

# Energy Efficient State Estimation With Wireless Sensors Through the Use of Predictive Power Control and Coding

Daniel E. Quevedo, *Member, IEEE*, Anders Ahlén, *Senior Member, IEEE*, and Jan Østergaard, *Member, IEEE*

**Abstract**—We study state estimation via wireless sensors over fading channels. Packet loss probabilities depend upon time-varying channel gains, packet lengths and transmission power levels of the sensors. Measurements are coded into packets by using either independent coding or distributed zero-error coding. At the gateway, a time-varying Kalman filter uses the received packets to provide the state estimates. To trade sensor energy expenditure for state estimation accuracy, we develop a predictive control algorithm which, in an online fashion, determines the transmission power levels and codebooks to be used by the sensors. To further conserve sensor energy, the controller is located at the gateway and sends coarsely quantized power increment commands, only whenever deemed necessary. Simulations based on real channel measurements illustrate that the proposed method gives excellent results.

**Index Terms**—Packet loss, power control, scheduling, source coding, state estimation, wireless sensors.

## I. INTRODUCTION

THE interest in estimation and control over lossy communication links has increased tremendously in recent years; see, e.g., [1]–[3]. In particular, with the rapid evolution of wireless sensor networks, see e.g., [4]–[6], the use of wireless sensors (and actuators) has become an interesting alternative. The driving force behind this evolution from wired to wireless is the low deployment cost: There is no need for extensive wiring, either in new installations or for upgrading old systems. In addition, wireless sensors and actuators can be placed where wires cannot go, or where power sockets are not available.

A drawback of using wireless communication channels lies in that they are subject to fading and interference, which frequently

lead to packet errors. Depending on the application, the wireless channel can be constant or time varying. The time variability may be caused by moving objects, vehicles, people, and so forth. Also, the receiver or the transmitter can be mounted on a moving object, which may be the case in process industry. Therefore, in addition to the propagation path loss, the channel may be subject to either slow or fast fading, or both.

Beside the fading channel, another important issue, which arises in the absence of power sockets, is the need for energy conservation. Even though power scavenging is presently a hot topic in wireless sensor research, saving energy is of utmost importance to avoid unnecessary maintenance, such as the replacement of batteries; see, e.g., [7]–[10].

Interestingly, the time-variability of the fading channel can be compensated for by adjusting the power levels and also the lengths of the transmitted packets. Indeed, the packet loss probabilities depend, in a nonlinear fashion, upon the time-varying channel gains, the packet lengths and the power levels used by the sensors. To keep packet error rates low, high transmission power and short packet lengths need to be used. Unfortunately, using high transmission power is not an option with wireless sensors, since sensors are unavoidably limited in power, computational capacity and memory [7]. Also, sending short packets will lead to large quantization effects, if coding is not done with care.

In this paper, we will present a power and coding control algorithm for state estimation with wireless sensors. In our architecture, several sensors take noisy and possibly different output measurements of an autoregressive moving average (ARMA) stochastic process. These measurements are coded and transmitted over a fading channel (generating random packet loss) to a single gateway. Received packets are then used to remotely estimate the system state sequence by means of a time-varying Kalman filter.

To keep the sensors simple and energy efficient, sensor nodes do not communicate with each other. Thus, joint coding of the measurements taken by different sensors is not possible. Instead, coding needs to be carried out either independently or with separate encoding followed by joint decoding at the gateway. This *distributed source coding* problem is a widely studied concept in information theory; see, e.g., [11]–[13]. One such technique is named zero-error coding [14], which is what we will use in this work.

Within the setting described above, our main contribution lies in developing a centralized dynamic controller, which is located at the gateway and jointly decides upon the transmission power

Manuscript received August 05, 2009; accepted April 29, 2010. Date of publication May 20, 2010; date of current version August 11, 2010. The associate editor coordinating the review of this manuscript and approving it for publication was Dr. Milica Stojanovic. This research was supported under the Australian Research Council's Discovery Projects funding scheme (project number DP0988601).

D. E. Quevedo is with the School of Electrical Engineering and Computer Science, The University of Newcastle, NSW 2308, Australia (e-mail: dquevedo@ieee.org).

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

J. Østergaard is with the Department of Electronic Systems, Aalborg University, 9220 Aalborg, Denmark (e-mail: janoe@ieee.org).

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/TSP.2010.2050883

levels and coding method to be used by each sensor. The controller uses elements of nonlinear (stochastic) predictive control [15], [16].<sup>1</sup> It trades sensor energy use for estimation accuracy. The latter is quantified by the expected value of the Kalman filter covariance matrix. To further conserve energy, a pre-designed set of codebooks is stored at the sensors and the proposed controller sends only codebook indexes and coarsely quantized power increments to the sensor nodes, whenever deemed necessary. Consequently, the system to be controlled is not only nonlinear and stochastic (due to the occurrence of random packet errors), but also subject to finite-set constraints on decision variables, i.e., the power level increments and codebook indices. Key to keeping the computational burden of calculating the optimal codebook indexes and power commands limited is the fact that the occurrence of packet errors constitutes a binary random variable. Thus, expected values can be exactly evaluated via finite sums over the possible transmission outcome scenarios, i.e., no integrals need to be evaluated or approximated. The present paper extends our recent works [21], [22] by incorporating coding of the sensor measurements into the formulation.

An outline of the remainder of this work is as follows: In Section II, we state the wireless state estimation problem in precise terms. Sections III–V revise elements of source coding, wireless transmission and state estimation which are relevant for power and coder design. The proposed control algorithm is then presented in Section VI. Computational aspects are discussed in Section VII, where also a suboptimal algorithm is presented. Simulation results are included in Section VIII. Section IX draws conclusions. Additional information on coder design is given in the Appendix.

## II. PROBLEM STATEMENT

Consider a stationary ARMA process described in state-space form via:

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

where  $A \in \mathbb{R}^{n_x \times n_x}$ ,  $n_x \in \mathbb{N} \triangleq \{1, 2, \dots\}$  is the system matrix and  $x = \{x(k)\}_{k \in \mathbb{N}_0}$  is the state sequence. The initial system state is zero-mean, but otherwise arbitrary distributed with covariance  $P_0 \in \mathbb{R}^{n_x \times n_x}$ . The driving noise process  $w = \{w(k)\}_{k \in \mathbb{N}_0}$  is also arbitrary distributed; each  $w(k)$  is zero-mean with covariance matrix  $Q$ .

To remotely estimate the state sequence  $x$ , a set of  $M \in \mathbb{N}$  wireless sensors are used. Each sensor  $m$  provides a scalar noisy measurement signal, say  $y_m = \{y_m(k)\}_{k \in \mathbb{N}_0}$ , where

$$y_m(k) = C_m x(k) + v_m(k), \quad m \in \{1, 2, \dots, M\}. \quad (2)$$

In (2),  $v_m = \{v_m(k)\}_{k \in \mathbb{N}_0}$  is an arbitrarily distributed zero-mean noise process, where each  $v_m(k)$  has covariance  $R_m$ .

The values  $y_m(k)$  are coded and then transmitted through wireless links to the gateway. The received signals are then used to remotely estimate the state of the system (1). Fig. 1 depicts the overall configuration of the system under study.

<sup>1</sup>Related techniques have also been used in the context of AD conversion [17], digital channel equalization [18], discrete coefficient filter design [19], and quantization in filter banks [20].

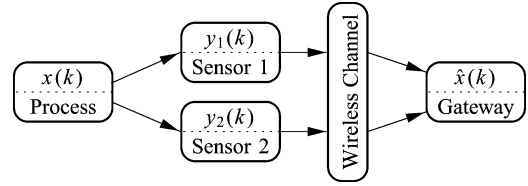


Fig. 1. State estimation with  $M = 2$  wireless sensors.

The distinguishing aspect of the problem at hand lies in that the use of a wireless channel introduces random packet loss. Loss probabilities depend upon the propagation medium, but are also affected by packet lengths and power levels used by the transmitter. In fact, for given channel gains, lower loss probabilities can be achieved by using shorter packet lengths and higher transmission power. Hence, the design of the state estimation scheme in Fig. 1 involves trading sensor battery use for state estimation quality. There also exists a tradeoff when designing the coding method for the sensor measurements, since reducing the bit rate (and, thus, the number of bits contained in one packet) reduces the total transmission energy, as well as the probability of packet error. However, low bit rates, unfortunately, also leads to large quantization effects in the transmitted signals.

In Section VI, we will propose a centralized control algorithm, which can be implemented at the gateway and aims to achieve an optimal tradeoff between sensor energy use and state estimation accuracy. For that purpose, the controller determines, in an online fashion, the power levels, bit rates, and coding schemes to be used by the  $M$  sensors. Before presenting our algorithm, we will first describe the components of the scheme in Fig. 1, namely the coding schemes used by the sensors, the wireless channels, and the state estimator used at the gateway.

## III. CODING THE SENSOR MEASUREMENTS

Each sensor node is equipped with an encoder, denoted  $\mathcal{E}_m$ , which maps each measurement value  $y_m(k) \in \mathbb{R}$  to a sequence of bits  $s_m(k)$ . For that purpose,  $\mathcal{E}_m$  consists of two components: a scalar uniform quantizer, denoted  $\mathcal{Q}_m$ , and an entropy coder,  $\text{EC}_m$ ; see [23] for an introduction to source coding. At the gateway, the received symbols are then passed through a joint entropy decoder and a reconstruction function which outputs the values  $\hat{y}_m(k)$ ; see Fig. 2. The entropy coders can either perform independent coding or distributed zero-error coding (ZEC) at a given set of pre-defined expected bit rates. The gateway determines, at each time-instant, which type of coding is to be used by the sensors (and at which bit rate) and communicates this decision to the  $M$  coders.

