On Network Topology Reconfiguration for Remote State Estimation

Alex S. Leong, *Member, IEEE*, Daniel E. Quevedo, *Senior Member, IEEE*, Anders Ahlén, *Senior Member, IEEE*, and Karl H. Johansson, *Fellow, IEEE*

***Abstract*—In this paper, we investigate network topology reconfiguration in wireless sensor networks for remote state estimation, where sensor observations are transmitted, possibly via intermediate sensors, to a central gateway/estimator. The time-varying wireless network environment is modelled by the notion of a network state process, which is a randomly time-varying semi–Markov chain and determines the packet reception probabilities of links at different times. For each network state, different network configurations can be used, which govern the network topology and routing of packets. The problem addressed is to determine the optimal network configuration to use in each network state, in order to minimize an expected error covariance measure. Computation of the expected error covariance cost function has a complexity of $O(2^{M\Delta_{\max}})$, where M is the number of sensors and $\Delta_{\max}$ is the maximum time between transitions of the semi–Markov chain. A sub-optimal method which minimizes the upper bound of the expected error covariance, that can be computed with a reduced complexity of $O(2^M)$, is proposed, which in many cases gives identical results to the optimal method. Conditions for estimator stability under both the optimal and suboptimal reconfiguration methods are derived using stochastic Lyapunov functions. Numerical results and comparisons with other low complexity approaches demonstrate the performance benefits of our approach.**

***Index Terms*—Fading channels, Kalman filtering, network topology reconfiguration, packet drops, sensor networks.**

I. Introduction

Wireless sensor networks consist of a number of small and inexpensive sensors which can communicate with each other over wireless links. In conjunction with advances in microelectronic technology in recent years, sensor networks have found many applications, e.g., in environmental and infrastructure monitoring, healthcare, military surveillance, and industrial monitoring and control. A major challenge in the deployment of wireless sensor networks is overcoming the time-varying nature of the wireless environment, due to the severe energy, computation and communication constraints on the sensors.

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. Kalman filtering for a single sensor over a packet dropping link was considered in [2], which showed the existence of a critical threshold on the packet arrival probability needed for estimator stability. Extensions of this work include further characterizations of the critical threshold [3], [4], multiple sensors [5]–[7], probabilistic notions of performance [8], Markovian [9], [10] and semi-Markovian [11] packet drops, and consideration of delays [12], to name a few.

Estimation in sensor networks using a variety of different architectures has also been considered. The architecture in [13] consists of one sensor making measurements, which is then transmitted over a lossy network with arbitrary topology. The article [14] looks at decentralized Kalman filtering with packet drops and/or delays. The works in [15], [16] consider one-hop transmission (or a star topology) over packet dropping links, with [15] investigating various different fusion rules, and [16] studying the effect of power control on stability. Sensor network architectures with relays are studied in [17], [18], adopting network coding [19] as a way to improve performance. Kalman filtering over networks with tree structures include [20]–[22], with [20] studying a stochastic sensor scheduling problem, and [21] studying routing algorithms and topology reconfiguration but no packet drops. In [22] the individual links in the tree can be packet dropping, and the notion of a *network state* process is introduced, which models random time variations in the wireless environment, for example due to moving machines and robots in a factory.

In [22] 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. Our work differs from [22] in that we consider the problem of determining the optimal network topology configuration to use in each network state. In [21], reconfiguration from a given topology to a topology with more direct sensor transmissions to the fusion center is studied for networks with no packet drops. In our work, the communication links in the network are packet dropping, and we optimize between a number of pre-computed topologies in our reconfiguration. We further assume that network topology reconfigurations do not occur instantly, but may

Manuscript received March 12, 2015; revised August 23, 2015; accepted January 23, 2016. Date of publication February 11, 2016; date of current version December 2, 2016. A preliminary version of parts of this work was presented at the IFAC World Congress, Cape Town, South Africa, Aug. 2014 [1]. This work was supported by the Australian Research Council under grant DE120102012. Recommended by Associate Editor W. X. Zheng.

A. S. Leong and D. E. Quevedo are with the Department of Electrical Engineering (EIM-E), Paderborn University, 33098 Paderborn, Germany (e-mail: alex.leong@upd.de; dquevedo@ieee.org).

A. Ahlén is with Signals and Systems, Uppsala University, 751 21 Uppsala, Sweden (e-mail: anders.ahlen@signal.uu.se).

K. H. Johansson is with ACCESS Linnaeus Centre, School of Electrical Engineering, Royal Institute of Technology, 100 44 Stockholm, Sweden (e-mail: kallej@ee.kth.se).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TAC.2016.2527788

incur a cost, in that changing from one configuration to another, unwanted links will need to be removed before new links can be established [23] (see also [24], [25] for examples of different cost functions). This leads to a transient time where some links may not be available, and poor transitory performance. The aim is to optimize an expected error covariance measure over the possible network configurations, taking into account this transient state when switching between different configurations. Computation of the expected error covariance cost function used in this paper has a complexity of $O(2^{M\Delta_{\max}})$, where M is the number of sensors and $\Delta_{\max}$ is the maximum time between transitions of the semi-Markov chain modelling the network state process. We also consider a suboptimal approach which optimizes an upper bound to the expected error covariance, with a reduced complexity of $O(2^M)$, which while still exponential in the number of sensors, could be useful in industrial settings where networks often have a hierarchical structure and are divided into smaller sub-networks.

The paper is organized as follows. The system model is described in Section II. The optimal network reconfiguration problem is studied in Section III, with stochastic stability analysis of the scheme given in Section III-D. A suboptimal method for network reconfiguration is proposed in Section IV. Some lower complexity schemes are described in Section V. An illustrative example is given in Section VI. Numerical results and comparisons with the lower complexity approaches of Section V are presented in Section VII. Section VIII draws conclusions.

Notation: We define $\text{col}(X_1,\ldots,X_n) \triangleq [X_1^T \ \ldots \ X_n^T]^T$ to be the matrix formed by stacking the matrices $X_1,\ldots,X_n$ on top of each other, and $\text{diag}(X_1,\ldots,X_n)$ to be the block diagonal matrix with $X_1,\ldots,X_n$ being the diagonal blocks.

II. System Model

The process is a discrete time linear system of the form

$$x(k+1) = Ax(k) + w(k), \quad k \in \mathbb{N}_0 \triangleq \{0,1,2,\ldots\}$$

with A possibly unstable, where $x(k) \in \mathbb{R}^n$, and $w(k)$ is Gaussian with zero mean and covariance matrix Q. The process is observed by M sensors, with measurements

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

where $y_m(k) \in \mathbb{R}^{l_m}$ and $v_m(k)$ is Gaussian with zero mean and covariance matrix R_m. We assume that $\{w\}$ and $\{v_m\}, m = 1,\ldots,M$ are i.i.d. over time (i.e., are discrete time white noise processes [26]) and mutually independent. We make the assumption that (A,C) is detectable and $(A,Q^{1/2})$ is stabilizable, where $C \triangleq \text{col}(C_1,\ldots,C_M)$. However, the individual (A,C_m) pairs are not required to be detectable.

A. Sensor Network Model

We consider the situation where some sensors and a gateway/fusion center are connected to form a sensor network, which in general is assumed to have a mesh structure. Sensor measurements are to be transmitted, possibly via intermediate nodes, to the gateway, which runs a Kalman filter. The paths used by the sensors in transmitting 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 or parallel paths) with the gateway as the root node. This reduces redundancy in transmissions and energy usage, and avoids sensors having to listen to multiple transmissions. For example, a tree structure will be obtained when using shortest path [27] or minimum energy [21] type routing algorithms.

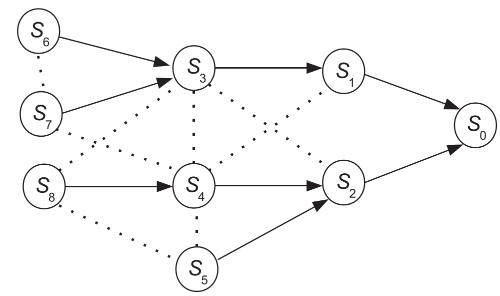

Fig. 1. Sensor network with nine nodes. The set of active links represented by arrows forms a tree, while the dotted lines represent inactive links.

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 packets from incoming nodes and transmits the resulting packet to a single destination node. We assume that transmissions can occur over a much faster time scale than the process, thus delays experienced in travelling through the network will be ignored.[1] We call the node that sensor S_m transmits to the *parent* of S_m, denoted by 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 Path(S_m), with Edges(Path(S_m)) being the corresponding edges.

B. Wireless Channel Model

We model changes in the characteristics of the wireless environment by the notion of a randomly time-varying *network state* process $\Xi(k) \in \mathbb{B} \triangleq \{1,2,\ldots,|\mathbb{B}|\}$. As motivation, consider Fig. 2, which plots some fading channel measurements acquired at a rolling mill at Sandvik in Sweden [29]. We see infrequent but substantial variations in the measured channel gains, due to mobile machinery and cranes in the ceiling blocking the line of sight between certain sensors, or changing the propagation pattern. Different network states can be used to represent the different positions (or similar groups of positions) that the

[1] For instance, in the industrial wireless sensor network standard WirelessHART [28], transmissions between nodes typically take around 10 ms, whereas in many estimation and control applications the process time constant might be 1 sec or more.

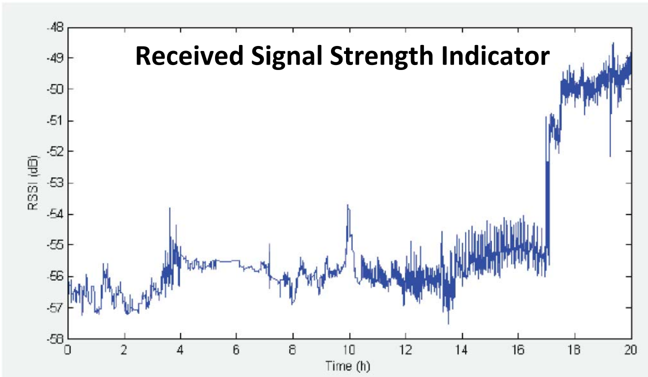

Fig. 2. Channel measurements taken at a rolling mill.

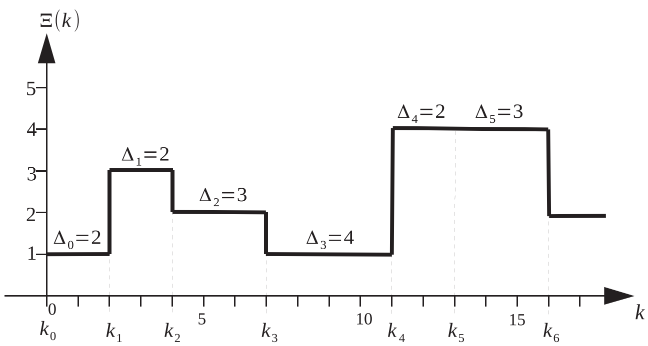

Fig. 3. Discrete time semi-Markov process.

machines are in.[2] We will assume that the network state process $\{\Xi\}$ is a discrete time semi-Markov process [30], [31], to model situations where network state transitions occur randomly, but not necessarily at every discrete time instant k, see Fig. 3. The transition instants between network states are denoted by $\mathbb{K} \triangleq \{k_l\}$, with $k_0 = 0$, and $k_0 < k_1 < k_2 \cdots$ all integers. The *holding times*, or the amounts of time spent in a network state between transitions, are defined as $\Delta_l \triangleq k_{l+1} - k_l$. We will also refer to the period between successive network state transitions as a *holding period*. We assume that the holding times are bounded, thus $\Delta_l \leq \Delta_{\max}, \forall l$. Let $\mathbb{D} \triangleq \{1, 2, \ldots, \Delta_{\max}\}$. We have

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

where, in the second line, we have made use of the fact that the Markov property holds at the transition instants (since the process is semi-Markov [30], [31]), with

$$q_{ij} \triangleq \mathbb{P}\{\Xi(k_{l+1}) = j|\Xi(k_l) = i\} \tag{1}$$

being the transition probabilities of the embedded Markov chain, and the fact that the conditional probabilities of the holding time

$$\psi_i(\delta) \triangleq \mathbb{P}\{\Delta_l = \delta|\Xi(k_l) = i\} \tag{2}$$

[2]In practice, network states $\Xi(k)$ can be estimated by either directly observing the positions of the machinery on the factory floor, or by using techniques to estimate variations in the radio environment [29].

depends only on the current state of the embedded Markov chain.

