERENCES

Baskaran, E., Llorca, J., Milner, S.D., and Davis, C.C. (2007). Topology reconfiguration with successive approximations. In *Proc. MILCOM*. Orlando, FL.

Bertsekas, D. and Gallager, R. (1992). *Data Networks*. Prentice-Hall, New Jersey, 2nd edition.

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

Chiuso, A. and Schenato, L. (2011). Information fusion strategies and performance bounds in packet-drop networks. *Automatica*, 47, 1304–1316.

Dey, S., Leong, A.S., and Evans, J.S. (2009). Kalman filtering with faded measurements. *Automatica*, 45(10), 2223–2233.

Epstein, M., Shi, L., Tiwari, A., and Murray, R.M. (2008). Probabilistic performance of state estimation across a lossy network. *Automatica*, 44, 3046–3053.

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

Gupta, V., Martins, N.C., and Baras, J.S. (2009b). Optimal output feedback control using two remote sensors over erasure channels. *IEEE Trans. Autom. Control*, 54(7), 1463–1476.

HART Communication Foundation (2009a). Control with WirelessHART.

HART Communication Foundation (2009b). System redundancy with WirelessHART.

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

Krasteva, Y.E., Portilla, J., de la Torre, E., and Riesgo, T. (2011). Embedded runtime reconfigurable nodes for wireless sensor networks applications. *IEEE Sensors Journal*, 11(9), 1800–1810.

Kurose, J.F. and Ross, K. (2012). *Computer Networking: A Top-Down Aproach*. Pearson, 6th edition.

Leong, A.S. and Quevedo, D.E. (2013). On the use of a relay for Kalman filtering over packet dropping links. In *Proc. ACC*. Washington, DC.

Liu, X. and Goldsmith, A.J. (2004). Kalman filtering with partial observation losses. In *Proc. IEEE Conf. Decision and Control*, 1413–1418. Bahamas.

Mo, Y., Garone, E., Casavola, A., and Sinopoli, B. (2011). Stochastic sensor scheduling for energy constrained estimation in multi-hop wireless sensor networks. *IEEE Trans. Autom. Control*, 56(10), 2489–2495.

Pham, V., Larsen, E., Øvsthus, K., Engelstad, P., and Kure, Ø. (2007). Rerouting time and queueing in proactive ad hoc networks. In *Proc. IPCCC*, 160–169. New Orleans, LA.

Quevedo, D.E., Ahlén, A., and Johansson, K.H. (2013a). State estimation over sensor networks with correlated wireless fading channels. *IEEE Trans. Autom. Control*, 58(3), 581–593.

Quevedo, D.E., Ahlén, A., Leong, A.S., and Dey, S. (2012). On Kalman filtering over fading wireless channels with controlled transmission powers. *Automatica*, 48(7), 1306 – 1316.

Quevedo, D.E., Østergaard, J., and Ahlén, A. (2013b). Power control and coding formulation for state estimation with wireless sensors. *IEEE Trans. Control Syst. Technol.* To be published. Available at http://arxiv.org/abs/1308.1745.

Ramakrishnan, R., Ram, N.S., and Alheyasat, O.A. (2012). A cost aware reconfiguration technique for recovery in wireless mesh networks. In *Proc. ICRTIT*, 302–307. Chennai, India.

Ross, S.M. (1996). *Stochastic Processes*. John Wiley & Sons, New York, 2nd edition.

Schenato, L. (2008). Optimal estimation in networked control systems subject to random delay and packet drop. *IEEE Trans. Autom. Control*, 53(5), 1311–1317.

Shi, L. (2009). Kalman filtering over graphs: Theory and applications. *IEEE Trans. Autom. Control*, 54(9), 2230–2234.

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

You, K., Fu, M., and Xie, L. (2011). Mean square stability for Kalman filtering with Markovian packet losses. *Automatica*, 47(12), 1247–1257.