### A. The Quantizers $\mathcal{Q}_m$

Uniform scalar quantization can be implemented by simply setting the quantizer output  $i_m(k)$  as

$$i_m(k) = \left\lfloor \frac{y_m(k)}{\Delta_m(k)} \right\rfloor$$

where  $\lfloor \cdot \rfloor$  denotes rounding to the nearest integer and where  $\Delta_m(k)$  is the stepsize at time  $k$ . The associated reconstructed

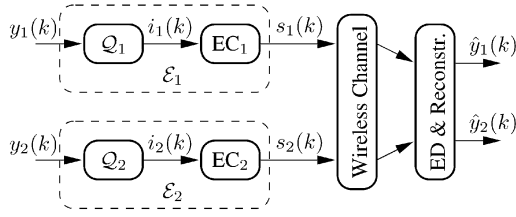


Fig. 2. Distributed Coding with  $M = 2$  sensors. Measurements  $y_1(k)$  and  $y_2(k)$  are quantized, entropy coded and transmitted over a fading channel. At the receiver, Error detection (ED) and reconstruction is performed yielding  $\hat{y}_1(k)$  and  $\hat{y}_2(k)$ .

signal is then obtained by applying the inverse scaling, i.e.,  $\hat{y}_m(k) = i_m(k)\Delta_m(k)$ . The stepsizes  $\Delta_m(k)$  are related to expected bit rates  $b_m(k)$  according to [13], [23]

$$b_m(k) \approx H(i_m(k)) \approx h(y_m(k)) - \log_2(\Delta_m(k)) \quad (3)$$

where  $H(i_m(k))$  denotes the discrete entropy of  $i_m(k)$ , and  $h(y_m(k))$  is the differential entropy of  $y_m(k)$ .

There exists a tradeoff between bit rates of a quantizer, and the distortion introduced, namely

$$D_m(k) \triangleq \mathbb{E}\|y_m(k) - \hat{y}_m(k)\|^2 \approx \frac{(\Delta_m(k))^2}{12} = \left(\frac{1}{12}\right) 2^{2(h(y_m(k)) - b_m(k))} \quad (4)$$

where we have used (3). Thus, larger bit rates give smaller quantization errors. However, larger bit rates also give rise to larger packets to be transmitted and, thus, come at the expense of more energy use and channel utilization. Consequently, bit rates have to be assigned carefully. This forms one of the main themes of the present work.

*Example 1 (The Gaussian Case):* If  $y_m$  in (2) is a stationary Gaussian process, then, under high-resolution assumptions, its differential entropy is given by

$$h(y_m) = \frac{1}{2} \log_2(2\pi e \sigma_{y_m}^2). \quad (5)$$

The approximation in (4) then gives that the expected distortion of an entropy-constrained scalar uniform quantizer satisfies

$$D_m(k) \approx \frac{\pi e}{6} \sigma_{y_m}^2 2^{-2b_m(k)} \quad (6)$$

where the variance of  $y_m$  and the covariance of  $x$  are

$$\sigma_{y_m}^2 = C_m P_x C_m^T + R_m \quad \text{and} \quad P_x = A P_x A^T + Q. \quad (7)$$

## B. Entropy Coding

Each entropy coder  $EC_m$ , see Fig. 2, consist of a finite collection of, say  $\mathcal{J}_m$ , codebooks. These depend on the bit rate  $b_m$  and the type of coding used. Entropy coding can be performed

by a simple table-lookup since the quantization operation directly gives the index of the codeword in the table.

1) *Independent Coding:* The simplest way of performing entropy coding on  $i_m$  is to use independent coding for each of the  $M$  quantizers. Since this requires a codebook for the entropy coder for every possible choice of scaling factor  $\Delta_m(k)$ , it is necessary (due to memory considerations) to discretize the alphabet of  $\Delta_m(k)$ . This is equivalent to discretizing the set of possible  $b_m(k)$  values; see (3). Discretization can be performed offline, and in advance, through computer simulations.

2) *Distributed Source Coding:* In distributed source coding schemes, the designer has the freedom to choose the hierarchy between individual coders. We will adopt an asymmetric ZEC strategy [14]. We, thus, first quantize the measurements independently using the scalar quantizers described in Section III-A and then use dependent entropy coders, i.e., coders which depend upon the statistics of other plant outputs. (They do not depend upon the actual realizations.)

*Example 2 (Asymmetric ZEC):* One of the coders, say  $EC_1$ , is the *dominant coder*. It performs independent coding as described above. Hereafter, another coder, say  $EC_2$ , performs independent scalar quantization followed by entropy coding, where now the entropy coding is done with respect to the entropy code of  $EC_1$ . This will generally give a smaller bit rate, than if independent coding is used.

If the gateway receives both  $s_1(k)$  and  $s_2(k)$ , then, with ZEC, it is possible to reconstruct  $\hat{y}_1(k)$  and  $\hat{y}_2(k)$ . If only  $s_1(k)$  is received, then the gateway can still obtain  $\hat{y}_1(k)$ , but, of course, not  $\hat{y}_2(k)$ . However, if only  $s_2(k)$  is received, then the gateway cannot reconstruct neither  $\hat{y}_1(k)$  nor  $\hat{y}_2(k)$ . Consequently, if channel 1 and 2 are both reliable or if, at least, the dominant channel is, then it will often be beneficial to employ ZEC. On the other hand, if all channels are poor, then often independent coding will give better performance. Independent coding does not exploit the redundancy between the sensors in the encoded data, which gives enhanced robustness in the case of transmission errors. This will become apparent in the simulation results presented in Section VIII.  $\square$

It is important to note that quantization is performed exactly in the same manner for ZEC and for independent coding. Thus, the distortion due to quantization (whenever measurements are reconstructed) remains the same and is given by (6). The gain by using ZEC is a reduction in the number of bits required for representing the set of encoded measurements,  $\{s_m\}$ , i.e., on the effective bit rates (or packet lengths) as seen by the communication channel.

In the sequel,  $\tilde{b}_m(k) > 0$  denotes the bit rates obtained after entropy coding. With independent coding, we have  $\tilde{b}_m(k) = b_m(k)$ , see (6) and (3).<sup>2</sup> On the other hand, ZEC often gives  $\tilde{b}_m(k) < b_m(k)$ , see simulation results included in Section VIII. Some additional background on ZEC is given in the Appendix.

*Remark 1 (Exploiting Temporal Correlation):* Given (1), one could reduce bit rates further by taking into account the temporal correlation in each signal  $y_m$ . This can be done, for example,

<sup>2</sup>We assume that the expected packet length is equal to  $\tilde{b}_m(k)$ , i.e., we do not explicitly take into account any channel coding.

by encoding innovations, requiring more complexity at the sensors. Furthermore, due to possible packet dropouts, care must be taken to guarantee that the encoder and decoder are synchronized. In the present work, we restrict our attention to simple and robust encoding schemes where the current measurements are encoded independently of past values.  $\square$

#### IV. TRANSMISSION EFFECTS AND ENERGY EXPENDITURE

Since the  $M$  links between sensors and gateway are wireless, see Fig. 1, transmission errors are likely to occur. In this section, we will give details on how packet loss probabilities depend upon the time-varying channel gains, packet lengths  $\{\tilde{b}_m\}$ , and power levels used by the sensors.<sup>3</sup>

We will model transmission effects by introducing the  $M$  arrival processes  $\gamma_m = \{\gamma_m(k)\}_{k \in \mathbb{N}_0}$ :

$$\gamma_m(k) = \begin{cases} 1 & \text{if } s_m(k) \text{ arrives error-free at time } k \\ 0 & \text{otherwise.} \end{cases} \quad (8)$$

The associated success probabilities depend on the propagation environment, on  $\tilde{b}_m(k)$ , and on the transmission power used by the sensor radio power amplifiers, which we denote as  $u_m(k)$ . To be more specific, for i.i.d. bit-errors the conditional success probabilities satisfy

$$\mathcal{P}\{\gamma_m(k) = 1 | u_m(k), g_m(k), \tilde{b}_m(k)\} = \lambda_m(k) \quad (9)$$

where

$$\lambda_m(k) = \left(1 - \beta_m(u_m(k)g_m(k))\right)^{\tilde{b}_m(k)}, \quad m \in \{1, \dots, M\} \quad (10)$$

where  $g_m(k)$  denotes the channel power gain, i.e., the square of the magnitude of the complex channel<sup>4</sup>, and  $\beta_m(\cdot): [0, \infty) \rightarrow [0, 1]$  denotes the bit-error rate (BER). The latter is a monotonically decreasing function, which depends on the modulation scheme employed. For simplicity, we shall in the sequel refer to  $g_m(k)$  as the channel gain.

*Example 3 (Transmission Model):* If binary phase shift keying is used over an additive white Gaussian noise channel with constant signal-to-noise ratio SNR, then

$$\beta = f_Q\left(\sqrt{2\text{SNR}}\right) \quad (11)$$

where  $f_Q(z) \triangleq (1/\sqrt{2\pi}) \int_z^\infty \exp(-\eta^2/2) d\eta$ ; see [24].

Although the above model is only valid in the time-invariant i.i.d. case, we shall adopt it also for time-varying channels, power levels and bit rates. For that purpose, we introduce the instantaneous signal-to-noise ratio for each channel  $m$  via<sup>5</sup>

$$\text{SNR}_m(k) = \frac{g_m(k)u_m(k)}{rk_B T}, \quad m \in \{1, \dots, M\}$$

<sup>3</sup>In the present work, we will assume that sensor data is not affected by Multiple Access Interference (MAI). Extensions of our framework to include MAI does not present conceptual difficulties.

<sup>4</sup>Note that  $g_m(k)$  is here defined to include also path-loss, power amplifier efficiency, antenna gain and noise figure.

<sup>5</sup> $\text{SNR}_m(k)$  denotes the signal-to-noise ratio at the receiver, after the matched filter.

where  $k_B$  is the Boltzmann constant,  $T$  is the temperature and  $r$  is the channel bit rate. We furthermore adopt a block fading model, where the channel is constant over the duration of one packet, but may be subject to fading between packets. Expression (10) then gives

$$\lambda_m(k) = \left(1 - f_Q\left(\sqrt{\frac{2g_m(k)u_m(k)}{rk_B T}}\right)\right)^{\tilde{b}_m(k)}. \quad (12)$$

We will use this model in Section VIII.  $\square$

It follows from (10) [see also (12)] that one can improve transmission reliability and, thus, state estimation accuracy for a given propagation environment by transmitting shorter packets and/or by simply increasing the power used by the transmitter. Unfortunately, as we have seen in Section III, smaller values of packet lengths  $\tilde{b}_m(k)$  will lead to larger quantization distortion. Furthermore, when using wireless sensors, it is of fundamental importance to save energy, since sensor nodes are expected to be operational for several years without maintenance. This motivates us to use the available energy resources with care.

Before proceeding, we note that one can quantify the energy used by each sensor  $m \in \{1, \dots, M\}$  at a given (discrete) time instant,  $k$ , via  $E_m(\tilde{b}_m(k)u_m(k))$ , where

$$E_m(\tilde{b}_m(k)u_m(k)) \triangleq \begin{cases} \frac{\tilde{b}_m(k)u_m(k)}{r} + E_P, & \text{if } u_m(k) > 0 \\ 0, & \text{if } u_m(k) = 0. \end{cases} \quad (13)$$

Here,  $E_P$  denotes the processing cost, i.e., the energy needed for wake-up, circuitry and sensing.