The *network configuration* $\pi(k)$ at time k fixes the transmission schedule that determines which nodes each sensor will receive from and forward to. The set of all possible network configurations is denoted by $\Pi = \{1, 2, \ldots, |\Pi|\}$, and the set of possible configurations when in network state j by $\Pi_j \subseteq \Pi$. We assume that the set of all possible network configurations has been precomputed and is known at the gateway. For instance, in each network state, one can compute a small number of reasonable configurations, using a few routing algorithms that optimize different objectives [32], which could also take into account possible link failures during operation. The set of all configurations in the different network states would then form our precomputed set of possible network configurations.

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

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

and the corresponding link success probabilities by

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

We will assume that, *conditioned on a network state*, the dropouts $\{\gamma_m\}$ are i.i.d. Bernoulli processes, with $\{\gamma_m\}$ independent of $\{\gamma_n\}$ for $m \neq n$. Note that the packet reception probabilities can differ in different network states. Situations with i.i.d. and Markovian packet drops can also be regarded as special cases of this model, see [22] for details.

C. Kalman Filter at Gateway

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

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

which determines whether the measurement of sensor m at time k is received by the gateway. Due to the fact that the set of active links forms a tree, we have

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

and, by independence,

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

Let $\theta(k) \triangleq \text{col}(\theta_1(k), \ldots, \theta_M(k))$, $y(k) \triangleq \text{col}(\theta_1(k)y_1(k), \ldots, \theta_M(k)y_M(k))$, $R \triangleq \text{diag}(R_1, \ldots, R_M)$, $C(k) \triangleq \text{col}(\theta_1(k)C_1, \ldots, \theta_M(k)C_M)$. The information set available at the gateway at time k is

$$\mathcal{I}(k) = \{\theta(0), \ldots, \theta(k), y(0), \ldots, y(k)\}.$$

The state estimates and estimation error covariances are defined as

$$\hat{x}(k|k-1) \triangleq \mathbb{E}\left\{x(k)|\mathcal{I}(k-1)\right\}$$
$$P(k|k-1) \triangleq \mathbb{E}\Big\{(x(k)-\hat{x}(k|k-1)) \times (x(k)-\hat{x}(k|k-1))^T \Big| \mathcal{I}(k-1)\Big\}.$$

The Kalman filtering equations can then be written as (see, e.g., [15], [22])

$$\hat{x}(k+1|k) = A\hat{x}(k|k-1) + K(k)\left(y(k) - C(k)\hat{x}(k|k-1)\right)$$
$$P(k+1|k) = AP(k|k-1)A^T + Q - K(k)C(k)P(k|k-1)A^T \tag{3}$$

where $K(k) \triangleq AP(k|k-1)C(k)^T(C(k)P(k|k-1)C(k)^T + R)^{-1}$. In the sequel, we will also use the shorthand $P(k) \triangleq P(k|k-1)$.

Remark II.1: An alternative form of the Kalman filter equations, similar to, e.g., [5], can be given as follows. Let $\tilde{C}(k) \triangleq \mathrm{col}(\{C_1,\ldots,C_M|\theta_m(k)=1\})$, $\tilde{y}(k) \triangleq \mathrm{col}(\{y_1(k),\ldots, y_M(k)|\theta_m(k)=1\})$, $\tilde{R}(k) \triangleq \mathrm{diag}(\{R_1,\ldots,R_M|\theta_m(k)=1\})$. Then we have

$$\hat{x}(k+1|k) = A\hat{x}(k|k-1) + \tilde{K}(k)\Big(\tilde{y}(k) - \tilde{C}(k)\hat{x}(k|k-1)\Big)$$
$$P(k+1|k) = AP(k|k-1)A^T + Q - \tilde{K}(k)\tilde{C}(k)P(k|k-1)A^T$$
$$\tilde{K}(k) = AP(k|k-1)\tilde{C}(k)^T \times \Big(\tilde{C}(k)P(k|k-1)\tilde{C}(k)^T + \tilde{R}(k)\Big)^{-1}. \tag{4}$$

III. Optimal Network Reconfiguration

As stated in Section II-A, network states model random changes in the characteristics of the wireless environment. Due to these changes, see, e.g., Fig. 2, 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.

A. Reconfiguration Issues

In what follows, we will use a similar cost of reconfiguration as in [23], 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* $T_l \in \mathbb{N}_0$ at the l-th state transition, where some links will not be available, resulting in poor transitory performance of the Kalman filter (see Section VI for a specific example). Therefore, there is potentially a tradeoff between choosing 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 poorer performance.

The reconfiguration time T_l 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, or even tens of seconds [33]. On the other hand, in WirelessHART which maintains multiple routes that can be switched at different time instances [34], it might be more appropriate to take $T_l = 0$. In this paper, T_l is taken to be random,[3] with a probability distribution that could depend on the current network state $\Xi(k_l)$, the previous network configuration $\pi(k_{l-1})$, and the new network configuration chosen $\pi(k_l)$. We will assume that the reconfiguration times are bounded, i.e., $T_l \le T_{\max}, \forall\, l$.

B. Optimization Problem

At each transition instant $k_l \in \mathbb{K}$, we seek to find a network configuration

$$\pi(k_l) \triangleq \pi\left(P(k_l), \Xi(k_l), \pi(k_{l-1})\right)$$

which is to be held until the next transition instant $k_{l+1} \in \mathbb{K}$, and which minimizes an expected estimation error covariance performance measure over this holding period. The gateway decides on the new configuration based on knowledge of the current error covariance $P(k_l)$, the current network state $\Xi(k_l)$, and the old network configuration $\pi(k_{l-1})$, which is then communicated back to the sensors. For ease of exposition, we introduce the aggregated process

$$\mathcal{U}(k_l) \triangleq \left(P(k_l), \Xi(k_l), \pi(k_{l-1})\right), \quad k_l \in \mathbb{K}. \tag{5}$$

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 optimization

$$\pi^*(k_l) = \underset{\pi(k_l)\in\Pi_j}{\arg\min}\, \mathcal{V}\left(\mathcal{U}(k_l), \pi(k_l)\right) \tag{6}$$

where the cost function

$$\mathcal{V}\left(\mathcal{U}(k_l), \pi(k_l)\right) \triangleq \mathbb{E}\left\{\sum_{d=1}^{\Delta_l} \mathrm{tr}\, P(k_l+d) \,\middle|\, \mathcal{U}(k_l), \pi(k_l)\right\} \tag{7}$$

with Δ_l being random. The quantity $\mathcal{V}(\mathcal{U}(k_l), \pi(k_l))$ amounts to the sum of the trace of expected error covariances over the random holding time Δ_l, when the configuration $\pi(k_l)$ is used. Similar cost functions have been considered in, e.g., [35], [36] in optimizing Kalman filter performance over a finite horizon.

[3] Suppose the new configuration is to be communicated from the gateway back to the sensors (either using a broadcast or transmitted via intermediate nodes). Then, due to random packet losses, information about this new configuration may not get through reliably to all nodes at the same time but will need to be retransmitted, resulting in a random T_l.

In computations, it is useful to further rewrite (7) as

$$\begin{aligned}&\mathcal{V}\left(\mathcal{U}(k_l),\pi(k_l)\right)\\&\quad=\sum_{\delta=1}^{\Delta_{\max}}\left[\sum_{t=0}^{T_{\max}}\mathbb{E}\left\{\sum_{d=1}^{\delta}\operatorname{tr}P(k_l+d)\middle|\mathcal{U}(k_l),\pi(k_l),T_l=t\right\}\right.\\&\qquad\left.\times\mathbb{P}\left\{T_l=t|\Xi(k_l)=j,\pi(k_{l-1}),\pi(k_l)\right\}\right]\\&\qquad\times\mathbb{P}\left\{\Delta_l=\delta|\Xi(k_l)=j\right\}.\end{aligned}\tag{8}$$

In (8), the expectations in the terms

$$\mathbb{E}\left\{P(k_l+d)|\mathcal{U}(k_l),\pi(k_l),T_l=t\right\}\tag{9}$$

are taken over the packet loss processes [which affect the Kalman filter recursions (3)], while the summations over δ and t average over the random holding times and random reconfiguration times respectively. Following the model of Section II-A, the network state $\Xi(k_l)$ determines the distribution of the holding times [see (2)], and thereby the upper limit of the sum over d in (8); differences between the decision variable $\pi(k_l)$ and the previous configuration $\pi(k_{l-1})$ determine which links would be moved to a transient state. In particular, (9) is computed based on whether the network is still in the transient mode (if $d \leq T_l$) or has been fully reconfigured (if $d > T_l$), with the expectation taken over the discrete random variables $\{\theta(k_l),\ldots,\theta(k_l+d-1)\}$.

C. Computational Aspects

In principle, problem (6) can be solved by checking the values of $\mathcal{V}(\mathcal{U}(k_l),\pi(k_l))$ for each of the different configurations $\pi(k_l)\in\Pi_j$. However, computation of the expectations in (9) involves considering the values of $P(k_l+d)$ for all possible combinations of $\{\theta(k_l),\ldots,\theta(k_l+d-1)\}$, with the number of possibilities being $O(2^{Md})$ in general. In particular, computing $\mathbb{E}\{P(k_l+\Delta_{\max})\,|\,\mathcal{U}(k_l),\pi(k_l),T_l\}$ will have a complexity of $O(2^{M\Delta_{\max}})$. Thus, for large holding times, which occur often in industrial settings, calculating the cost function (7) is computationally intensive. Section IV proposes a suboptimal method, which minimizes an alternative cost function that can be computed with complexity $O(2^M)$.

D. Stochastic Stability Analysis

In this subsection, we will present a criterion for estimator stability with network configurations chosen by solving the optimal reconfiguration problem (6), by extending the methods developed in [22]. It is worth noting that establishing stability is non-trivial, even for simple scheduling problems, see, e.g., [37].

Definition 1: The Kalman filter is said to be uniformly bounded if there exists a finite constant $B>0$ such that $\mathbb{E}\{\operatorname{tr}P(k)\}\leq B,\forall\,k\in\mathbb{N}$.

First, we have the following:

Lemma III.1: The process $\{Z\}_{\mathbb{K}}$ defined by

$$Z(k_l)\triangleq\left(P(k_{l-1}+1),\ldots,P(k_l),\Xi(k_l),\pi(k_{l-1})\right),\quad k_l\in\mathbb{K}$$

is Markovian.