Due to physical limitations of the radio power amplifiers, power levels are constrained according to

$$0 \leq u_m(k) \leq u_m^{\max}, \quad \forall k \in \mathbb{N}_0, \quad \forall m \in \{1, 2, \dots, M\} \quad (14)$$

for given values  $\{u_m^{\max}\}$ . Thus, the maximum transmission energies per measurement value are given by

$$(E_{\text{TX}}^{\max})_m \triangleq \left(\frac{\tilde{b}_m^{\max}}{r}\right) u_m^{\max}, \quad m \in \{1, 2, \dots, M\}. \quad (15)$$

#### V. STATE ESTIMATION WITH INTERMITTENT SENSOR LINKS

In the present work, we will assume that the data transmitted incorporates error detection coding [24]. Hence, the gateway knows, whether packets received from the sensors contain errors or not. Faulty packets will be discarded when reconstructing the measurement values and when estimating the system state; cf. [25]. Thus, for state estimation purposes, the system amounts to sampling (1) and (2) only at the successful reconstruction instants of each sensor measurement.

To formulate the state estimator, we introduce the  $M$  binary stochastic reconstruction processes

$$\theta_m = \{\theta_m(k)\}_{k \in \mathbb{N}_0}, \quad m \in \{1, 2, \dots, M\}$$

where

$$\theta_m(k) = \begin{cases} 1 & \text{if } \hat{y}_m(k) \text{ can be reconstructed at time } k \\ 0 & \text{otherwise.} \end{cases} \quad (16)$$

Clearly, if at time  $k$ , independent coding is used, then the reconstruction success of  $\hat{y}_m(k)$  depends on the transmission outcome of link  $m$ , i.e., we have

$$\theta_m(k) = \begin{cases} \gamma_m(k) & \text{if } u_m(k) > 0, \\ 0 & \text{if } u_m(k) = 0, \end{cases} \quad \forall m \in \{1, 2, \dots, M\}.$$

On the other hand, if, at time  $k$ , ZEC with dominant coder<sup>6</sup>  $EC_{m^*}$  is used (see Section III-B-2)), then successful reconstruction of  $\hat{y}_m(k)$  also requires that  $\hat{y}_{m^*}(k)$  be successfully reconstructed. Thus, with ZEC, the reconstruction processes  $\theta_m(k)$  are given by

$$\theta_m(k) = \begin{cases} \gamma_m(k)\gamma_{m^*}(k), & \text{if } u_m(k)u_{m^*}(k) > 0, \\ 0, & \text{if } u_m(k)u_{m^*}(k) = 0, \end{cases} \quad (17)$$

for all  $m \in \{1, 2, \dots, M\}$ .

A key point is that the realizations in

$$\theta^k \triangleq \{\theta_m(k - \ell)\}_{\ell \geq 0, m \in \{1, \dots, M\}} \quad (18)$$

are available at the gateway at time  $k$ . Thus, the time-varying Kalman filter (KF) for the system (1) with output matrix

$$C(k) \triangleq \begin{bmatrix} \theta_1(k)C_1 \\ \vdots \\ \theta_M(k)C_M \end{bmatrix} \quad (19)$$

can be used at the gateway. It gives the best linear state estimates. These are given by

$$\hat{x}(k+1) = A\hat{x}(k) + K(k+1)(\hat{y}(k+1) - C(k+1)A\hat{x}(k)) \quad (20)$$

where

$$\begin{aligned} \hat{y}(k+1) &\triangleq [\hat{y}_1(k+1) \quad \hat{y}_2(k+1) \quad \dots \quad \hat{y}_M(k+1)]^T \\ K(k) &\triangleq P(k)C(k)^T (C(k)P(k)C(k)^T + R(k))^{-1} \\ P(k+1) &\triangleq AP(k)A^T + Q - AK(k)C(k)P(k)A^T \\ R(k) &\triangleq \text{diag}(R_1 + D_1(k), \dots, R_M + D_M(k)) \end{aligned} \quad (21)$$

and where  $\{D_m(k)\}$  are the distortions introduced by quantization and, thus, depend upon the bit rate chosen; see (6). The recursion in (21) is initialized with  $P(0) = P_0$  and  $\hat{x}(0) = 0$ .

It is worth emphasizing that  $C(k)$ ,  $K(k)$  and  $P(k)$  are stochastic. Furthermore, the recursion (21) will, in general, not converge to a steady-state value for  $P(k)$ , even if the reconstruction processes  $\{\theta_m(k)\}$  were i.i.d. Bernoulli [26]. However, unlike in the unstable case considered in [26], we here assume that the system (1) is stable. Thus,  $P(k)$  will remain bounded; see also simulations included in Section VIII.

*Example 4 (Scalar System With Two Sensors):* Suppose that  $A \in \mathbb{R}$  and that there are two sensors measuring the system state  $x(k)$  with  $C_1 = C_2 = 1$ . Expression (21) gives that, if

<sup>6</sup>Here, the subscript  $m^*$  is used to indicate that the current signal refers to the dominant coder and associated signals.

$\theta_1(k) = \theta_2(k) = 0$ , then  $K(k) = 0$  so that  $\hat{x}(k) = A\hat{x}(k-1)$ . On the other hand, if  $\theta_1(k) = 1$  and  $\theta_2(k) = 0$ , then

$$\begin{aligned} K(k) &= \begin{bmatrix} \frac{P(k)}{P(k) + R_1 + D_1(k)} & 0 \\ 0 & 0 \end{bmatrix} \\ \hat{x}(k) &= A\hat{x}(k-1) + \frac{P(k)(\hat{y}_1(k) - A\hat{x}(k-1))}{P(k) + R_1 + D_1(k)}. \end{aligned}$$

If  $\theta_1(k) = \theta_2(k) = 1$ , then the filter gain becomes

$$K(k) = \alpha P(k) [R_2 + D_2(k) \quad R_1 + D_1(k)]$$

with

$$\alpha^{-1} \triangleq (P(k) + R_1 + D_1(k))(P(k) + R_2 + D_2(k)) - P(k)^2.$$

Note that in all situations, the Kalman filter uses all successfully reconstructed measurements. In fact, whenever there is information correctly received from the sensors, it is included in the Kalman filter yielding a nonzero filter gain.  $\square$

*Remark 2 (The Gaussian Case):* If the initial state  $x(0)$ , the driving noise  $w$ , the measurement noises  $v_m$  and the quantization noises would all be Gaussian and independent, then the above KF would also provide the conditional mean and covariance

$$\begin{aligned} \hat{x}(k) &\triangleq \mathbb{E}\{x(k) \mid \hat{y}^k, \theta^k\} \\ \bar{P}(k) &\triangleq \mathbb{E}\{(\hat{x}(k) - x(k))(\hat{x}(k) - x(k))^T \mid \hat{y}^k, \theta^k\} \end{aligned}$$

where the expectation  $\mathbb{E}$  is taken with respect to the distributions of  $w, v_1, \dots, v_M, x(0)$ , where the sequence  $\hat{y}^k$  contains all reconstructed measurements of the  $M$  sensors up to time  $k$ , and where

$$\bar{P}(k) = P(k) - K(k)C(k)P(k). \quad (22)$$

Properties of this, and related, state estimators have been studied within the context of state estimation over lossy links with constant dropout probabilities; see, e.g., [26]–[29]. It is worth emphasizing that in the case under study in the present work, transmission probabilities are time varying. Thus, the tools developed in [26]–[29] cannot be applied directly.  $\square$

## VI. PREDICTIVE CONTROL OF SENSOR POWERS AND CODING

In the previous sections, we have shown that the design of sensor powers and bit rates involves a tradeoff between distortion introduced by coding, transmission error probabilities (and, thus, state estimation accuracy), and energy consumption. In what follows, we will present a predictive control strategy which takes into account energy consumption and estimation quality over a future prediction horizon. To keep processing at the sensors to a minimum, the control algorithm is located at the gateway. It provides the power levels and the codebook indexes to be used by the  $M$  sensors.

### A. Constraints

In order to save energy required to process the received signals at the sensors, we would like to keep the signaling from the

gateway to the sensors as low as possible. In particular, the control signal will contain the codebook indices, say

$$j(k) \triangleq [j_1(k) \ j_2(k) \ \dots \ j_M(k)]^T \in \mathcal{J} \quad (23)$$

and information on the power levels. Here, we will use coding ideas frequently used in power control architectures for cellular networks; see, e.g., [30]. The controller, thus, sends coarsely quantized power increments, say  $\delta u_m(k)$ , rather than actual power values,  $u_m(k)$ , to each sensor  $m \in \{1, 2, \dots, M\}$ . All signals  $\{\delta u_m\}$  are constrained according to

$$\delta u_m(k) \in \mathcal{U}_m, \quad \forall k \in \mathbb{N}_0, \quad \forall m \in \{1, 2, \dots, M\} \quad (24)$$

where  $\{\mathcal{U}_m\}$  are given finite sets, each having a small number of elements; see also [31].<sup>7</sup>

Upon reception of the pair  $(\delta u_m(k), j_m(k))$ , each sensor  $m$  chooses the codebook  $j_m(k)$  and reconstructs the power level to be used by its radio power amplifier by setting

$$u_m(k) = u_m(k-1) + \delta u_m(k). \quad (25)$$

Note that in addition to the constraints on actual power levels, see (14), the quantization constraint on the transmitted power control values, see (24), imposes

$$\delta u(k) \in \mathcal{U} \triangleq \mathcal{U}_1 \times \mathcal{U}_2 \times \dots \times \mathcal{U}_M, \quad \forall k \in \mathbb{N}_0 \quad (26)$$

where

$$\delta u(k) \triangleq [\delta u_1(k) \ \dots \ \delta u_M(k)]^T, \quad \forall k \in \mathbb{N}_0. \quad (27)$$

## B. Cost Function

We will quantify estimation accuracy via the trace of the matrix  $\bar{P}(k)$  defined in (22), which, in the Gaussian case, would correspond to the posterior covariance, see Remark 2.

At each time instant  $k \in \mathbb{N}_0$ , the predictive controller calculates the value of  $\bar{P}(k)$ , which results from iterating (20)–(21) for the (known) past realizations of reconstruction outcomes  $\theta^k$ , see (18), and uses channel gain predictions over a finite horizon of fixed length  $N$ . These will be denoted via:

$$\hat{G}(k) = \{\hat{g}_m(k+1|k), \dots, \hat{g}_m(k+N|k)\}_{m \in \{1, 2, \dots, M\}} \quad (28)$$

and are assumed to be error-free. With this information, the controller minimizes the finite-set constrained cost function

$$V(\delta U, J) \triangleq \mathbb{E}\{V_1(\Theta(k), J) | P(k), \hat{G}(k), \delta U, J\} + \rho V_2(\delta U, J) \quad (29)$$

<sup>7</sup>In some wireless sensor technologies, transmission power can be selected only from a finite number of power levels. In these cases, the quantization constraint (24) is a system requirement, see also [32].

where<sup>8</sup>

$$V_1(\Theta(k), J) \triangleq \sum_{\ell=k+1}^{k+N} \text{trace}(\bar{P}'(\ell))$$

$$V_2(\delta U, J) \triangleq \sum_{\ell=k+1}^{k+N} \sum_{m=1}^M E_m(\tilde{b}'_m(\ell) u'_m(\ell)) \quad (30)$$

quantify the expected estimation quality and the associated energy expenditure. Thus,  $\rho \geq 0$  is a design parameter which allows one to trade estimation accuracy for energy consumption.

The stochastic aspect of the power and bit rate control problem, namely the possibility of transmission errors, is captured in (29) by taking expectation with respect to the probability mass distribution of the discrete random variable  $\Theta(k)$ , defined via

$$\Theta(k) \triangleq \begin{bmatrix} \theta'_1(k+1) & \theta'_1(k+2) & \dots & \theta'_1(k+N) \\ \theta'_2(k+1) & \theta'_2(k+2) & \dots & \theta'_2(k+N) \\ \vdots & \vdots & \ddots & \vdots \\ \theta'_M(k+1) & \theta'_M(k+2) & \dots & \theta'_M(k+N) \end{bmatrix}. \quad (31)$$

In this matrix, the entries  $\{\theta'_m(k+\ell)\}$  are predictions of the reconstruction outcomes of the values  $\{\hat{y}_m(k+\ell)\}$ ; see (16). Thus, the probability mass distribution of  $\Theta(k)$  depends upon the tentative power levels  $\{u'_m(k+\ell)\}$ , the tentative future codebook indexes  $\{j'_m(k+\ell)\}$ , and the channel gain predictions  $\{\hat{g}_m(k+\ell|k)\}$ , see (10) and (28).

For a given realization of  $\Theta(k)$ ,  $\text{trace}(\bar{P}'(\ell))$  is obtained from (22) after iterating (20)–(21) with initial value  $P(k+1)$ ;  $E_m(\tilde{b}'_m(\ell) u'_m(\ell))$  is the energy function (13) evaluated for the tentative values  $u'_m(\ell)$  and  $\tilde{b}'_m(\ell)$ , the latter being the expected bit rate of the  $j'_m(\ell)$ th codebook, where

$$j'(\ell) \triangleq [j'_1(\ell) \ j'_2(\ell) \ \dots \ j'_M(\ell)]^T. \quad (32)$$

The decision variables, i.e., the tentative future codebook indexes  $\{j'_m(k+\ell)\}$  and power value increments  $\{\delta u'_m(k+\ell)\}$ , are collected in [see (23) and (27)]:

$$\delta U = \begin{bmatrix} \delta u'(k+1) \\ \delta u'(k+2) \\ \vdots \\ \delta u'(k+N) \end{bmatrix}, \quad J = \begin{bmatrix} j'(k+1) \\ j'(k+2) \\ \vdots \\ j'(k+N) \end{bmatrix}. \quad (33)$$

Following (25),  $\delta U$  yields the tentative future power levels  $\{u'_m(\ell)\}$  in (30) via

$$u'_m(\ell) = u'_m(\ell-1) + \delta u'_m(\ell), \quad \ell \in \{k+1, k+2, \dots, k+N\}$$

starting from the current levels, i.e., where  $u'_m(k) = u_m(k)$ .

<sup>8</sup>In the sequel, primed variables refer to predicted values of the corresponding physical variables.

### C. Moving Horizon Optimization

As described in Section VI-B, at each time instant  $k \in \mathbb{N}_0$ , and given channel gain predictions in (28), the control algorithm finds the optimizing sequences

$$(\delta U^{\text{opt}}, J^{\text{opt}}) \triangleq \arg \min V(\delta U, J) \quad (34)$$

subject to the constraints

$$\begin{aligned} \delta U &\in \mathcal{U}^N, \quad J \in \mathcal{J}^N \\ 0 &\leq u'_m(\ell) \leq u_m^{\text{max}}, \quad \forall \ell \in \{k+1, \dots, k+N\}, \forall m \end{aligned}$$

where

$$\mathcal{U}^N \triangleq \mathcal{U} \times \mathcal{U} \times \dots \times \mathcal{U}, \quad \mathcal{J}^N \triangleq \mathcal{J} \times \mathcal{J} \times \dots \times \mathcal{J}.$$

In this work, we will adopt the moving horizon principle; see, e.g., [15] and [20]. Thus, at each time  $k$ , only the indexes and power updates corresponding to the time instant  $k+1$  are transmitted to the  $M$  wireless sensors. That is, only the following values are used<sup>9</sup>:

$$\begin{aligned} \delta u(k+1)^{\text{opt}} &\triangleq [I_M \quad 0_M \quad \dots \quad 0_M] \delta U^{\text{opt}} \\ j(k+1)^{\text{opt}} &\triangleq [I_M \quad 0_M \quad \dots \quad 0_M] J^{\text{opt}}. \end{aligned}$$

At the next time instant, namely  $k+1$ , the optimization procedure is repeated, giving rise to power control increments  $\delta u(k+2)^{\text{opt}}$  and codebook indexes  $j(k+2)^{\text{opt}}$ . This procedure is repeated *ad infinitum*.

The prediction horizon  $N$  allows the designer to tradeoff performance versus online computational effort. Larger horizons give, in general, better performance since more information is taken into account in the decision process.<sup>10</sup> Computational issues are discussed in Section VII.

The proposed algorithm jointly decides upon the power levels and codebooks of all  $M$  sensors by using future channel gain predictions. The resulting control law respects finite set and magnitude constraints for the power level signaling, see Section VI-A. We emphasize that, for general time-varying channels, the power levels and codebooks thus obtained are not constant, but are assigned dynamically through minimization of the criterion in (29) and, therefore, optimize the resulting performance.

*Remark 3 (Channel Gain Predictors):* To calculate future success probabilities, one requires channel gain predictions, see (28). For that purpose one can use techniques described, e.g., in [33] and [34]. Note that, even if received packets are discarded for signal reconstruction, they may still be used for channel gain estimation and prediction.  $\square$

## VII. COMPUTATIONAL ASPECTS AND SUBOPTIMAL ALGORITHM

The controller presented in the previous section uses a brute-force approach to determine the optimal power increments and codebooks. In fact, all combinations of possible bit-rates (with and without zero-error-coding) and power levels of all sensors

<sup>9</sup> $I_M$  denotes the  $M \times M$  identity matrix and  $0_M$  the all zeros  $M \times M$  matrix.

<sup>10</sup>The effect of  $N$  has been studied in other contexts; see, e.g., [17]–[20].

are examined. Based upon the cost function  $V(\delta U, J)$ , which takes into account state estimation accuracy and energy expenditure, the jointly optimal combination is chosen.

Whilst minimization of  $V(\delta U, J)$  in (34) is carried out at the gateway, where computational issues play less of a role than at the sensors, in some situations, a brute-force search might be impractical due to a large number of possible combinations. In what follows, we will briefly outline some computational aspects and also present a computationally convenient strategy for approximately solving the optimization problem (34).

### A. Computational Issues

Despite the fact that we are dealing with a stochastic nonlinear and partially discrete optimization problem, which has no closed-form solution, solving (34) in real-time is surprisingly simple, due to the binary nature of the matrix of predicted future reconstruction outcomes  $\Theta(k)$ , see (31). To be more precise,  $\Theta(k)$  takes only values in a finite set, say  $\{\Theta_i\}_{i \in \{0, 1, \dots, 2^{MN}-1\}}$ . For a given codebook sequence  $J \in \mathcal{J}^N$ , and current value  $P(k)$ , each realization  $\Theta_i$  corresponds to a particular value  $V_1(\Theta_i, J)$ ; see (30). Consequently, by using the law of total expectation, (29) reduces to evaluating a finite sum, i.e., we have

$$\begin{aligned} &\mathbb{E}\{V_1(\Theta(k), J) \mid P(k), \hat{G}(k), \delta U, J\} \\ &= \sum_{i=0}^{2^{MN}-1} \mathcal{P}_i \mathbb{E}\{V_1(\Theta(k), J) \mid (P(k), \hat{G}(k), \delta U, J), \\ &\quad \Theta(k) = \Theta_i\} \\ &= \sum_{i=0}^{2^{MN}-1} \mathcal{P}_i V_1(\Theta_i, J) \end{aligned}$$

where  $\mathcal{P}_i \triangleq \mathcal{P}\{\Theta(k) = \Theta_i \mid \hat{G}(k), \delta U, J\}$  depends upon future channel gain predictions, power levels and bit rates to be used by the sensors. These probabilities can be calculated by simply forming the product of the individual terms  $\lambda_m(\ell)$ ; see (10). Since the optimization (34) requires evaluating  $V(\delta U, J)$  for at most  $|\mathcal{U}|^N |\mathcal{J}|^N$  possibilities of the pair  $(\delta U, J)$ , the number of calculations to be carried out at each time instant is, at most, proportional to  $(2^M |\mathcal{U}| |\mathcal{J}|)^N$ . Note that, if  $\Theta(k)$  would be a continuous random variable, then the optimization procedure would be more involved; cf. [35].

### B. Suboptimal Algorithm

As seen above, the proposed power and bit rate control algorithm, involves an exhaustive enumeration of a finite set of code books, power value increments and transmission scenarios. Thus, in practice, the algorithm may become too complex to implement. To simplify calculations, one could use approximations based upon convex relaxations, e.g., as described in [36]. Alternatively, the moving horizon optimization idea underlying the controller of Section VI motivates us to formulate a suboptimal algorithm as presented next.

The online computational load can be significantly reduced by limiting the search set for power increments to a subset

<sup>11</sup> $|\mathcal{U}|$  denotes the cardinality of the set  $\mathcal{U}$ ;  $|\mathcal{J}|$  that of  $\mathcal{J}$ .