Proof: Note that $\{\Xi\}_{\mathbb{K}}$ is Markovian and $\pi(k_l)$ depends only on $(P(k_l),\Xi(k_l),\pi(k_l-1))$. We also have

$$\begin{aligned}&\mathbb{P}\left\{C(k_l)|P(k_l),\ldots,P(k_{l-1}+1),P(k_{l-1}),\ldots,\right.\\&\qquad\left.\Xi(k_l),\Xi(k_{l-1}),\ldots,\pi(k_{l-1}),\pi(k_{l-2}),\ldots\right\}\\&\quad=\mathbb{P}\left\{C(k_l)|P(k_l),\ldots,P(k_{l-1}+1),\Xi(k_l),\pi(k_{l-1})\right\}.\end{aligned}$$

The result then follows from (3). □

Next, define the observability matrices $\mathcal{O}(k,k)=C(k)$,

$$\mathcal{O}(k+n,k)=\begin{bmatrix}C(k)\\C(k+1)A\\\vdots\\C(k+n)A^n\end{bmatrix},\quad n\in\mathbb{N}.\tag{10}$$

Consider the processes $\{\varrho_d\}_{\mathbb{K}}, d=1,\ldots,\Delta_l$, given by

$$\varrho_d(k_l)=\begin{cases}1, & \text{if } \mathcal{O}(k_l+d-1,k_l) \text{ is full rank}\\0, & \text{otherwise.}\end{cases}$$

Taking into account the network state, network configurations, and reconfiguration times, define

$$\begin{aligned}&\mu_d(j,p,p^-)\\&\quad\triangleq\mathbb{P}\left\{\varrho_d(k_l)=0|\Xi(k_l)=j,\pi(k_l)=p,\pi(k_{l-1})=p^-\right\}\\&\quad=\sum_{t=0}^{T_{\max}}\mathbb{P}\left\{\varrho_d(k_l)=0|\Xi(k_l)=j,\pi(k_l)=p,\pi(k_{l-1})=p^-,T_l=t\right\}\\&\qquad\times\mathbb{P}\left\{T_l=t|\Xi(k_l)=j,\pi(k_l)=p,\pi(k_{l-1})=p^-\right\}.\end{aligned}\tag{11}$$

Then we have:

Theorem III.2: Suppose there exists a policy $\pi^\sharp(k_l)\triangleq\pi^\sharp(\Xi(k_l),\pi^\sharp(k_{l-1}))$, dependent only on the current network state $\Xi(k_l)=j$ and existing configuration $\pi^\sharp(k_{l-1})=p^-$, such that

$$\sum_{\delta=1}^{\Delta_{\max}}\mu_\delta\left(j,\pi^\sharp(j,p^-),p^-\right)\|A\|^{2\delta}\psi_j(\delta)<1,\ \forall\,j\in\mathbb{B},\ \forall\,p^-\in\Pi\tag{12}$$

where $\|A\|$ denotes the spectral norm of A and $\psi_j(\delta)$ is as defined in (2). Then, under the optimal network reconfiguration method (6), the Kalman filter is uniformly bounded.

Proof: See Appendix A. □

Theorem III.2 establishes a sufficient condition on estimator stability, see Section VI for an example of how this condition can be verified numerically. Intuitively, condition (12) averages out non-full rank observation outcomes over the random holding times $\Delta_l=\delta$.

Remark III.3: In the case of a single network state with i.i.d. packet drops, we have $\delta=1$, and $\psi_j(\delta)=1,\forall\,j$. Then $\mu_\delta(j,p,p^-)$ reduces to the probability that $C(k)$ is not full rank, and (12) becomes

$$\mathbb{P}\left\{C(k)\text{ is not full rank}\right\}\|A\|^2<1$$

which is similar to the stability condition of [16]. Further reducing to a single sensor with C_1 full rank, the probability of $C(k)$ not being full rank is the probability of dropping a packet, so (12) becomes

$$\mathbb{P}\{\gamma_1(k)=0\}\|A\|^2<1$$

which resembles the stability conditions of, for example, [2], [38].

Remark III.4: Theorem III.2 differs from [22, Theorem 2] in that the probabilities $\mu_d(j,p,p^-)$ also depends on the network configurations $\pi(k_{l-1})$ and $\pi(k_l)$, a concept which was not considered in [22]. In addition, $\mu_d(j,p,p^-)$ is defined to be a probability conditional on $\Xi(k_l)$ rather than $\Xi(k_{l-1})$, which is perhaps more natural since our chosen configurations depend on $\Xi(k_l)$ rather than $\Xi(k_{l-1})$.

E. Multiple Holding Periods

In Section III-B network reconfigurations are carried out by considering the sum of expected error covariances over one network state holding period (involving several time steps k). By looking further ahead over multiple holding periods, one can possibly achieve better performance. For the case of averaging over N holding periods, the new configuration $\pi(k_l)\in\Pi_j$ when $\Xi(k_l)=j$ is found via the following optimization:

$$\begin{aligned}\arg\min_{\pi(k_l)\in\Pi_j}\Bigg[&\mathbb{E}\left\{\sum_{d_0=1}^{\Delta_l}\operatorname{tr}P(k_l+d_0)\middle|\mathcal{U}(k_l),\pi(k_l)\right\}\\&+\min_{\pi(k_{l+1})}\mathbb{E}\left\{\mathbb{E}\left\{\sum_{d_1=1}^{\Delta_{l+1}}\operatorname{tr}P(k_{l+1}+d_1)\middle|\mathcal{U}(k_{l+1}),\pi(k_{l+1})\right\}\middle|\mathcal{U}(k_l),\pi(k_l)\right\}+\cdots\\&+\min_{\pi(k_{l+N-1})}\mathbb{E}\left\{\mathbb{E}\left\{\sum_{d_{N-1}=1}^{\Delta_{l+N-1}}\operatorname{tr}P(k_{l+N-1}+d_{N-1})\middle|\mathcal{U}(k_{l+N-1}),\pi(k_{l+N-1})\right\}\middle|\mathcal{U}(k_l),\pi(k_l)\right\}\Bigg].\end{aligned}\tag{13}$$

We observe that in solving the multiple holding period optimal reconfiguration problem (13), we actually also obtain reconfiguration policies for $\pi(k_{l+1}),\ldots,\pi(k_{l+N-1})$. However, here we will adopt a moving horizon approach similar to [35], so that the optimal $\pi^*(k_{l+1})$ will be obtained by solving problem (13) at the next transition instant $k_{l+1}\in\mathbb{K}$, the optimal $\pi^*(k_{l+2})$ is obtained by solving problem (13) at the transition instant k_{l+2}, and so on. We note that optimization over N holding periods will require the computation of cost functions with an increased complexity of $O(2^{M\Delta_{\max}N})$.

IV. Suboptimal Network Reconfiguration

To address the computational issues outlined in Section III-C, in this section we study a suboptimal scheme which minimizes upper bounds to the expected error covariances, where these upper bounds can be computed recursively with lower complexity than the expected error covariance (7).

A. Optimization Problem

We adopt a suboptimal approach wherein, using $\mathcal{U}(k_l)$ defined as in (5), the new configuration $\pi^*(k_l)\in\Pi_j$ is obtained via

$$\pi^*(k_l)=\arg\min_{\pi(k_l)\in\Pi_j}\mathcal{W}(\mathcal{U}(k_l),\pi(k_l))\tag{14}$$

where

$$\mathcal{W}(\mathcal{U}(k_l),\pi(k_l))\triangleq\sum_{\delta=1}^{\Delta_{\max}}\sum_{d=1}^{\delta}\operatorname{tr}Y(k_l+d)\mathbb{P}\{\Delta_l=\delta|\Xi(k_l)=j\}.\tag{15}$$

The sequence $\{Y(k_l+1),Y(k_l+2),\ldots,Y(k_l+\Delta_{\max})\}$ is given by the following recursion:

$$\begin{aligned}Y(k+1)&=AY(k)A^T+Q\\&\quad-\mathbb{E}\left\{AY(k)C(k)^T\left(C(k)Y(k)C(k)^T+R\right)^{-1}\times C(k)Y(k)A^T\middle|\mathcal{U}(k_l),\pi(k_l)\right\}\\&=AY(k)A^T+Q\\&\quad-\sum_{t=0}^{T_{\max}}\mathbb{E}\left\{AY(k)C(k)^T\left(C(k)Y(k)C(k)^T+R\right)^{-1}\times C(k)Y(k)A^T\middle|\mathcal{U}(k_l),\pi(k_l),T_l=t\right\}\\&\quad\times\mathbb{P}\{T_l=t|\Xi(k_l)=j,\pi(k_l),\pi(k_{l-1})\}\end{aligned}\tag{16}$$

with initial condition $Y(k_l)=P(k_l)$. The expectations

$$\mathbb{E}\left\{AY(k)C(k)^T\left(C(k)Y(k)C(k)^T+R\right)^{-1}C(k)Y(k)A^T\middle|\mathcal{U}(k_l),\pi(k_l),T_l=t\right\},\quad k\in\{k_l,\ldots,k_l+\Delta_{\max}-1\}$$

in (16) are computed with respect to the random packet loss processes, taking into account whether the network is still in the transient mode ($k-k_l\le T_l$) or has been fully reconfigured ($k-k_l>T_l$), similar to the computation of (9). We have the following result:

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

Proof: Define

$$g_k(X)=AXA^T+Q-\mathbb{E}\left\{AXC(k)^T\left(C(k)XC(k)^T+R\right)^{-1}\times C(k)XA^T\middle|\mathcal{U}(k_l),\pi(k_l)\right\}.$$

Lemma IV.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 [2], [5], [39]. The details are omitted for brevity. $\square$

Thus, when the suboptimal method minimizes (15), what is minimized is not the expected error covariance performance measure (7), but by Lemma IV.1, an upper bound to (7).

B. Computational Aspects

Upper bounding sequences of the form (16) are much easier to compute than the expected error covariance when the holding times are large, since one now needs to consider $O(2^M)$ combinations of packet drops at each stage in (16), rather than $O(2^{M\Delta_{\max}})$ when computing the expected error covariance.[4] Furthermore, the bounds often seem to be quite tight, see, e.g., [18].[5] In Section VII we will see that in numerical simulations the configurations obtained using the suboptimal method are in many cases identical to the configurations obtained using the optimal method.

C. Stochastic Stability Analysis

We now give a stability condition for the suboptimal network reconfiguration method. First, we have

Lemma IV.2: The process $\{\bar{Z}\}_{\mathbb{K}}$ defined by

$$\bar{Z}(k_l) \triangleq (Y(k_{l-1}+1), \dots, Y(k_l), \Xi(k_l), \pi(k_{l-1})), \quad k_l \in \mathbb{K}$$

is Markovian.

Proof: The proof follows from the fact that 1) $\{Y\}_{\mathbb{N}}$ is Markovian since $Y(k+1)$ depends only on $Y(k)$, 2) $\{\Xi\}_{\mathbb{K}}$ is Markovian, and 3) $\pi(k_l)$ depends only on $(Y(k_l), \Xi(k_l), \pi(k_l-1))$. $\square$

Now consider a process $\{s(k)\}$ defined by

$$s(k) = \begin{cases} 1, & \text{if } C(k) \text{ is full rank} \\ 0, & \text{otherwise.} \end{cases}$$

For $d = 1, \dots, \Delta_l$, let

$$\begin{aligned} &\nu_d(j, p, p^-) \\ &\quad \triangleq \mathbb{P}\big\{s(k_l+d-1)=0 \big| \, \Xi(k_l)=j, \pi(k_l)=p, \pi(k_{l-1})=p^-\big\} \\ &\quad = \sum_{t=0}^{T_{\max}} \mathbb{P}\big\{s(k_l+d-1)=0 | \Xi(k_l)=j, \pi(k_l)=p, \pi(k_{l-1})=p^-, \\ &\qquad\qquad T_l=t\big\}\mathbb{P}\big\{T_l=t|\Xi(k_l)=j, \pi(k_l)=p, \pi(k_{l-1})=p^-\big\}. \end{aligned}$$

We have:

Theorem IV.3: Suppose there exists a policy $\pi^\sharp(k_l) \triangleq \pi^\sharp(\Xi(k_l), \pi^\sharp(k_{l-1}))$, dependent only on $\Xi(k_l)=j$ and $\pi^\sharp(k_{l-1}) = p^-$, such that

$$\sum_{\delta=1}^{\Delta_{\max}} \nu_\delta \left(j, \pi^\sharp(j,p^-), p^-\right) \|A\|^{2\delta} \psi_j(\delta) < 1, \ \forall\, j \in \mathbb{B}, \ \forall\, p^- \in \Pi. \tag{17}$$

Then, under the suboptimal reconfiguration method (14), the Kalman filter is uniformly bounded.

Proof: See Appendix B. $\square$

Remark IV.4: Comparing Theorems III.2 and IV.3, we see that the condition (17) in Theorem IV.3 involves probabilities of the matrices $C(k)$ not being full rank, which in general is larger than the probability of the observability matrices in (10) not being full rank. Thus, condition (17) in Theorem IV.3 is more stringent than condition (12) of Theorem III.2.

D. Multiple Holding Periods

Similar to Section III-E, for the case of averaging over N holding periods, the new configuration $\pi^*(k_l) \in \Pi_j$ when $\Xi(k_l) = j$ is found via the following optimization:

$$\begin{aligned} \arg\min_{\pi(k_l)\in\Pi_j} \mathbb{E} \Bigg\{ &\sum_{d_0=1}^{\Delta_l} \operatorname{tr} Y_0(k_l+d_0) + \min_{\pi(k_{l+1})} \sum_{d_1=1}^{\Delta_{l+1}} \operatorname{tr} Y_1(k_{l+1}+d_1) \\ &+ \dots + \min_{\pi(k_{l+N-1})} \sum_{d_{N-1}=1}^{\Delta_{l+N-1}} \operatorname{tr} Y_{N-1}(k_{l+N-1}+d_{N-1}) \Bigg\}. \end{aligned} \tag{18}$$

The N sequences $\{Y_0(k_l+1), \dots, Y_0(k_l+\Delta_{\max})\}, \dots, \{Y_{N-1}(k_{l+N-1}+1), \dots, Y_{N-1}(k_{l+N-1}+\Delta_{\max})\}$ in (18) are defined, for $n = 0, \dots, N-1$, as follows:

$$\begin{aligned} Y_n(k+1) = {} & AY_n(k)A^T + Q \\ & - \sum_{t_n=0}^{T_{\max}} \mathbb{E}\Big\{AY_n(k)C(k)^T\big(C(k)Y_n(k)C(k)^T + R\big)^{-1} \\ & \quad \times C(k)Y_n(k)A^T \Big| \bar{\mathcal{U}}(k_{l+n}), \pi(k_{l+n}), T_{l+n}=t_n\Big\} \\ & \quad \times \mathbb{P}\{T_{l+n}=t_n|\Xi(k_{l+n}), \pi(k_{l+n}), \pi(k_{l+n-1})\} \end{aligned} \tag{19}$$

for $k \in \{k_{l+n}, \dots, k_{l+n}+\Delta_{\max}-1\}$, with initial condition $Y_n(k_{l+n}) = Y_{n-1}(k_{l+n-1}+\Delta_{l+n-1}) = Y_{n-1}(k_{l+n})$. In (19), we have $\bar{\mathcal{U}}(k_l) \triangleq (P(k_l), \Xi(k_l), \pi(k_{l-1}))$, and $\bar{\mathcal{U}}(k_{l+n}) \triangleq (Y_n(k_{l+n}), \Xi(k_{l+n}), \pi(k_{l+n-1}))$ for $n > 0$. Note that in the suboptimal reconfiguration problem (18), the minimization over $\pi(k_{l+n})$ for $n > 0$ is computed based on $\bar{\mathcal{U}}(k_{l+n})$, rather than $\mathcal{U}(k_{l+n}) = (P(k_{l+n}), \Xi(k_{l+n}), \pi(k_{l+n-1}))$ as in the optimal method (13).

When looking over N holding periods, computation of the cost functions has a complexity of $O(2^{MN})$, which could be very intensive for large values of N. However, from numerical simulations, it appears that in many situations even the case $N=1$ already provides most of the gains achieved by solving the N-period problem, see Section VII.

V. Other Low Complexity Reconfiguration Schemes

The suboptimal scheme of Section IV requires minimizing a cost function that has complexity $O(2^M)$ to compute. In

[4] While still exponential in the number of sensors, for industrial settings with small subnetworks this is quite feasible.

[5] Some tighter but more complicated bounds based on techniques in [40] can also be used.

this section we briefly describe some schemes with even lower complexity (though poorer performance), which will be used as a performance comparison in Section VII. A more thorough analysis on the scalability of these schemes, and whether they can be modified to give better performance, will be the subject of future work.

A. Network Reconfiguration by Maximizing Packet Reception Probabilities

This network reconfiguration method maximizes a measure of the probability of receiving all the sensor measurements, for a given network state.[6] This maximization will depend on the packet reception probabilities, but doesn't use information about the error covariance, observation matrices, measurement noise or dynamics of the system.

In this scheme, the new configuration $\pi^*(k_l) \in \Pi_j$ is obtained by solving the problem

$$\begin{aligned} \arg\max_{\pi(k_l)\in\Pi_j} \sum_{\delta=1}^{\Delta_{\max}} \sum_{t=0}^{T_{\max}} \sum_{d=1}^{\delta} \mathbb{E}\left\{ \prod_{m=1}^{M} \theta_m(k_l+d-1) \middle| \Xi(k_l)=j, \right. \\ \left. \pi(k_{l-1})=p^-, \pi(k_l)=p, T_l=t \right\} \\ \times \mathbb{P}\left\{T_l=t|\Xi(k_l)=j, \pi(k_{l-1})=p^-, \pi(k_l)=p\right\} \\ \times \mathbb{P}\left\{\Delta_l=\delta_0|\Xi(k_l)=j\right\}. \end{aligned} \tag{20}$$

Note that $\mathbb{E}\{\theta_1(k)\times\cdots\times\theta_M(k)\}$ gives the probability of receiving all M sensor measurements at time k. Thus problem (20) maximizes an average of the probability of receiving all sensor measurements over a single holding period (of random length Δ_l). Since the active links have a tree structure, all M sensor measurements at time k will be received if transmission along all M links $\mathcal{E}_m, m=1,\ldots,M$, are successful at time k. For the case of reconfiguration time $T_l=0$, we then have

$$\mathbb{E}\left\{ \prod_{m=1}^{M} \theta_m(k_l+d-1) \middle| j, p^-, p, t \right\} = \prod_{m=1}^{M} \phi_{m|(j,p)}$$

and in the case of $T_l>0$, we have

$$\begin{aligned} \mathbb{E}\left\{ \prod_{m=1}^{M} \theta_m(k_l+d-1) \middle| j, p^-, p, t \right\} \\ = \begin{cases} 0, & \text{if } d \le T_l \text{ and at least one link} \\ & \text{needs to be changed} \\ \prod_{m=1}^{M} \phi_{m|(j,p)}, & \text{otherwise.} \end{cases} \end{aligned} \tag{21}$$

[6]Receiving all the sensor measurements has similarities with the *convergecast* operation in networking, where data from multiple sources is delivered to a single destination, see, e.g., [41], [42].

In the optimization problems considered in Sections III and IV, the main computational effort is in calculating the expected error covariances (or upper bounds) which form the cost function. However, when maximizing the probability of receiving all sensor measurements, we have the closed form expression (21), which means that problem (20) can be solved very efficiently.

B. Network Reconfiguration by Optimizing Steady State Values of Upper Bounds

This scheme is a "steady state" version of the suboptimal method which minimizes the steady state value of the upper bounds $\{Y(k)\}$, where the steady value Y_s for given $\mathcal{U}(k_l)$ and $\pi(k_l)$ satisfies

$$\begin{aligned} Y_s = AY_sA^T + Q - \mathbb{E}\left\{ AY_sC(k)^T\big(C(k)Y_sC(k)^T+R\big)^{-1} \right. \\ \left. \times C(k)Y_sA^T \middle| \mathcal{U}(k_l), \pi(k_l) \right\} \end{aligned}$$

and be computed by, e.g., iterating the recursion (16) until convergence. One can pre-compute and store these steady state values for different combinations of network state and network configuration, so that in operation one can simply use a lookup table to compare the cost functions for different configurations.

Since this method assumes a steady state, information about the reconfiguration time and current error covariance $P(k_l)$ ends up being not utilized, however from simulations it performs quite well if the holding times are long, see Section VII.

C. Network Reconfiguration Using Monte Carlo Simulation of Cost Functions

In this scheme, at each transition time instant, rather than computing the cost function

$$\mathbb{E}\left\{ \sum_{d=1}^{\Delta_l} \operatorname{tr} P(k_l+d) \middle| \mathcal{U}(k_l), \pi(k_l) \right\}$$

for different configurations analytically which has high complexity, the cost functions instead are approximated by simulating many different realizations of the packet drops

$$\gamma_1(k_l),\ldots,\gamma_1(k_l+\Delta_l-1),\ldots,\gamma_M(k_l),\ldots,\gamma_M(k_l+\Delta_l-1),$$

random holding times Δ_l, and random reconfiguration times T_l. For each realization, we compute

$$\sum_{d=1}^{\Delta_l} \operatorname{tr} P(k_l+d)$$

and then take the average over these realizations.

This scheme may be attractive for larger networks in that it is not exponential in the number of sensors M when compared to the suboptimal method, since for additional sensors one merely simulates additional packet drop realizations for these new links.

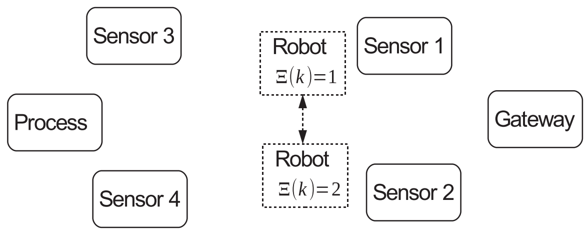

Fig. 4. Sensor network for example of Section VI.

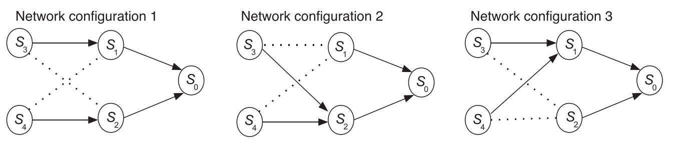

Fig. 5. Network configurations for example of Section VI.

VI. An Illustrative Example

Here, we give an example to illustrate some of the concepts introduced in the paper, in particular how to verify the stability condition (12) of Theorem III.2. We will consider an example with four sensor nodes, see Fig. 4 for a diagram of the physical layout. The system has parameters

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

$C_1 = C_2 = C_3 = C_4 = [1\ 1]$, $R_1 = R_2 = 20$, $R_3 = R_4 = 0.2$. The differences in the sensor measurement noise variances correspond to situations where either the process is located much closer to sensors 3 and 4 than to sensors 1 and 2, or if sensors 1 and 2 are located in a more hostile radio environment than sensors 3 and 4 [29]. However, sensors 1 and 2 have better connectivity to the gateway.

The set of all network configurations is shown in Fig. 5. 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 packet reception probabilities for the links in each of the network configurations are

$$\begin{aligned} &\phi_{1|(1,1)} = 0.5, \phi_{2|(1,1)} = 0.5, \phi_{3|(1,1)} = 0.1, \phi_{4|(1,1)} = 0.5 \\ &\phi_{1|(1,2)} = 0.5, \phi_{2|(1,2)} = 0.5, \phi_{3|(1,2)} = 0.8, \phi_{4|(1,2)} = 0.5 \\ &\phi_{1|(2,1)} = 0.5, \phi_{2|(2,1)} = 0.5, \phi_{3|(2,1)} = 0.5, \phi_{4|(2,1)} = 0.1 \\ &\phi_{1|(2,3)} = 0.5, \phi_{2|(2,3)} = 0.5, \phi_{3|(2,3)} = 0.5, \phi_{4|(2,3)} = 0.8. \end{aligned} \tag{22}$$

Network state 1 corresponds to the case where there is a robot blocking the line of sight between sensor nodes 1 and 3, giving a packet reception probability of 0.1 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 blocking the line of sight between sensors 2 and 4.

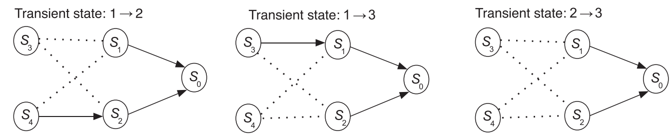

Fig. 6. Transient states when reconfiguring between two network configurations.

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

$$\begin{aligned} \mathbb{P}\{\Xi(k_{l+1}) = 1 | \Xi(k_l) = 1\} &= q_{11} = 0.5, \quad q_{12} = 0.5 \\ \mathbb{P}\{\Xi(k_{l+1}) = 1 | \Xi(k_l) = 2\} &= q_{21} = 0.5, \quad q_{22} = 0.5. \end{aligned}$$

The reconfiguration times have the distribution

$$\begin{aligned} \mathbb{P}\{T_l = 1 | \Xi(k_l), \pi(k_l), \pi(k_{l-1})\} &= 0.8 \\ \mathbb{P}\{T_l = 2 | \Xi(k_l), \pi(k_l), \pi(k_{l-1})\} &= 0.2 \end{aligned} \tag{23}$$

$\forall\,(\Xi(k_l), \pi(k_l), \pi(k_{l-1}))$. The transient states in reconfiguring between different network configurations are shown in Fig. 6. For instance, in reconfiguring from network configuration 2 to configuration 3, the active 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 where sensors 3 and 4 do not have connectivity to the rest of the network for some time T_l. Similarly, reconfiguring from configuration 3 to configuration 2 will also lead to the same transient state.