$\tilde{\mathcal{U}}(k) \subset \mathcal{U}^N$  and by restricting the code book to be constant over the prediction horizon. This leads to suboptimal sequences, say  $\delta U^{\text{sub-opt}}(k)$  and  $J^{\text{sub-opt}}(k)$ ; see (34). A useful choice for  $\tilde{\mathcal{U}}(k)$ , which has been successfully employed in the context of audio quantization [37], can be made by re-using the sequence obtained at the previous step. To be more specific, suppose that, at time  $k-1$ , the suboptimal sequence of increments  $\delta U^{\text{sub-opt}}(k-1)$  has been found and form the *shifted* sequence:

$$\delta U^{\#}(k) = \begin{bmatrix} 0_M & I_M & 0_M & \cdots & \cdots & 0_M \\ \vdots & 0_M & I_M & 0_M & & \vdots \\ \vdots & & 0_M & \ddots & \ddots & \vdots \\ \vdots & & & \ddots & \ddots & 0_M \\ 0_M & \cdots & \cdots & \cdots & 0_M & I_M \end{bmatrix} \times \delta U^{\text{sub-opt}}(k-1).$$

To compute the power increment sequence at time  $k$ , we will make use of the previous predicted values contained in  $\delta U^{\#}(k)$ . More precisely, the reduced search set at time  $k$  is formed via the Cartesian product  $\tilde{\mathcal{U}}(k) = \mathcal{U}^{\#}(k) \times \mathcal{U}$ , where  $\mathcal{U}$  is as in (26) and where  $\mathcal{U}^{\#}(k)$  contains only  $\delta U^{\#}(k)$  and its immediate neighbors. The cardinality of  $\tilde{\mathcal{U}}(k)$  is, at most, equal to  $3^{M(N-1)}|\mathcal{U}|$ . Clearly, in applications where  $|\mathcal{U}| > 3^M$ , the cardinality of  $\tilde{\mathcal{U}}(k)$  is smaller than the cardinality of  $\mathcal{U}^N$ . The number of calculations of this suboptimal algorithm is, at most, proportional to  $3^{M(N-1)}|\mathcal{U}||J|2^{MN}$ .

*Example 5 (Reduced Search Set):* Consider a setting with  $M = 2$  sensors, constraint sets  $\mathcal{U}_1 = \mathcal{U}_2 = \{-4, -3, \dots, 4\}$  and a horizon  $N = 4$ . The optimization in (34) requires the controller to evaluate  $(|\mathcal{U}_1| \cdot |\mathcal{U}_2|)^N = 81^4 = 43046721$  possible power level increments  $\delta U$ .

To illustrate how much can be gained by using the suboptimal algorithm described above, suppose that, at time  $k-1$ , the suboptimal sequence of increments

$$\delta U^{\text{sub-opt}}(k-1) = [2 \ 3 \ 4 \ 2 \ 1 \ 3 \ -1 \ 4]^T$$

has been found. We then have

$$\delta U^{\#}(k) = [4 \ 2 \ 1 \ 3 \ -1 \ 4]^T$$

and

$$\tilde{\mathcal{U}}(k) = \left\{ \{3, 4\} \times \{1, 2, 3\} \right\} \times \left\{ \{0, 1, 2\} \times \{2, 3, 4\} \right\} \\ \times \left\{ \{-2, -1, 0\} \times \{3, 4\} \right\} \times \left\{ \mathcal{U}_1 \times \mathcal{U}_2 \right\}$$

which contains only  $6 \times 9 \times 6 \times 81 = 26244$  elements.  $\square$

## VIII. SIMULATION STUDY

In this section, we will apply the predictive power and coding control algorithm proposed in Section VI to a model (1). The initial state is taken from a Gaussian distribution with covariance matrix  $P_0 = 0.3I_2$ . The driving noise is also Gaussian, with variance  $Q = 1/2I_2$ . The system matrix is chosen as

$$A = \begin{bmatrix} 1.6718 & -0.9048 \\ 1 & 0 \end{bmatrix}.$$

### A. Experimental Setup

We simulate a system consisting of  $M = 2$  sensors with output matrices  $C_1 = [1 \ 0]$ ,  $C_2 = [0 \ 1]$ , and Gaussian measurement noises with variances  $R_1 = R_2 = 1/100$ . The constraints on the power values are  $u_1^{\max} = u_2^{\max} = 0.4$  mW. Bit rates are restricted to a maximum of  $\tilde{b}_1^{\max} = \tilde{b}_2^{\max} = 7$  bits.

Channel data were acquired in an office space area at the Signals and Systems Group of Uppsala University, Uppsala, Sweden. The transmitter was placed in an office and the receiver was located 8 m away in the corridor outside the office. The transmitter position was fixed, whereas the receiver was mounted on a rail allowed to move over a distance of 1.25 m, perpendicular to the corridor. Measurements were collected at the 2.4 GHz ISM band. The top diagram of Fig. 4 illustrates the channel gains of two realizations, one with horizontal and one with vertical polarization.<sup>12</sup> The channel gains are displayed as a function of relative position, and, therefore, correspond to what a receiver would perceive if it moved through space. The gains shown in Fig. 4 also give an indication of what fading to expect, if objects are moving between fixed sensor and gateway locations. We note that the channels vary considerably, with some dips dropping more than 20 dB. Without appropriate dynamic power control, such large variations would require excessive energy consumption to avoid frequent transmission errors.

### B. Controller Design

To implement the control algorithm proposed in Section VI we first construct the zero-error coder (along the lines described in the Appendix) to obtain Table II. The controller parameters are chosen as  $N = 1$  and  $\rho = 10^6$  and we adopt the simple expression (12) for the probability of successful packet arrival.<sup>13</sup> Power increments are restricted to belong to the finite sets  $\mathcal{U}_1 = \mathcal{U}_2 = \{0, \pm 10 \mu\text{W}\}$ . To emulate a realistic scenario, we will assume that the gateway is using only noisy one-step-ahead channel predictions.<sup>14</sup>

### C. Results

Fig. 3 illustrates the state estimation performance when  $g_1(k)$  enters a deep fade located at  $k = 3950$ . See the top diagram of Fig. 4. From the top diagram of Fig. 3 we note that  $x_1$  and  $\hat{x}_1$  essentially coincide everywhere except for the area around the fading dip. Because of the poor channel gain in  $g_2(k)$ , the predictive controller increases the control action and decreases the bit rate. Since the control signal saturates, packet errors will occur and affect the estimation accuracy. This is further illustrated in the bottom diagram of Fig. 3, where estimated and

<sup>12</sup>The channels were found to be well described by time-varying gains. Here we have assumed a radio power amplifier efficiency of 1/3, a path loss of 90 dB, and a receiver noise amplification (noise figure) of 10 dB, yielding an average gain for  $g_1(k)$  and  $g_2(k)$  of  $-105$  dB. At maximum power, this corresponds to an SNR of 17 dB. It is worth emphasizing that, since we have real channel data, we do not require to ascribe probability distributions to the fading. However, information about the underlying fading statistics, can be incorporated into the formulation, e.g., by averaging (11) over the fading distribution.

<sup>13</sup>We choose  $r k_B T = 2.3 \cdot 10^{-10} \mu\text{W}$ , which corresponds to a channel bit rate  $r = 40$  kbits/s at room temperature.

<sup>14</sup>Here a normalized prediction error MSE of 0.01 is used. This corresponds to a prediction horizon of approximately 1/4 of a wavelength of the carrier frequency [33], [34].



TABLE I  
PERFORMANCE COMPARISON BETWEEN PROPOSED VARIABLE BIT RATE PREDICTIVE CONTROLLER, FIXED RATE PREDICTIVE CONTROLLERS, AND FIXED LEVEL CONTROLLERS. FIXED LEVEL CONTROLLERS ARE KEPT AT THE SAME AVERAGE ENERGY AS THE VARIABLE LEVEL PREDICTIVE CONTROLLER

Bit rate	Predictive Controllers			Fixed level Controllers		
	$V_1$	$V_2$ nJ	$V$	$V_1$	$V_2$ nJ	$V$
variable	<b>0.0291</b>	<b>60.3265</b>	<b>0.0894</b>	-	-	-
7 bits	0.0339	64.5253	0.0984	0.0686	60.3265	0.1289
6.5 bits	0.0350	62.4227	0.0974	0.0679	60.3265	0.1282
6 bits	0.0370	60.3442	0.0973	0.0649	60.3265	0.1252
5 bits	0.0548	56.2098	0.1110	0.0778	60.3265	0.1382
4 bits	0.1257	52.1200	0.1778	0.1455	60.3265	0.2058

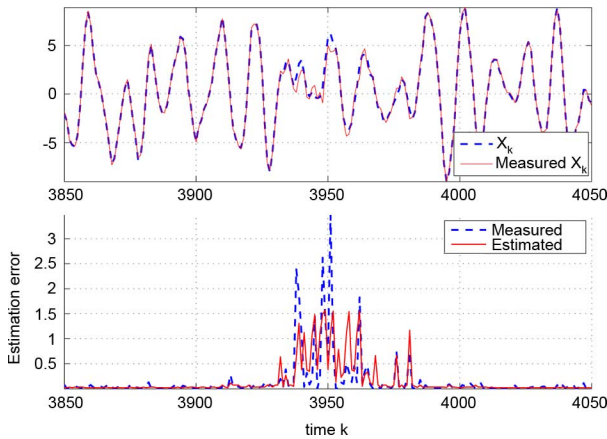


Fig. 3. State estimation with power and bit rate constraints: Upper plot  $[1 \ 0]x(k)$  (dashed line) and  $[1 \ 0]\hat{x}(k)$  (solid line); lower plot:  $\|x(k) - \hat{x}(k)\|^2$  (dashed line) and  $\text{trace}(\hat{P}(k))$  (solid line).

measured estimation errors  $\|x(k) - \hat{x}(k)\|^2$  are compared. Evidently, the measured estimation error increases due to the packet errors, cf., the bottom diagram of Fig. 4. The estimated state estimation error also increases, but not to the same extent. This is due to the smoothing effect built into  $\text{trace}(\hat{P}(k))$ .

Fig. 4 illustrates that the control algorithm tries to find the best compromise between the two sensor links by manipulating power levels, bit rates and by using ZEC whenever appropriate. Because of the constraints in the power levels and bit rates, the controller cannot fully compensate for the large variability of the channel gains. This is apparent by inspecting the performance around time  $k = 900$ . Here, the predictive controller increases the power  $u_2$  to compensate for the dropping channel gain  $g_2$ . Since  $u_2$  saturates, the increased cost due to maximum power use is balanced by decreasing the bit rate  $\tilde{b}_2$  (less bits per measurement value require less energy). As is evident from the low probability of successful packet arrival,  $\lambda_2$ , the controller does not fully counteract the fast drop in  $g_2$ . Since  $g_1(900) \gg g_2(900)$ , the controller decides that Sensor 1 should spend only little power and also selects Sensor 1 to be dominant in the ZEC scheme. A similar behavior can be observed for  $g_1$  at  $k = 1250$ , and for  $g_2$  at  $k = 4000$  and  $k = 4700$ , respectively.

It is interesting to note that independent coding, which is indicated by dots on top of both bit rate graphs  $\tilde{b}_1$  and  $\tilde{b}_2$  of Fig. 4 simultaneously, is rarely used. Thus, there is significant

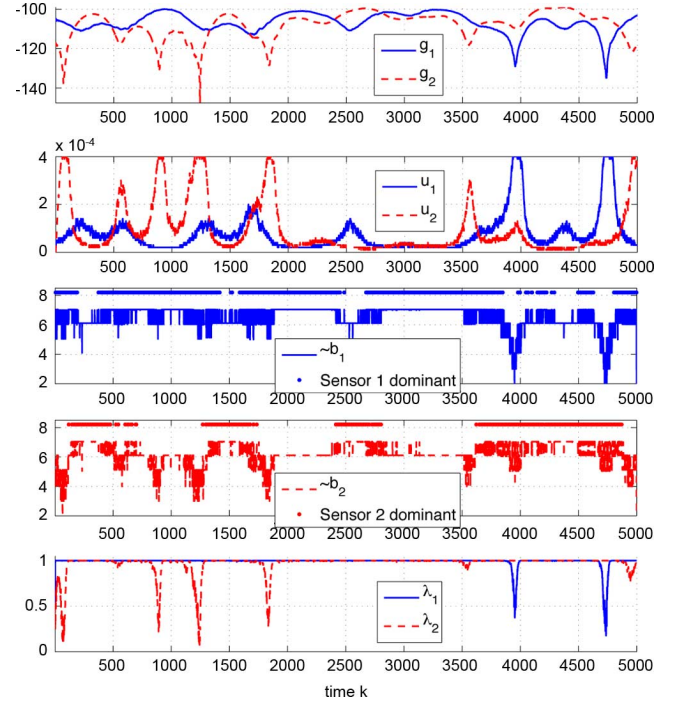


Fig. 4. System performance with power and bit rate constraints.

redundancy to be exploited among the two channels.<sup>15</sup> At time  $k = 2700$ , both channels have high gains and the power levels can be kept low while maintaining  $\lambda_1$  and  $\lambda_2$  close to one. Instead, energy is spent on decreasing the quantization error by using the maximum allowable bit rate.

To give further insight, the performance of the variable bit rate control algorithm presented in Section VI is compared to predictive controllers which use only independent coding at a fixed bit rate<sup>16</sup> and also to schemes which use fixed power levels at all times. The fixed level controllers are designed to use the same amount of average energy  $V_2$  as does the variable bit rate predictive control algorithm proposed in this work.

We shall evaluate the performance of the different controllers according to the *achieved cost*:

$$V = V_1 + V_2 \quad (35)$$

where

$$V_1 \triangleq \frac{1}{5000} \sum_{k=1}^{5000} \|x(k) - \hat{x}(k)\|^2$$

quantifies the estimation accuracy, whereas

$$V_2 \triangleq \rho \left( \frac{1}{5000} \sum_{k=1}^{5000} \sum_{m=1}^M E_m(\tilde{b}_m(k)u_m(k)) \right)$$

refers to the energy use. The results are summarized in Table I. We observe that both the fixed bit rate predictive controllers and the fixed level controllers obtain their lowest total cost  $V$  [see (35)] for a fixed rate of six bits. For this bit rate, the fixed

<sup>15</sup>Indeed, this is to be expected since  $x_1(k)$  depends on  $x_2(k)$ .

<sup>16</sup>These predictive controllers essentially amount to the algorithms introduced in our recent work [21].

rate predictive controller, and the fixed level controllers are 9% and 40% worse, respectively, as compared to the variable bit rate predictive controller proposed in the present work. Considering  $V_1$  only, for this case, these simpler schemes give, 27% and 123% worse performance, respectively.<sup>17</sup>

## IX. CONCLUSION

An energy efficient bit rate and power control scheme for state estimation via wireless sensors operating over fading channels was presented. The time variability of the fading channels frequently leads to transmission errors with subsequent random packet drops. State estimation in the face of intermittent observations was then performed by a time-varying Kalman filter.

In our scheme, the transmission power of the radio power amplifier and the bit rate for each of the wireless sensors are controlled by the gateway. For that purpose, we proposed a control algorithm which utilizes predictions of radio channel gains and incorporates signal coding. Thus, our proposal amounts to channel aware source coding. A distinctive characteristic of the approach taken is that the controller directly aims at optimizing the tradeoff between estimation quality and energy use. Performance was illustrated on measured channel data acquired from an office space area. The optimal scheme was shown to perform significantly better when compared to predictive controllers which use only independent coding at fixed bit rates, and to fixed power level controllers.

Further work may include studying closed loop stability aspects and also the examination of more general wireless sensor network topologies.

## APPENDIX

### BACKGROUND ON ZERO-ERROR CODING

One of the key results in distributed source coding is that there is essentially no loss in coding rate-distortion performance by doing separate encoding, rather than joint encoding of correlated variables, when the decoding is done jointly [11], [12]. However, existing practical distributed coding schemes, which come close to the optimal joint performance bounds, require forming long sequences of the variables, thus, incurring large decoding delays. Fortunately, there exist practical low delay distributed coding schemes which, at the expense of a somewhat larger rate, work for arbitrary block lengths, including blocks having only one value. This is the case in ZEC schemes, which are adopted in the present work. We next recall some results on bit rate reducing properties of ZEC.

*Bit Rates:* If coder  $EC_{m^*}$  is dominant, then the expected bit rate (as seen by the channel) is given by  $\tilde{b}_{m^*}(k) = H(i_{m^*}(k))$ . On the other hand,  $\tilde{b}_n(k)$ ,  $n \neq m^*$  is lower bounded by the conditional entropy

$$\tilde{b}_n(k) \geq H(i_n(k)|i_{m^*}(k)) \quad (36)$$

where equality can be achieved at the expense of large delays at the decoder.

<sup>17</sup>The optimum bit rate and power levels, used here for comparison, are in general unknown *a priori*. Thus, in a real application, significantly worse performance for the fixed bit rate predictive controller and the fixed level controller can be expected.

Since, for independent entropy coding, we have [see (3)]

$$b_n(k) \approx H(i_n(k)) \geq H(i_n(k)|i_{m^*}(k)),$$

it follows that we can expect a bit rate reduction which is, at most

$$H(i_n(k)) - H(i_n(k)|i_{m^*}(k)) \geq 0$$

bits per sample. Thus, if the measurements are uncorrelated (and thereby independent since they are Gaussian), then we have  $H(i_n(k)|i_{m^*}(k)) = H(i_n(k))$  and no rate reduction is possible. At the other extreme, if  $i_{m^*}(k)$  is deterministically related to  $i_n(k)$ , then  $H(i_n(k)|i_{m^*}(k)) = 0$ , which gives maximum bit rate reduction.

Note that, in the present case, the amount of correlation between the measurements depends upon the underlying system (1) as well as the covariances  $Q$  and  $R_m$ . Furthermore, the rate reduction achieved in practice, depends not only on the inherent correlation between the measurements, but also on the actual implementation of the zero-error code.

*Design Example:* Zero-error coders can be designed by adopting graph theoretic ideas [14]. To present this approach, let  $\mathcal{Y}_i$  denote the discrete alphabet of  $\hat{y}_i(k)$  and, for simplicity, let us consider the case of only two sensors, i.e.,  $i \in \{1, 2\}$ , where the second sensor is the dominant one. Let the pair of random variables  $(\hat{y}_1, \hat{y}_2)$  be distributed over the product set  $\mathcal{Y}_1 \times \mathcal{Y}_2$  according to the joint probability distribution  $\mathcal{P}(\hat{y}_1, \hat{y}_2)$ . Let  $\mathcal{S}$  be the support set of  $(\hat{y}_1, \hat{y}_2)$ , i.e.,

$$\mathcal{S} \triangleq \{(\hat{y}_1, \hat{y}_2) \in \mathcal{Y}_1 \times \mathcal{Y}_2 : \mathcal{P}(\hat{y}_1, \hat{y}_2) > 0\}. \quad (37)$$

Given the above, we say that distinct  $\hat{y}_1, \hat{y}'_1 \in \mathcal{Y}_1$  are *confusable* if there is a  $\hat{y}_2 \in \mathcal{Y}_2$  such that  $(\hat{y}_1, \hat{y}_2) \in \mathcal{S}$  and  $(\hat{y}'_1, \hat{y}_2) \in \mathcal{S}$ . Viewed from this perspective, a zero-error code is a deterministic map  $\phi$  such that if  $\hat{y}_1$  and  $\hat{y}'_1$  are confusable then  $\phi(\hat{y}_1) \neq \phi(\hat{y}'_1)$ ; see, e.g., [38].

Let  $\mathcal{G}$  be the characteristic graph of the source pair  $(\hat{y}_1, \hat{y}_2)$ . For the case of  $\hat{y}_2$  being the dominant signal, the vertex set of  $\mathcal{G}$  is  $\mathcal{Y}_1$ . Two distinct vertices  $\hat{y}_1$  and  $\hat{y}'_1$  are connected if they are confusable. Zero-error coder design amounts to choosing a graph coloring on  $\mathcal{G}$ , which is a function that partitions  $\mathcal{Y}_1$  into color classes so that every element of  $\mathcal{Y}_1$  has a color and no pair of confusable elements has the same color. It follows that, if  $\hat{y}_1 \in \mathcal{Y}_1$  has color  $c$ ,  $\hat{y}_2 \in \mathcal{Y}_2$  and  $(\hat{y}_1, \hat{y}_2) \in \mathcal{S}$ , then  $(\hat{y}'_1, \hat{y}_2) \notin \mathcal{S}$  for all  $\hat{y}'_1 \in \mathcal{Y}_1, \hat{y}'_1 \neq \hat{y}_1$ . In other words, if  $\hat{y}_2$  is known at the gateway, then we only have to convey information about the color of  $\hat{y}_1$  in order for the decoder to be able to faithfully reconstruct  $\hat{y}_1$ .