We now illustrate how to verify the stability condition (12). We need to compute the terms $\mu_d(j, p, p^-)$, which, using (11), requires us to compute the probabilities

$$\mathbb{P}\left\{\varrho_d(k_l) = 0 | \Xi(k_l) = j, \pi(k_l) = p, \pi(k_{l-1}) = p^-, T_l = t\right\}. \tag{24}$$

The observability matrices $\mathcal{O}(k_l + d - 1, k_l)$ are as in (10), where each $C(k) = \mathrm{col}(\theta_1(k)C_1, \ldots, \theta_M(k)C_M)$, $k = k_l, k_l + 1, \ldots, k_l + d - 1$. One can easily verify that if $\theta_{m_1}(k_1) = 1$ and $\theta_{m_2}(k_2) = 1$ for any $m_1, m_2 \in \{1, \ldots, M\}$, and any $k_1, k_2 \in \{k_l, k_l + 1, \ldots, k_l + d - 1\}$ with $k_1 \neq k_2$, then $\mathcal{O}(k_l + d - 1, k_l)$ has full rank. Thus, $\mathcal{O}(k_l + d - 1, k_l)$ is not full rank when either:

1) $\theta_m(k) = 0, \forall\, m \in \{1, \ldots, M\}$ and $\forall\, k \in \{k_l, k_l + 1, \ldots, k_l + d - 1\}$, or
2) there exists a $k^* \in \{k_l, k_l + 1, \ldots, k_l + d - 1\}$ such that $\sum_{m=1}^{M} \theta_m(k^*) \geq 1$ and $\theta_m(k) = 0, \forall\, m \in \{1, \ldots, M\}$ and $k \neq k^*$.

First, consider the instance $d = 4$, $\Xi(k_l) = 2$, $\pi(k_{l-1}) = 2$, $\pi(k_l) = 3$, $T_l = 1$. With these parameters, the network will be in the transient state $(2 \to 3)$ of Fig. 6 at time k_l, and be in network configuration 3 at times $k_l + 1, k_l + 2, k_l + 3$. Note that $\theta_m(k) = 0, \forall\, m$ when $\gamma_1(k) = 0$ and $\gamma_2(k) = 0$, both in the transient state and in network configuration 3. For case 1) above, note that for fixed k, the situation that

$\theta_m(k)=0, \forall m$ occurs with probability $(1-\phi_{1|(2,3)})(1-\phi_{2|(2,3)})$. Thus case 1) occurs with probability

$$\left[\left(1-\phi_{1|(2,3)}\right)\left(1-\phi_{2|(2,3)}\right)\right]^4.$$

For case 2) above, consider individually the four situations when $k^* = k_l, k_l+1, k_l+2, k_l+3$. One can easily verify that each of these four situations occurs with probability

$$\left[1-\left(1-\phi_{1|(2,3)}\right)\left(1-\phi_{2|(2,3)}\right)\right]\left[\left(1-\phi_{1|(2,3)}\right)\left(1-\phi_{2|(2,3)}\right)\right]^3$$

and so case 2) occurs with probability

$$4\left[1-\left(1-\phi_{1|(2,3)}\right)\left(1-\phi_{2|(2,3)}\right)\right]\left[\left(1-\phi_{1|(2,3)}\right)\left(1-\phi_{2|(2,3)}\right)\right]^3.$$

Hence

$$\begin{aligned}&\mathbb{P}\left\{\varrho_4(k_l)=0|\Xi(k_l)=2,\pi(k_l)=3,\pi(k_{l-1})=2,T_l=1\right\}\\&=\left[\left(1-\phi_{1|(2,3)}\right)\left(1-\phi_{2|(2,3)}\right)\right]^4+4\left[1-\left(1-\phi_{1|(2,3)}\right)\left(1-\phi_{2|(2,3)}\right)\right]\\&\quad\times\left[\left(1-\phi_{1|(2,3)}\right)\left(1-\phi_{2|(2,3)}\right)\right]^3.\end{aligned}$$

Following the same arguments, it is not difficult to show that for other values of d, $\Xi(k_l)=j$, $\pi(k_{l-1})=p^-$, $\pi(k_l)=p$, $T_l=t$, case 1) occurs with probability

$$\left[\left(1-\phi_{1|(j,p)}\right)\left(1-\phi_{2|(j,p)}\right)\right]^d,$$

case 2) occurs with probability

$$d\left[1-\left(1-\phi_{1|(j,p)}\right)\left(1-\phi_{2|(j,p)}\right)\right]\left[\left(1-\phi_{1|(j,p)}\right)\left(1-\phi_{2|(j,p)}\right)\right]^{d-1},$$

and hence

$$\begin{aligned}\mu_d(j,p,p^-)&=\left[\left(1-\phi_{1|(j,p)}\right)\left(1-\phi_{2|(j,p)}\right)\right]^d\\&\quad+d\left[1-\left(1-\phi_{1|(j,p)}\right)\left(1-\phi_{2|(j,p)}\right)\right]\left[\left(1-\phi_{1|(j,p)}\right)\left(1-\phi_{2|(j,p)}\right)\right]^{d-1}.\end{aligned}\tag{25}$$

Let the network state holding times have the following distribution:

$$\begin{aligned}&\mathbb{P}\left\{\Delta_l=1|\Xi(k_l)\right\}=0.1,\ \mathbb{P}\left\{\Delta_l=2|\Xi(k_l)\right\}=0.1\\&\mathbb{P}\left\{\Delta_l=3|\Xi(k_l)\right\}=0.1,\mathbb{P}\left\{\Delta_l=4|\Xi(k_l)\right\}=0.7,\ \forall\Xi(k_l).\end{aligned}\tag{26}$$

Suppose we try the policy $\pi^\sharp$ which uses network configuration 1 at all times. Then using (25) and (26), we find that for $j \in \{1,2\}$

$$\begin{aligned}&\sum_{\delta=1}^{\Delta_{\max}}\mu_\delta\left(j,\pi^\sharp(j,p^-),p^-\right)\|A\|^{2\delta}\psi_j(\delta)\\&\qquad=\sum_{\delta=1}^{\Delta_{\max}}\mu_\delta(j,1,1)\|A\|^{2\delta}\psi_j(\delta)=0.4342<1.\end{aligned}$$

Since we can find at least one policy satisfying condition (12) of Theorem III.2, the Kalman filter will be uniformly bounded.

TABLE I
COMPARISON BETWEEN OPTIMAL AND SUBOPTIMAL RECONFIGURATION SCHEMES

$\phi_{3\|(1,1)}$ $=\phi_{4\|(2,1)}$	$\mathbb{E}[\mathrm{tr}P(k)]$ Optimal	$\mathbb{E}[\mathrm{tr}P(k)]$ Suboptimal	Differences in configurations b/w optimal and suboptimal
0.1	1.650	1.650	0
0.2	1.650	1.650	0
0.3	1.644	1.649	27
0.4	1.574	1.574	0
0.5	1.442	1.442	0
0.6	1.329	1.329	0
0.7	1.239	1.239	0
0.8	1.162	1.162	0
0.9	1.098	1.098	0

VII. NUMERICAL STUDIES

A. Comparison Between Optimal and Suboptimal Reconfiguration Schemes

We will use the same example as Section VI, with the holding time distribution (26). The maximum holding time $\Delta_{\max}=4$ here is chosen to be small in order to allow for a comparison between the optimal and suboptimal reconfiguration methods of Sections III and IV.

We first simulated a single realization of time length 10 000. The trace of the time averaged error covariance, $\mathbb{E}[\mathrm{tr}\,P(k)]$, when performing network reconfiguration is 1.65, whereas $\mathbb{E}[\mathrm{tr}\,P(k)]$ with no reconfiguration is 2.14, which amounts to a performance gain of about 30%. The network configurations obtained using both optimal and suboptimal methods behaved identically: Whenever the network state was equal to 1, the network changed to network configuration 2, while if the network state was equal to 2, the network changed to network configuration 3. However, different behaviour can be observed by modifying the packet reception probabilities. For instance, if in (22), both $\phi_{3|(1,1)}$ and $\phi_{4|(2,1)}$ are increased (so that the probability of packet reception in these two links for network configuration 1 is increased), then the network becomes less likely to reconfigure. From simulations, we found that for values of $\phi_{3|(1,1)}$ and $\phi_{4|(2,1)}$ greater than around 0.4, the network is always in network configuration 1, i.e., the network never reconfigures.

In Table I, we give the values of $\mathbb{E}[\mathrm{tr}\,P(k)]$ under the optimal and suboptimal methods, for different values of $\phi_{3|(1,1)}$ and $\phi_{4|(2,1)}$, with $\phi_{3|(1,1)}=\phi_{4|(2,1)}$. Each $\mathbb{E}[\mathrm{tr}\,P(k)]$ entry is computed by taking the time average of Monte Carlo realizations of length 10 000. We also list the number of times when the optimal and suboptimal methods gave different network configurations. Only when $\phi_{3|(1,1)}$ and $\phi_{4|(2,1)}$ are around 0.3 did we observe significant differences (27 times in a realization of length 10 000) in the configurations obtained using the optimal and suboptimal methods, with the resulting performance being very similar. In terms of computational complexity, here $M=4$, $\Delta_{\max}=4$, and $T_{\max}=2$. To compute the cost function for the optimal method requires consideration of approximately $(2^M+2^{2M}+\cdots+2^{\Delta_{\max}M})\times T_{\max}=(2^4+2^8+2^{12}+2^{16})\times 2=139\,808$ different terms. On the other hand, computing the cost function for the suboptimal method requires consideration of approximately

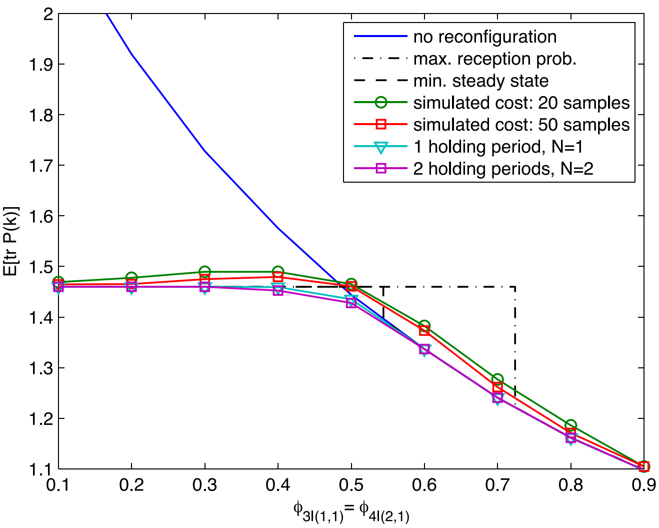

Fig. 7. $\mathbb{E}[\mathrm{tr}\, P(k)]$ for suboptimal network reconfiguration over one and two holding periods and low complexity schemes.

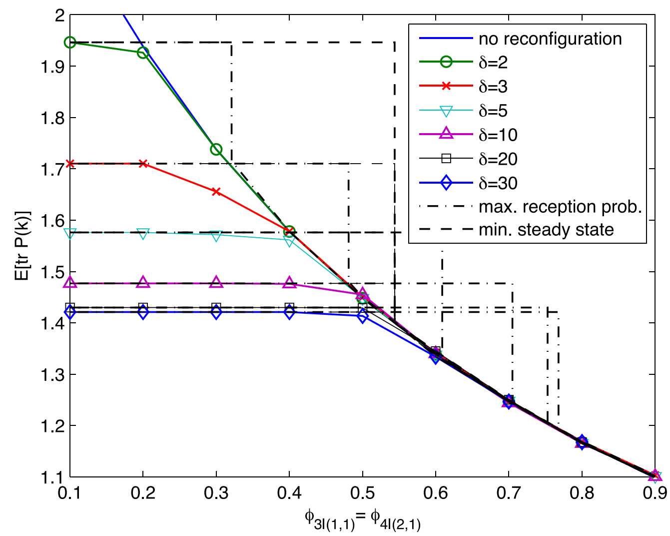

Fig. 8. $\mathbb{E}[\mathrm{tr}\, P(k)]$ for different holding times.

$2^M \times T_{\max} \times \Delta_{\max} = 2^4 \times 2 \times 4 = 128$ different terms, substantially less than for the optimal method.

B. Comparison With Low Complexity Schemes and Optimization Over $N = 2$ Holding Periods

We now consider the case where the network state holding times have the following distribution:

$$\mathbb{P}\{\Delta_l = 11|\Xi(k_l)\} = \mathbb{P}\{\Delta_l = 12|\Xi(k_l)\} = \mathbb{P}\{\Delta_l = 13|\Xi(k_l)\} = \mathbb{P}\{\Delta_l = 14|\Xi(k_l)\} = \frac{1}{4}, \ \forall \Xi(k_l)$$

so that the minimum duration of a holding period is at least 11. Longer holding times are typically encountered in industrial environments, see, e.g., Fig. 2. Due to the substantial increase in the computational complexity of solving the optimal reconfiguration problem for long holding times and/or the case of two holding periods, here we will only present results for the suboptimal methods of Section IV, and the low complexity schemes of Section V.

In Fig. 7, we plot $\mathbb{E}[\mathrm{tr}\, P(k)]$ for solving the suboptimal network reconfiguration problem over $N = 1$ or $N = 2$ holding periods, together with the case of no reconfiguration and the low complexity schemes of Section V, for different values of $\phi_{3|(1,1)}$ and $\phi_{4|(2,1)}$, with $\phi_{3|(1,1)} = \phi_{4|(2,1)}$, where $\mathbb{E}[\mathrm{tr}\, P(k)]$ for each point on the graphs is obtained by taking the time average of Monte Carlo realizations of length 10 000. For small values of $\phi_{3|(1,1)}$ and $\phi_{4|(2,1)}$, the performance gains from reconfiguration are larger than in Section VII-A, due to the longer periods of time in which one can use a good network configuration before needing to reconfigure. For instance, when $\phi_{3|(1,1)} = \phi_{4|(2,1)} = 0.1$, $\mathbb{E}[\mathrm{tr}\, P(k)]$ is 1.46 with reconfigurations and 2.14 without reconfigurations, resulting in a performance gain of 47%, compared to 30% for the case examined in Section VII-A. We also see that the results are very similar when optimizing over both one or two holding periods. In fact, in our simulation results, only for values of $\phi_{3|(1,1)}$ and $\phi_{4|(2,1)}$ around 0.4–0.5 did we observe differences in the network configurations obtained, with the resulting performance differences being very small.

The scheme of Section V-A that maximizes the probability of receiving all packets has good performance for small or large values of $\phi_{3|(1,1)}$ and $\phi_{4|(2,1)}$, but performs poorly otherwise. There is a threshold behavior, due to the network configurations being determined based only on the packet reception probabilities, and not the error covariances or system parameters: For values of $\phi_{3|(1,1)}$ and $\phi_{4|(2,1)}$ below a certain threshold, the scheme always changes configuration when the network state changes, while above this threshold the network never reconfigures. The scheme of Section V-B that minimizes the steady state value also exhibits threshold behavior, but has good performance over most of the range, due to the relatively long holding times (minimum of 11) in this example. For the scheme of Section V-C that approximates the cost functions by simulation, the performance trend follows that of the suboptimal schemes, though with a noticeable gap in performance, which can be reduced by increasing the number of samples used. For this example, using 50 samples to approximate each cost function gave a similar running time to the suboptimal method with $N = 1$.

C. Performance Gains With Different Holding Times

We now consider the case where the network state holding times have the following distribution:

$$\mathbb{P}\{\Delta_l = \delta|\Xi(k_l) = 1\} = \mathbb{P}\{\Delta_l = \delta|\Xi(k_l) = 2\} = 1 \quad (27)$$

for different values of δ. Fig. 8 depicts $\mathbb{E}[\mathrm{tr}\, P(k)]$ in solving the suboptimal network reconfiguration problem over one holding period, for holding times of duration $\delta = 2, 3, 5, 10, 20,$ and 30, together with the case of no reconfiguration, and the low complexity schemes of Section V-A and B. We see that for larger δ, there is a greater performance gain by performing network reconfiguration. Additionally, there is a wider range

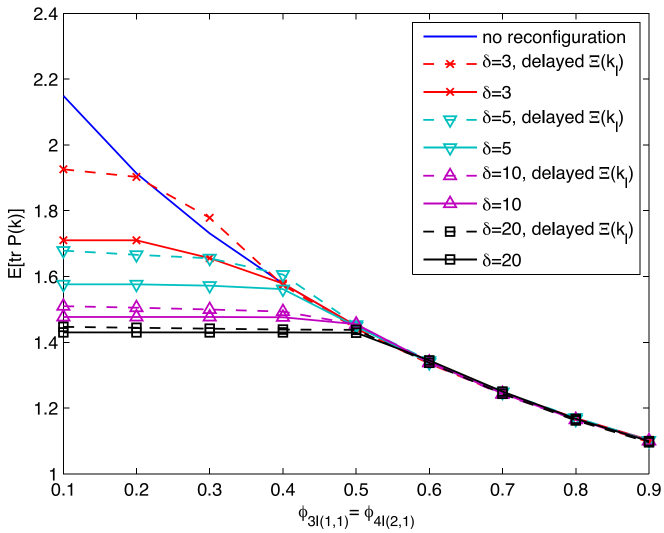

Fig. 9. $\mathbb{E}[\mathrm{tr}\, P(k)]$ with network state detection delays for different holding times.

of values of $\phi_{3|(1,1)}$ and $\phi_{4|(2,1)}$, where reconfiguration gives performance benefits. However, the relative performance gains diminish as δ increases, with little difference between the cases $\delta = 20$ and $\delta = 30$.

The steady state method of Section V-B switches behavior from always changing configuration to no reconfiguration when $\phi_{3|(1,1)}$ and $\phi_{4|(2,1)}$ are greater than around 0.544, irrespective of the holding time distribution. In this case, the method maximizing the packet reception probability outperforms the method of minimizing the steady state upper bounds when the holding times are short, while the steady state method again performs well for long holding times.

D. Delays in Detection of Network State Transitions

An interesting issue arises when changes in the network state $\Xi(k_l)$ are not perfectly detected, but could be delayed or erroneous. Here we present some numerical results on the performance of our methods with respect to delays in the detection of network state transitions. A more thorough investigation of ways to compensate for imperfect knowledge of network states is left for future work.

To model the effect of delays, we will assume that if the detection of the network state transition is delayed, the network will continue to use the links in the current configuration if supported by the new network state. For instance, if we are currently in network configuration 2 and the new network state is 2, then during the time of detection delay the link from sensor 3 to sensor 2 will not be available, since this link is not supported in network configurations 1 or 3 (which are the possible configurations when in network state 2). Once the network state transition has been detected, the network reconfigures as before (with a delay).

We will use the same holding time distributions as in (27). Changes in the network state $\Xi(k_l)$ are detected at time k_l with probability 0.5, and detected at time $k_l + 1$ (i.e., with delay 1) with probability 0.5. Fig. 9 depicts $\mathbb{E}[\mathrm{tr}\, P(k)]$ in solving the suboptimal network reconfiguration problem over one holding period, for holding times of duration $\delta = 3, 5, 10, 20$, both with and without network state detection delay, together with the case of no reconfiguration. For short holding times, there is a significant performance loss, but as the holding times become larger, the performance with network state detection delay becomes closer to the case with no detection delay.

VIII. Conclusion

This paper has presented network topology reconfiguration methods for state estimation in sensor networks over time-varying wireless channels. The optimization of an expected error performance measure which takes into account the cost of reconfiguration, has been studied. A less computationally intensive suboptimal method has been proposed, which in many cases gives identical results to the optimal method. In situations with long holding times, which are likely to be encountered in an industrial setting, numerical results suggest that significant performance gains can be achieved by network reconfiguration.

There are several possible directions to extend the current work. Further analysis of sub-optimal reconfiguration methods which scale well with the number of sensors will be important for larger networks. Another direction is the consideration of imperfect knowledge of the network states, e.g., delays or errors in the estimation of the network state, and ways to compensate for this. In addition, in industrial settings one might also try to anticipate changes in the network state by considering the future movements of machinery, which may possibly reduce the reconfiguration time. Such topics will form the basis of future investigation.

Appendix

A. Proof of Theorem III.2

Consider a policy $\pi^\sharp$. Define the candidate stochastic Lyapunov function:

$$V_l \triangleq \mathrm{tr}\, P(k_l) \tag{28}$$

where $k_l \in \mathbb{K}$ are the random switching times of the semi-Markov chain $\{\Xi\}$. We have

$$\begin{aligned}\mathbb{E}&\left\{V_{l+1}|Z(k_l), \pi^\sharp(j,p^-)\right\}\\ &= \sum_{\delta=1}^{\Delta_{\max}} \mathbb{E}\left\{V_{l+1}|Z(k_l), \pi^\sharp(j,p^-), \Delta_l = \delta\right\}\psi_j(\delta). \end{aligned}\tag{29}$$

Noting that $k_{l+1} = k_l + \Delta_l$, we can write:

$$\begin{aligned}&\mathbb{E}\left\{V_{l+1}|Z(k_l), \pi^\sharp(j,p^-), \Delta_l = \delta\right\}\\ &\quad= \mathbb{E}\left\{\mathrm{tr}\, P(k_l+\delta)|Z(k_l), \pi^\sharp(j,p^-)\right\}\\ &\quad= \mathbb{E}\left\{\mathrm{tr}\, P(k_l+\delta)|Z(k_l), \pi^\sharp(j,p^-), \varrho_\delta(k_l) = 1\right\}\\ &\qquad\times \mathbb{P}\left\{\varrho_\delta(k_l) = 1|Z(k_l), \pi^\sharp(j,p^-)\right\}\\ &\qquad+ \mathbb{E}\left\{\mathrm{tr}\, P(k_l+\delta)|Z(k_l), \pi^\sharp(j,p^-), \varrho_\delta(k_l) = 0\right\}\\ &\qquad\times \mathbb{P}\left\{\varrho_\delta(k_l) = 0|Z(k_l), \pi^\sharp(j,p^-)\right\}.\end{aligned}$$

For the first term above, we can show by similar arguments to [22] that

$$\mathbb{E}\left\{\operatorname{tr} P(k_l+\delta)|Z(k_l),\pi^\sharp(j,p^-),\varrho_\delta(k_l)=1\right\} \times \mathbb{P}\left\{\varrho_\delta(k_l)=1|Z(k_l),\pi^\sharp(j,p^-)\right\} \le W_1^\delta$$

for some finite constant W_1^δ. For the case when $\varrho_\delta(k_l)=0$, the error covariance matrix $P(k_l+\delta)$ is bounded by the worst case, where $\gamma_m(k)=0, \forall\,(m,k)\in\{1,\ldots,M\}\times\{k_l,\ldots,k_l+\delta-1\}$. Therefore,

$$\begin{aligned}
&\mathbb{E}\left\{\operatorname{tr} P(k_l+\delta)|Z(k_l),\pi^\sharp(j,p^-),\varrho_\delta(k_l)=0\right\}\\
&\quad\times \mathbb{P}\left\{\varrho_\delta(k_l)=0|Z(k_l),\pi^\sharp(j,p^-)\right\}\\
&\quad\le \operatorname{tr}\left(A^\delta P(k_l)(A^\delta)^T + A^{\delta-1}Q(A^{\delta-1})^T+\cdots+Q\right)\\
&\quad\times \mu_\delta\left(j,\pi^\sharp(j,p^-),p^-\right)\\
&\quad\le \operatorname{tr} P(k_l)\|A\|^{2\delta}\mu_\delta\left(j,\pi^\sharp(j,p^-),p^-\right)+W_2^\delta\\
&\quad= V_l\|A\|^{2\delta}\mu_\delta\left(j,\pi^\sharp(j,p^-),p^-\right)+W_2^\delta
\end{aligned} \tag{30}$$

for some finite constant W_2^δ. Then

$$\mathbb{E}\left\{V_{l+1}|Z(k_l),\pi^\sharp(j,p^-)\right\} \le \sum_{\delta=1}^{\Delta_{\max}} \mu_\delta\left(j,\pi^\sharp(j,p^-),p^-\right)\|A\|^{2\delta}\psi_j(\delta)V_l + \sum_{\delta=1}^{\Delta_{\max}}\left(W_1^\delta+W_2^\delta\right)\psi_j(\delta).$$