To design the zero-error coder used in Section VIII, we first calculate the differential entropies  $h(y_1) = 4.27$  and  $h(y_2) = 4.29$ . The resulting zero-error code depends upon which of the two variables we choose to be the dominating one. It also depends upon the bit rates of the two scalar quantizers. For this example, we choose  $y_2$  to be the dominating one and bit rates  $b_1 = 2$  and  $b_2 = 3$  bits/dim. With this choice, it follows that the step sizes are  $\Delta_1 = 4.82$  and  $\Delta_2 = 2.44$ , respectively.

We now independently scalar quantize  $y_1$  and  $y_2$  to obtain  $\hat{y}_1$  and  $\hat{y}_2$ . Even though the processes have unbounded support, in practice they will always have bounded support. In our case, it

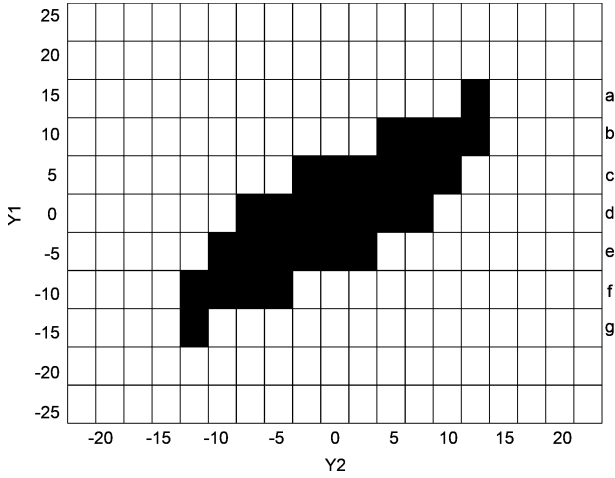


Fig. 5. The discrete alphabet  $\mathcal{Y}_1$  of  $\hat{y}_1$  contains 11 elements (rows) and the reduced set  $\mathcal{Y}'_1$  (solid region) contains only seven elements (rows). Similarly,  $\mathcal{Y}_2$  contains 19 elements (columns) and  $\mathcal{Y}'_2$  has 11 elements (columns).

turns out that  $\hat{y}_1$  is discretized to a finite alphabet  $\mathcal{Y}_1$  having 11 distinct elements whereas the alphabet  $\mathcal{Y}_2$  of  $\hat{y}_2$  has 19 distinct elements. (Recall that a zero-error code is defined on all pairs of elements of  $\mathcal{Y}_1 \times \mathcal{Y}_2$  where  $P(\hat{y}_1, \hat{y}_2) > 0$ .) The cardinality of the zero-error code in this example is therefore  $11 \cdot 19 = 209$ . However, several of the pairs  $(\hat{y}_1, \hat{y}_2) \in \mathcal{Y}_1 \times \mathcal{Y}_2$  are very unlikely. Thus, we can significantly reduce the number of pairs (and thereby the bit rate) by excluding unlikely pairs. In this simulation, we choose to keep only pairs where

$$[\hat{y}_1, \hat{y}_2] [\text{cov}(y_1, y_2)]^{-1} [\hat{y}_1, \hat{y}_2]^T \leq k_\epsilon \quad (38)$$

where we set  $k_\epsilon = 10$ . It can be shown that the probability of being inside this reduced set is given by  $1 - \exp(-k_\epsilon/2) \approx 0.993$ ; see [39].

In Fig. 5, we have shaded the region (solid rectangles) of the joint probability mass function of  $\hat{y}_1$  and  $\hat{y}_2$  containing the 29 out of the 209 pairs that satisfy (38). Since only 7 rows and 11 columns contain solid rectangles, it follows that seven distinct values of  $\hat{y}_1$  and 11 distinct values of  $\hat{y}_2$  need to be used.

Let  $\mathcal{Y}'_1$  and  $\mathcal{Y}'_2$  denote the reduced alphabets containing only the elements which are in pairs that occur inside the solid region of Fig. 5. We now obtain the characteristic graph  $\mathcal{G}$  with vertex set  $\mathcal{Y}'_1$ . Recall that vertices in this graph are connected, if they are confusable. Thus, the elements  $\hat{y}_1$  and  $\hat{y}'_1$  both in  $\mathcal{Y}'_1$  are connected, if they occur in pairs  $(\hat{y}_1, \hat{y}_2)$  and  $(\hat{y}'_1, \hat{y}_2)$  where  $\hat{y}_2 \in \mathcal{Y}'_2$ , i.e., if they share a node of  $\mathcal{Y}'_2$ . For example, referring to Fig. 5, the vertex set of  $\mathcal{Y}'_1$  consists of the seven rows, here denoted a to g. To obtain the characteristic graph  $\mathcal{G}$  we need seven nodes, which we label from 1 to 7, respectively. This is illustrated in Fig. 6(a), where Row a corresponds to Node 1, Row b to Node 2, and so forth. Row a is connected to Row b, since they share columns. Thus, a line is drawn between Node 1 and Node 2 in Fig. 6(a). Similarly, since Row b is connected to both Row c, and Row d, via overlapping columns, we draw lines from Node 2 to nodes 3 and 4, respectively. By repeating this procedure until we reach Row g, the graph in Fig. 6(a) is readily obtained.

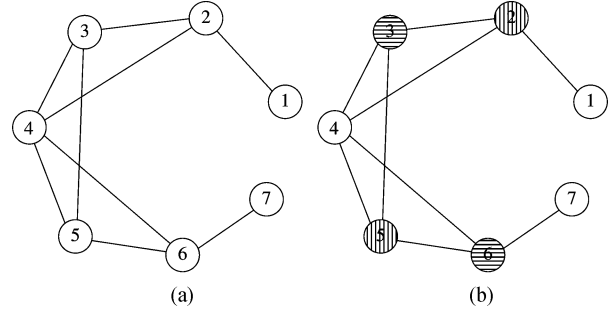


Fig. 6. Design of zero-error coding: (a) Characteristic graph  $\mathcal{G}$  and (b) graph coloring.

TABLE II  
BIT RATES WITH  $y_2$  DOMINANT.  $b_i$  IS THE INITIAL BIT RATE,  $H(\hat{y}_i)$  IS THE MEASURED DISCRETE ENTROPY, AND  $\tilde{b}_i$  IS THE MEASURED RATE AFTER APPLYING AN ENTROPY CODER

$b_1$	$H(\hat{y}_1)$	$\tilde{b}_1$	$b_2$	$H(\hat{y}'_2)$	$\tilde{b}_2$
2	1.54	1.55	2	2.06	2.17
3	2.28	2.31	2	2.06	2.17
3	2.31	2.36	3	3.02	3.08
4	3.14	3.15	2	2.06	2.17
4	3.15	3.17	3	3.02	3.08
4	3.16	3.18	4	4.00	4.05
5	4.05	4.06	2	2.06	2.17
5	4.06	4.07	3	3.02	3.08
5	4.08	4.09	4	4.00	4.05
5	4.08	4.09	5	5.00	5.04
6	5.10	5.11	2	2.06	2.17
6	5.10	5.12	3	3.02	3.08
6	5.11	5.12	4	4.00	4.05
6	5.11	5.13	5	5.00	5.04
6	5.11	5.12	6	6.00	6.04
7	6.08	6.09	2	2.06	2.17
7	6.09	6.11	3	3.02	3.08
7	6.10	6.11	4	4.00	4.05
7	6.10	6.11	5	5.00	5.04
7	6.10	6.11	6	6.00	6.04
7	6.10	6.11	7	7.00	7.04

If two nodes in the characteristic graph are not connected, it means that the two nodes are paired with different values of  $\hat{y}_2$ . This is an important observation. Clearly, given a specific  $\hat{y}_2$ , we know the set of  $\hat{y}_1$ 's that are paired with this element and therefore we also know those that are not paired with the given element. Thus, nodes that are not connected in the characteristic graph, may be combined into a *super-node*. For example, in Fig. 6(a) we may combine nodes 1, 4, and 7 into a single super-node. Similarly, nodes 2 and 5 can be combined into a super-node and nodes 3 and 6 may be combined into a super-node. This is illustrated in Fig. 6(b).

The problem described above is usually referred to as graph coloring [14]. In graph coloring, we aim at using the least number of colors so that all nodes in the graph are colored, but two nodes that are connected do not get the same color. Graph coloring is an NP hard problem, but there exist several heuristic methods that, for our purpose, yield near optimal performance. It should be clear that a graph coloring on the characteristic graph of  $\mathcal{Y}'_1$  effectively reduces the number of distinct elements of  $\mathcal{Y}'_1$  that need to be transmitted to the decoder. In fact, the number of elements to be transmitted is equal to the number of colors in the graph. This is so, since, at the decoder, having received a color as well as any  $\hat{y}_2 \in \mathcal{Y}'_2$ , one can uniquely determine the correct  $\hat{y}_1$  among the set of  $\hat{y}_1$ 's having the same

color. In the particular case depicted in Fig. 6(b), only three colors are required.

So far we have been able to reduce the bit rate by only keeping the most likely outcomes  $(\hat{y}_1, \hat{y}_2)$  from the quantizers and then using graph coloring. The bit rate can be reduced further by employing two independent Huffman entropy codes; one for the set of colors and one for the elements of  $\mathcal{Y}_2$ . These codes are designed offline for every pair of bit rates  $(b_1, b_2) \in \{2, \dots, 7\} \times \{2, \dots, 7\}$  as well as when  $y_1$  is dominating instead of  $y_2$ . The resulting numerically measured average bit rates give rise to Table II.

In the case of more than two sensors, one can either pair sensors and use the above approach independently for each pair, or also let several sensors be dependent upon a single sensor. To the best of the authors' knowledge, how to design efficient zero-error codes, in general cases, for more than two sensors remains an open problem.