The second summation above is bounded. Thus, if

$$\sum_{\delta=1}^{\Delta_{\max}} \mu_\delta\left(j,\pi^\sharp(j,p^-),p^-\right)\|A\|^{2\delta}\psi_j(\delta)<1$$

then, since $\{Z\}_{\mathbb{K}}$ is Markovian by Lemma III.1, we can use [43, Proposition 3.2] to show that, under the policy $\pi^\sharp$,

$$\mathbb{E}\left\{P(k_l)|Z(0),\pi^\sharp\right\} \le \alpha_1\rho^{k_l}+\beta_1, \quad \forall\, k_l\in\mathbb{K} \tag{31}$$

for some $\rho\in[0,1)$ and finite constants α_1 and β_1. For the times in between transition instants, note that similar to (30), we can find finite constants α_2 and W_3 such that

$$\operatorname{tr} P(k_l+d)\le \|A\|^{2d}\operatorname{tr} P(k_l)+W_3 \le \alpha_2\rho^d\operatorname{tr} P(k_l)+W_3$$

holds for all $d\in\{1,\ldots,\Delta_l\}$. Then, using (31)

$$\begin{aligned}
\mathbb{E}\left\{\operatorname{tr} P(k_l+d)|\pi^\sharp\right\} &\le \alpha_2\alpha_1\rho^{k_l+d}+\alpha_2\beta_1\rho^d+W_3\\
&\le \alpha\rho^{k_l+d}+\beta, \qquad \forall\, d\in\{1,\ldots,\Delta_{\max}\}
\end{aligned}$$

for some finite constants α and β. Since $\rho<1$, this implies that

$$\mathbb{E}\left[\operatorname{tr} P(k)|\pi^\sharp\right] \le \alpha+\beta \triangleq B.$$

This establishes uniform boundedness at all times $k\in\mathbb{N}$ under policy $\pi^\sharp$ when condition (12) is satisfied.

Now, under the optimal reconfiguration policy π^*, we have

$$\begin{aligned}
&\mathbb{E}\left\{\operatorname{tr} P(k_l+1)+\cdots+\operatorname{tr} P(k_l+\Delta_l)|Z(k_l),\pi^*\right\}\\
&\quad\le \mathbb{E}\left\{\operatorname{tr} P(k_l+1)+\cdots+\operatorname{tr} P(k_l+\Delta_l)|Z(k_l),\pi^\sharp\right\}
\end{aligned}$$

for all $Z(k_l)$, so that

$$\begin{aligned}
&\mathbb{E}\left\{\operatorname{tr} P(k_l+1)+\cdots+\operatorname{tr} P(k_l+\Delta_l)|\pi^*\right\}\\
&\quad\le \mathbb{E}\left\{\operatorname{tr} P(k_l+1)+\cdots+\operatorname{tr} P(k_l+\Delta_l)|\pi^\sharp\right\}\\
&\quad\le \Delta_l B\le \Delta_{\max}B.
\end{aligned}$$

Since error covariance matrices have non-negative trace, we have for all $d\in\{1,\ldots,\Delta_l\}$,

$$\mathbb{E}\left\{\operatorname{tr} P(k_l+d)|\pi^*\right\} \le \Delta_{\max}B \triangleq \tilde{B}.$$

This thus establishes uniform boundedness of the Kalman filter under the optimal policy π^*.

B. Proof of Theorem IV.3

Firstly, for $k\in\{k_l,\ldots,k_l+\Delta_{\max}-1\}$, the recursion (16) can be written as

$$\begin{aligned}
&Y(k+1)\\
&= \mathbb{E}\Big\{AY(k)A^T+Q-AY(k)C(k)^T\big(C(k)Y(k)C(k)^T+R\big)^{-1}\\
&\qquad\times C(k)Y(k)A^T\Big|\,\mathcal{U}(k_l),\pi(k_l),s(k)=1\Big\}\\
&\quad\times \mathbb{P}\left\{s(k)=1|\mathcal{U}(k_l),\pi(k_l)\right\}\\
&\quad+\mathbb{E}\Big\{AY(k)A^T+Q-AY(k)C(k)^T\big(C(k)Y(k)C(k)^T+R\big)^{-1}\\
&\qquad\times C(k)Y(k)A^T\Big|\,\mathcal{U}(k_l),\pi(k_l),s(k)=0\Big\}\\
&\quad\times \mathbb{P}\left\{s(k)=0|\mathcal{U}(k_l),\pi(k_l)\right\}
\end{aligned}$$

from which one can derive the bounds

$$\begin{aligned}
\operatorname{tr} Y(k_l+1) &\le W_{1,1}+\operatorname{tr}\left(AY(k_l)A^T+Q\right)\\
&\qquad\times \mathbb{P}\left\{s(k_l)=0|\mathcal{U}(k_l),\pi(k_l)\right\}\\
&= W_{1,1}+\left(\operatorname{tr} Y(k_l)\|A\|^2+W_{1,2}\right)\\
&\qquad\times \nu_1\left(\Xi(k_l),\pi(k_l),\pi(k_{l-1})\right)\\
\operatorname{tr} Y(k_l+2) &\le W_{1,2}+\operatorname{tr}\left(AY(k_l+1)A^T+Q\right)\\
&\qquad\times \mathbb{P}\left\{s(k_l+1)=0|\mathcal{U}(k_l),\pi(k_l)\right\}\\
&\le W_{1,2}+\left(\operatorname{tr} Y(k_l)\|A\|^4+W_{2,2}\right)\\
&\qquad\times \nu_2\left(\Xi(k_l),\pi(k_l),\pi(k_{l-1})\right)\\
&\;\;\vdots\\
\operatorname{tr} Y(k_l+d) &\le W_{1,d}+\left(\operatorname{tr} Y(k_l)\|A\|^{2d}+W_{2,d}\right)\\
&\qquad\times \nu_d\left(\Xi(k_l),\pi(k_l),\pi(k_{l-1})\right)
\end{aligned} \tag{32}$$

for some finite constants $W_{1,d}$ and $W_{2,d}$.

Consider a policy $\pi^\sharp$. Define

$$\bar{V}_l \triangleq \operatorname{tr} Y^\sharp(k_{l-1}+\Delta_{l-1})$$

where $Y^\sharp$ denotes the recursion (16) under policy $\pi^\sharp$. We have

$$\begin{aligned}&\mathbb{E}\left\{\bar{V}_{l+1}|\bar{Z}(k_l),\pi^\sharp(j,p^-)\right\}\\&\quad=\sum_{\delta=1}^{\Delta_{\max}}\mathbb{E}\left\{\bar{V}_{l+1}|\bar{Z}(k_l),\pi^\sharp(j,p^-),\Delta_l=\delta\right\}\psi_j(\delta)\end{aligned}$$

and

$$\begin{aligned}&\mathbb{E}\left\{\bar{V}_{l+1}|\bar{Z}(k_l),\pi^\sharp(j,p^-),\Delta_l=\delta\right\}\\&\quad=\mathbb{E}\left\{\operatorname{tr}Y^\sharp(k_l+\delta)|\bar{Z}(k_l),\pi^\sharp(j,p^-),s(k_l+\delta-1)=1\right\}\\&\qquad\times\mathbb{P}\left\{s(k_l+\delta-1)=1|\bar{Z}(k_l),\pi^\sharp(j,p^-)\right\}\\&\qquad+\mathbb{E}\left\{\operatorname{tr}Y^\sharp(k_l+\delta)|\bar{Z}(k_l),\pi^\sharp(j,p^-),s(k_l+\delta-1)=0\right\}\\&\qquad\times\mathbb{P}\left\{s(k_l+\delta-1)=0|\bar{Z}(k_l),\pi^\sharp(j,p^-)\right\}\\&\quad\le W_{1,\delta}+\left(\operatorname{tr}Y^\sharp(k_l)\|A\|^{2\delta}+W_{2,\delta}\right)\nu_\delta\left(j,\pi^\sharp(j,p^-),p^-\right)\end{aligned}$$

for some finite constants $W_{1,\delta}$ and $W_{2,\delta}$, where the inequality comes from making use of the bounds (32). Using Lemma IV.2 and similar arguments as in the proof of Theorem III.2, we can show that if

$$\sum_{\delta=1}^{\Delta_{\max}}\nu_\delta\left(j,\pi^\sharp(j,p^-),p^-\right)\|A\|^{2\delta}\psi_j(\delta)<1$$

holds, then there exists finite constants $B,\tilde{B}$, such that

$$\operatorname{tr}Y^\sharp(k)\le B,\quad\forall\,k$$

under policy $\pi^\sharp$, and

$$\operatorname{tr}Y(k)\le\tilde{B},\quad\forall\,k$$

under policy π^*. Since $Y(k)$ upper bounds $\mathbb{E}\{P(k)\}$ by Lemma IV.1, this then implies

$$\mathbb{E}\{\operatorname{tr}P(k)\}\le\tilde{B},\quad\forall\,k$$

and, hence, the uniform boundedness of the Kalman filter under policy π^*.

REFERENCES

[1] A. S. Leong, D. E. Quevedo, A. Ahlén, and K. H. Johansson, "Network topology reconfiguration for state estimation over sensor networks with correlated packet drops," in *Proc. IFAC World Congress*, Cape Town, South Africa, Aug. 2014, pp. 5532–5537.

[2] B. Sinopoli, L. Schenato, M. Franceschetti, K. Poolla, M. I. Jordan, and S. S. Sastry, "Kalman filtering with intermittent observations," *IEEE Trans. Autom. Control*, vol. 49, no. 9, pp. 1453–1464, Sep. 2004.

[3] K. Plarre and F. Bullo, "On Kalman filtering for detectable systems with intermittent observations," *IEEE Trans. Autom. Control*, vol. 54, no. 2, pp. 386–390, Feb. 2009.

[4] Y. Mo and B. Sinopoli, "Kalman filtering with intermittent observations: Tail distribution and critical value," *IEEE Trans. Autom. Control*, vol. 57, no. 3, pp. 677–689, Mar. 2012.

[5] X. Liu and A. J. Goldsmith, "Kalman filtering with partial observation losses," in *Proc. IEEE Conf. Decision Control*, Bahamas, Dec. 2004, pp. 1413–1418.

[6] V. Gupta, N. C. Martins, and J. S. Baras, "Optimal output feedback control using two remote sensors over erasure channels," *IEEE Trans. Autom. Control*, vol. 54, no. 7, pp. 1463–1476, Jul. 2009.

[7] T. Sui, K. You, and M. Fu, "Stability conditions for multi-sensor state estimation over a lossy network," *Automatica*, vol. 53, no. 1, pp. 1–9, 2015.

[8] M. Epstein, L. Shi, A. Tiwari, and R. M. Murray, "Probabilistic performance of state estimation across a lossy network," *Automatica*, vol. 44, pp. 3046–3053, Dec. 2008.

[9] M. Huang and S. Dey, "Stability of Kalman filtering with Markovian packet losses," *Automatica*, vol. 43, pp. 598–607, 2007.

[10] K. You, M. Fu, and L. Xie, "Mean square stability for Kalman filtering with Markovian packet losses," *Automatica*, vol. 47, no. 12, pp. 1247–1257, Dec. 2011.

[11] A. Censi, "Kalman filtering with intermittent observations: Convergence for semi-Markov chains and an intrinsic performance measure," *IEEE Trans. Autom. Control*, vol. 56, no. 2, pp. 376–381, Feb. 2011.

[12] L. Schenato, "Optimal estimation in networked control systems subject to random delay and packet drop," *IEEE Trans. Autom. Control*, vol. 53, no. 5, pp. 1311–1317, Jun. 2008.

[13] V. Gupta, A. F. Dana, J. P. Hespanha, R. M. Murray, and B. Hassibi, "Data transmission over networks for estimation and control," *IEEE Trans. Autom. Control*, vol. 54, no. 8, pp. 1807–1819, Aug. 2009.

[14] L. Shi, "Kalman filtering over graphs: Theory and applications," *IEEE Trans. Autom. Control*, vol. 54, no. 9, pp. 2230–2234, Sep. 2009.

[15] A. Chiuso and L. Schenato, "Information fusion strategies and performance bounds in packet-drop networks," *Automatica*, vol. 47, pp. 1304–1316, Jul. 2011.

[16] D. E. Quevedo, A. Ahlén, A. S. Leong, and S. Dey, "On Kalman filtering over fading wireless channels with controlled transmission powers," *Automatica*, vol. 48, no. 7, pp. 1306–1316, Jul. 2012.

[17] D. E. Quevedo, J. Østergaard, and A. Ahlén, "Power control and coding formulation for state estimation with wireless sensors," *IEEE Trans. Control Syst. Technol.*, vol. 22, no. 2, pp. 413–427, Mar. 2014.

[18] A. S. Leong and D. E. Quevedo, "On the use of a relay for Kalman filtering over packet dropping links," in *Proc. Amer. Control Conf. (ACC)*, Washington, DC, Jun. 2013.

[19] T. Ho and D. S. Lun, *Network Coding: An Introduction*. Cambridge, U.K.: Cambridge University Press, 2008.

[20] Y. Mo, E. Garone, A. Casavola, and B. Sinopoli, "Stochastic sensor scheduling for energy constrained estimation in multi-hop wireless sensor networks," *IEEE Trans. Autom. Control*, vol. 56, no. 10, pp. 2489–2495, Oct. 2011.

[21] L. Shi, A. Capponi, K. H. Johansson, and R. M. Murray, "Resource optimisation in a wireless sensor network with guaranteed estimator performance," *IET Control Theory Appl.*, vol. 4, no. 5, pp. 710–723, 2010.

[22] D. E. Quevedo, A. Ahlén, and K. H. Johansson, "State estimation over sensor networks with correlated wireless fading channels," *IEEE Trans. Autom. Control*, vol. 58, no. 3, pp. 581–593, Mar. 2013.

[23] E. Baskaran, J. Llorca, S. D. Milner, and C. C. Davis, "Topology reconfiguration with successive approximations," in *Proc. Military Commun. Conf. (MILCOM)*, Orlando, FL, 2007.

[24] Y. E. Krasteva, J. Portilla, E. de la Torre, and T. Riesgo, "Embedded runtime reconfigurable nodes for wireless sensor networks applications," *IEEE Sensors J.*, vol. 11, no. 9, pp. 1800–1810, Sep. 2011.

[25] R. Ramakrishnan, N. S. Ram, and O. A. Alheyasat, "A cost aware reconfiguration technique for recovery in wireless mesh networks," in *Proc. Int. Conf. Recent Trends in Inform. Technol. (ICRTIT)*, Chennai, India, 2012, pp. 302–307.

[26] B. D. O. Anderson and J. B. Moore, *Optimal Filtering*. Upper Saddle River, NJ, USA: Prentice Hall, 1979.

[27] D. Bertsekas and R. Gallager, *Data Networks*, 2nd ed. Upper Saddle River, NJ, USA: Prentice-Hall, 1992.

[28] HART Commun. Found., Control with WirelessHART, 2009.

[29] P. Agrawal, A. Ahlén, T. Olofsson, and M. Gidlund, "Long term channel characterization for energy efficient transmission in industrial environments," *IEEE Trans. Commun.*, vol. 62, no. 8, pp. 3004–3014, Aug. 2014.

[30] S. M. Ross, *Stochastic Processes*, 2nd ed. Hoboken, NJ, USA: John Wiley & Sons, 1996.

[31] V. G. Kulkarni, *Modeling and Analysis of Stochastic Systems,* 2nd ed. Boca Raton, FL, USA: CRC Press, 2010.

[32] J. N. Al-Karaki and A. E. Kamal, "Routing techniques in wireless sensor networks: A survey," *IEEE Wireless Commun. Mag.*, vol. 11, no. 6, pp. 6–28, Dec. 2004.

[33] V. Pham, E. Larsen, K. Øvsthus, P. Engelstad, and Ø. Kure, "Rerouting time and queueing in proactive *ad hoc* networks," in *Proc. IEEE Int. Performance Comput. Commun. Conf. (IPCCC)*, New Orleans, LA, USA, 2007, pp. 160–169.

[34] HART Commun. Found., System redundancy with Wireless HART, 2009.

[35] D. E. Quevedo, A. Ahlén, and J. Østergaard, "Energy efficient state estimation with wireless sensors through the use of predictive power control and coding," *IEEE Trans. Signal Process.*, vol. 58, no. 9, pp. 4811–4823, Sep. 2010.

[36] M. Nourian, A. S. Leong, and S. Dey, "Optimal energy allocation for Kalman filtering over packet dropping links with imperfect acknowledgments and energy harvesting constraints," *IEEE Trans. Autom. Control*, vol. 59, no. 8, pp. 2128–2143, Aug. 2014.

[37] S. T. Jawaid and S. L. Smith, "A complete algorithm for the infinite horizon sensor scheduling problem," in *Proc. Amer. Control Conf. (ACC)*, Portland, OR, USA, Jun. 2014.

[38] L. Schenato, B. Sinopoli, M. Franceschetti, K. Poolla, and S. S. Sastry, "Foundations of control and estimation over lossy networks," *Proc. IEEE*, vol. 95, no. 1, pp. 163–187, Jan. 2007.

[39] S. Dey, A. S. Leong, and J. S. Evans, "Kalman filtering with faded measurements," *Automatica*, vol. 45, no. 10, pp. 2223–2233, Oct. 2009.

[40] E. Rohr, D. Marelli, and M. Fu, "Statistical properties of the error covariance in a Kalman filter with random measurement losses," in *Proc. IEEE Conf. Decision Control*, Atlanta, GA, USA, Dec. 2010, pp. 5881–5886.

[41] O. Incel, A. Ghosh, B. Krishnamachari, and K. Chintalapudi, "Fast data collection in tree-based wireless sensor networks," *IEEE Trans. Mobile Comput.*, vol. 11, no. 1, pp. 86–99, Jan. 2012.

[42] H. Zhang, P. Soldati, and M. Johansson, "Performance bounds and latency-optimal scheduling for convergecast in WirelessHART networks," *IEEE Trans. Wireless Commun.*, vol. 12, no. 6, pp. 2688–2696, Jun. 2013.

[43] S. P. Meyn, "Ergodic theorems for discrete time stochastic systems using a stochastic Lyapunov function," *SIAM J. Control Optim.*, vol. 27, no. 6, pp. 1409–1439, Nov. 1989.

Alex S. Leong (S'03–M'08) was born in Macau, in 1980. He received the B.S. degree in mathematics and B.E. degree in electrical engineering, in 2003, and the Ph.D. degree in electrical engineering, in 2008, all from the University of Melbourne, Parkville, Australia.

He is currently a Research Associate at Paderborn University, Germany. He was with the Department of Electrical and Electronic Engineering at the University of Melbourne from 2008 to 2015. His research interests include statistical signal processing, signal processing for sensor networks, and networked control systems.

Dr. Leong was the recipient of the L. R. East Medal from Engineers Australia in 2003, an Australian Postdoctoral Fellowship from the Australian Research Council in 2009, and a Discovery Early Career Researcher Award from the Australian Research Council in 2012.

Daniel E. Quevedo (S'97–M'05–SM'14) holds the Chair in Automatic Control (*Regelungs- und Automatisierungstechnik*) at Paderborn University, Germany. He received Ingeniero Civil Electrónico and M.Sc. degrees from the Universidad Técnica Federico Santa María, Chile, in 2000. In 2005, he was awarded the Ph.D. degree from The University of Newcastle, Australia.

Dr. Quevedo was supported by a full scholarship from the alumni association during his time at the Universidad Técnica Federico Santa María and received several university-wide prizes upon graduating. He received the IEEE Conference on Decision and Control Best Student Paper Award in 2003 and was also a finalist in 2002. In 2009, he was awarded a five-year Research Fellowship from the Australian Research Council.

Professor Quevedo is Editor of the *International Journal of Robust and Nonlinear Control* and serves as Chair of the IEEE Control Systems Society *Technical Committee on Networks & Communication Systems*. His research interests are in automatic control, signal processing, and power electronics.

Anders Ahlén (S'80–M'84–SM'90) is full professor and holds the chair in Signal Processing at Uppsala University, where he is the head of the Signals and Systems Division of The Department of Engineering Sciences. He was born in Kalmar, Sweden, and received the PhD degree in Automatic Control from Uppsala University. He was with the Systems and Control Group, Uppsala University from 1984–1992 as an Assistant and Associate Professor in Automatic Control. During 1991, he was a visiting researcher at the Department of Electrical and Computer Engineering, The University of Newcastle, Australia. He was a visiting professor at the same university in 2008. In 1992, he was appointed Associate Professor of Signal Processing at Uppsala University. During 2001–2004, he was the CEO of Dirac Research AB, a company offering state-of-the-art audio signal processing solutions. He is currently the chairman of the board of directors of the same company. He was a member of the Center of Excellence in Wireless Sensor Networks, WISENET, from 2007 to 2013. His research interest includes Signal Processing, Communications and Control, and is currently focused on Signal Processing for Wireless Communications, Wireless Sensor Networks, Wireless Control, and Audio Signal Processing.

From 1998 to 2004, he was the Editor of Signal and Modulation Design for the IEEE TRANSACTIONS ON COMMUNICATIONS.

Karl H. Johansson (S'92–M'98–SM'08–F'13) is Director of the ACCESS Linnaeus Centre and Professor at the School of Electrical Engineering, KTH Royal Institute of Technology, Sweden. He is a Wallenberg Scholar and has held a Senior Researcher Position with the Swedish Research Council. He also heads the Stockholm Strategic Research Area ICT The Next Generation. He received MSc and PhD degrees in Electrical Engineering from Lund University. He has held visiting positions at UC Berkeley, California Institute of Technology, Nanyang Technological University, and Institute of Advanced Studies Hong Kong University of Science and Technology. His research interests are in networked control systems, cyber-physical systems, and applications in transportation, energy, and automation systems. He has been a member of the IEEE Control Systems Society Board of Governors and the Chair of the IFAC Technical Committee on Networked Systems. He has been on the Editorial Boards of several journals, including Automatica, IEEE TRANSACTIONS ON AUTOMATIC CONTROL, AND IET CONTROL THEORY AND APPLICATIONS. He is currently a Senior Editor of IEEE TRANSACTIONS ON CONTROL OF NETWORK SYSTEMS and Associate Editor of European Journal of Control. He has been Guest Editor for a special issue of IEEE TRANSACTIONS ON AUTOMATIC CONTROL on cyber-physical systems and one of IEEE Control Systems Magazine on cyber-physical security. He was the General Chair of the ACM/IEEE Cyber-Physical Systems Week 2010 in Stockholm and IPC Chair of many conferences. He has served on the Executive Committees of several European research projects in the area of networked embedded systems. He received the Best Paper Award of the IEEE International Conference on Mobile Ad-hoc and Sensor Systems in 2009 and the Best Theory Paper Award of the World Congress on Intelligent Control and Automation in 2014. In 2009 he was awarded Wallenberg Scholar, as one of the first ten scholars from all sciences, by the Knut and Alice Wallenberg Foundation. He was awarded Future Research Leader from the Swedish Foundation for Strategic Research in 2005. He received the triennial Young Author Prize from IFAC in 1996 and the Peccei Award from the International Institute of System Analysis, Austria, in 1993. He received Young Researcher Awards from Scania in 1996 and from Ericsson in 1998 and 1999.