## REFERENCES

- [1] J. P. Hespanha, P. Naghshtabrizi, and Y. Xu, "A survey of recent results in networked control systems," *Proc. IEEE*, vol. 1, pp. 138–162, Jan. 2007.
- [2] G. N. Nair, F. Fagnani, S. Zampieri, and R. J. Evans, "Feedback control under data rate constraints: An overview," *Proc. IEEE*, vol. 95, pp. 108–137, Jan. 2007.
- [3] L. Schenato, B. Sinopoli, M. Franceschetti, K. Poolla, and S. S. Sastry, "Foundations of control and estimation over lossy networks," *Proc. IEEE*, vol. 95, pp. 163–187, Jan. 2007.
- [4] M. Ilyas, I. Mahgoub, and L. Kelly, *Handbook of Sensor Networks: Compact Wireless and Wired Sensing Systems*. Boca Raton, FL: CRC Press, 2004.
- [5] X. Shen, Q. Zhang, and R. C. Qiu, "Wireless sensor networking (Guest Editorial)," *IEEE Wireless Commun.*, vol. 14, pp. 4–5, Dec. 2007.
- [6] H. Gharavi and S. P. Kumar, *Proc. IEEE (Special Section on Sensor Networks and Applications)*, vol. 91, Aug. 2003.
- [7] N. A. Pantazis and D. D. Vergados, "A survey on power control issues in wireless sensor networks," *Commun. Surveys Tuts.*, vol. 9, no. 4, pp. 86–107, 2007.
- [8] J.-J. Xiao, S. Cui, Z.-Q. Luo, and A. J. Goldsmith, "Linear coherent decentralized estimation," *IEEE Trans. Signal Process.*, vol. 56, no. 2, pp. 757–770, Feb. 2008.
- [9] E. Björnemo, A. Ahlén, and M. Johansson, "On the energy-efficiency of cooperative MIMO in Nakagami fading wireless sensor networks," in *41st Asilomar Conf. Signals, Systems, Computers*, Monterey, CA, 2007.
- [10] M. Johansson, E. Björnemo, and A. Ahlén, "Fixed link margins outperform power control in energy-limited wireless sensor networks," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process.*, Honolulu, HI, 2007, vol. 3, pp. 513–516.
- [11] D. Slepian and J. K. Wolf, "Noiseless coding of correlated information sources," *IEEE Trans. Inf. Theory*, vol. 19, pp. 471–480, Jul. 1973.
- [12] A. D. Wyner and J. Ziv, "The rate-distortion function for source coding with side information at the decoder," *IEEE Trans. Inf. Theory*, vol. 22, pp. 1–10, Jan. 1976.
- [13] T. M. Cover and J. A. Thomas, *Elements of Information Theory*, 2nd ed. New York: Wiley-Interscience, 2006.
- [14] P. Koulgi, E. Tuncel, S. L. Regunathan, and K. Rose, "On zero-error coding of correlated sources," *IEEE Trans. Inf. Theory*, vol. 49, pp. 2856–2873, Nov. 2003.
- [15] D. Q. Mayne, J. B. Rawlings, C. V. Rao, and P. O. M. Scokaert, "Constrained model predictive control: Optimality and stability," *Automatica*, vol. 36, pp. 789–814, Jun. 2000.
- [16] G. C. Goodwin, J. Østergaard, D. E. Quevedo, and A. Feuer, "A vector quantization approach to scenario generation for stochastic NMPC," in *Nonlinear Model Predictive Control: Towards New Challenging Applications*, of LNCIS, L. Magni, D. M. Raimondo, and F. Allgöwer, Eds. Berlin, Germany: Springer-Verlag, 2009, vol. 384, pp. 235–248.
- [17] D. E. Quevedo and G. C. Goodwin, "Multistep optimal analog-to-digital conversion," *IEEE Trans. Circuits Syst. I*, vol. 52, pp. 503–515, Mar. 2005.
- [18] D. E. Quevedo, G. C. Goodwin, and J. A. De Doná, "Multistep detector for linear ISI-channels incorporating degrees of belief in past estimates," *IEEE Trans. Commun.*, vol. 55, pp. 2092–2103, Nov. 2007.
- [19] D. E. Quevedo and G. C. Goodwin, "Moving horizon design of discrete coefficient FIR filters," *IEEE Trans. Signal Process.*, vol. 53, no. 6, pp. 2262–2267, Jun. 2005.
- [20] D. E. Quevedo, H. Bölcskei, and G. C. Goodwin, "Quantization of filter bank frame expansions through moving horizon optimization," *IEEE Trans. Signal Process.*, vol. 57, pp. 503–515, Feb. 2009.
- [21] D. E. Quevedo and A. Ahlén, "A predictive power control scheme for energy efficient state estimation via wireless sensor networks," in *Proc. IEEE Conf. Decision Control*, Cancún, México, Dec. 2008.
- [22] D. E. Quevedo, A. Ahlén, and G. C. Goodwin, "Predictive power control of wireless sensor networks for closed loop control," in *Nonlinear Model Predictive Control: Towards New Challenging Applications of LNCIS*, L. Magni, D. M. Raimondo, and F. Allgöwer, Eds. Berlin, Germany: Springer-Verlag, 2009, vol. 384, pp. 215–224.
- [23] A. Gersho and R. M. Gray, *Vector Quantization and Signal Compression*. Boston, MA: Kluwer Academic, 1992.
- [24] J. G. Proakis, *Digital Communications*, 3rd ed. New York: McGraw-Hill, 1995.
- [25] Y. Mostofi and R. M. Murray, "To drop or not to drop: Design principles for Kalman filtering over wireless fading channels," *IEEE Trans. Autom. Control*, vol. 54, pp. 376–381, Feb. 2009.
- [26] 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, pp. 1453–1464, Sep. 2004.
- [27] M. Huang and S. Dey, "Stability of Kalman filtering with Markovian packet losses," *Automatica*, vol. 43, pp. 598–607, 2007.
- [28] X. Liu and A. Goldsmith, "Kalman filtering with partial observation losses," in *Proc. IEEE Conf. Decision Control*, Paradise Island, Bahamas, 2004, pp. 4180–4186.
- [29] K. Plarre and F. Bullo, "On Kalman filtering for detectable systems with intermittent observations," *IEEE Trans. Autom. Control*, vol. 54, pp. 386–390, Feb. 2009.
- [30] *WCDMA for UMTS*, H. Holma and A. Toskala, Eds., 3rd ed. West Sussex, U.K.: Wiley, 2004.
- [31] D. E. Quevedo, G. C. Goodwin, and J. A. De Doná, "Finite constraint set receding horizon quadratic control," *Int. J. Robust Nonlin. Control*, vol. 14, pp. 355–377, Mar. 2004.
- [32] A. M. M. Alavi, M. J. Walsh, and M. J. Hayes, "Robust distributed active power control technique for IEEE 802.15.4 wireless sensor networks—A quantitative feedback theory approach," *Control Eng. Pract.*, vol. 17, pp. 805–814, 2009.
- [33] T. Ekman, M. Sternad, and A. Ahlén, "Unbiased power prediction of Rayleigh fading channels," in *Proc. Vehicular Technology Conf.*, 2002, vol. 1, pp. 280–284.
- [34] T. Ekman, "Prediction of mobile radio channels: Modeling and design." Ph.D. dissertation, Uppsala Univ., Uppsala, Sweden, 2002.
- [35] R. Mirkov and G. C. Pflug, "Tree approximations of stochastic dynamic programs," *SIAM J. Optim.*, vol. 18, no. 3, pp. 1082–1105, 2007.
- [36] S. Joshi and S. Boyd, "Sensor selection via convex optimization," *IEEE Trans. Signal Process.*, vol. 57, no. 2, pp. 451–462, Feb. 2009.
- [37] D. E. Quevedo and G. C. Goodwin, "Audio quantization from a receding horizon control perspective," in *Proc. Amer. Control Conf.*, 2003, pp. 4131–4136.
- [38] N. Alon and A. Orlitsky, "Source coding and graph entropies," *IEEE Trans. Inf. Theory*, vol. 42, pp. 1329–1339, Sep. 1996.
- [39] R. C. Smith and P. Cheeseman, "On the representation and estimation of spatial uncertainty," *Int. J. Robotics Res.*, vol. 5, no. 4, pp. 56–68, 1986.



**Daniel E. Quevedo** (S'97–M'05) received the Ingeniero Civil Electrónico and Magister en Ingeniería Electrónica degrees from the Universidad Técnica Federico Santa María, Valparaíso, Chile, in 2000 and the Ph.D. degree from the University of Newcastle, Australia, in 2005.

He is currently a Research Academic at the University of Newcastle. He has been a visiting researcher at ETH Zürich, Switzerland, at Uppsala University, Sweden, at The University of Melbourne, Australia, at Aalborg University, Denmark, and at Kyoto University, Japan. His research interests cover several areas of automatic control, signal processing, communications, and power electronics.

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 an Australian Research Fellowship.



**Anders Ahlén** (S'80–M'84–SM'90) was born in Kalmar, Sweden. He received the Ph.D. degree in automatic control from Uppsala University, Uppsala, Sweden.

He was with the Systems and Control Group, Uppsala University, from 1984 to 1992 as an Assistant and Associate Professor in automatic control. In 1992, he was appointed Associate Professor of Signal Processing at Uppsala University. He currently is a Full Professor and holds the Chair in Signal Processing at Uppsala University, where he

is also the head of the Signals and Systems Division of The Department of Engineering Sciences. 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. From 2001 to 2004 he was the CEO of Dirac Research AB, a company offering state-of-the-art audio signal processing solutions, where he is currently the Chairman of the Board. His research interests, which include signal processing, communications and control, currently focus on signal processing for wireless communications, wireless systems beyond 3G, wireless sensor networks, wireless control, and audio signal processing.

Dr. Ahlén was the Editor of Signal and Modulation Design for the IEEE TRANSACTIONS ON COMMUNICATIONS from 1998 to 2004.



**Jan Østergaard** (S'98–M'99) received the M.Sc. degree in electrical engineering from Aalborg University, Aalborg, Denmark, in 1999 and the Ph.D. degree (*cum laude*) in electrical engineering from the Delft University of Technology, Delft, The Netherlands, in 2007.

From 1999 to 2002, he worked as an R&D Engineer at ETI A/S, Aalborg, Denmark, and from 2002 to 2003, he worked as an R&D Engineer at ETI, Inc., Virginia. Between September 2007 and June 2008, he worked as a Postdoctoral Researcher in the Centre

for Complex Dynamic Systems and Control, School of Electrical Engineering and Computer Science, The University of Newcastle, NSW, Australia. He has also been a visiting researcher at Tel Aviv University, Tel Aviv, Israel, and at Universidad Técnica Federico Santa María, Valparaíso, Chile. He is currently a Postdoctoral Researcher at Aalborg University, Aalborg, Denmark.

Dr. Østergaard has received a Danish Independent Research Council's Young Researcher's Award and a fellowship from the Danish Research Council for Technology and Production Sciences.