Available online at www.sciencedirect.com
SCIENCE DIRECT
automatica
Automatica 42 (2006) 251–260
www.elsevier.com/locate/automatica

On a stochastic sensor selection algorithm with applications in sensor scheduling and sensor coverage☆

Vijay Gupta*, Timothy H. Chung, Babak Hassibi, Richard M. Murray

Division of Engineering and Applied Science, California Institute of Technology, Pasadena, USA

Received 26 September 2004; accepted 18 September 2005
Available online 2 December 2005

Abstract

In this note we consider the following problem. Suppose a set of sensors is jointly trying to estimate a process. One sensor takes a measurement at every time step and the measurements are then exchanged among all the sensors. What is the sensor schedule that results in the minimum error covariance? We describe a stochastic sensor selection strategy that is easy to implement and is computationally tractable. The problem described above comes up in many domains out of which we discuss two. In the sensor selection problem, there are multiple sensors that cannot operate simultaneously (e.g., sonars in the same frequency band). Thus measurements need to be scheduled. In the sensor coverage problem, a geographical area needs to be covered by mobile sensors each with limited range. Thus from every position, the sensors obtain a different view-point of the area and the sensors need to optimize their trajectories. The algorithm is applied to these problems and illustrated through simple examples.

Keywords: Sensor selection; Sensor location selection; Distributed sensor networks; Riccati-like equation; Random algorithms

1. Introduction and motivation

Recently there has been a lot of interest in networks of sensing agents which act cooperatively to obtain the best estimate possible, e.g., see Roumeliotis and Bekey (2002), Hall and Llinas (1997), Viswanathan and Varshney (1997) and the references therein. While such a scheme admittedly has higher complexity than the strategy of treating each sensor independently, the increased accuracy often makes it worthwhile. If all the sensors exchange their measurements, the resulting estimate can be better even than the sensor with the least measurement noise (were no information exchange happening). Works such as the EYES project (Karl, 2002), WINS (Estrin, Govindan, Heidemann, & Kumar, 1999), and Smart Dust (Kahn, Katz, & Pister, 1999), are examples of systems implementing such networks. In addition, sensor network ideas are also being used for fulfilling specific tasks like reconnaissance, surveillance, data gathering and so on (Curtin, Bellingham, Catipovic, & Webb, 1993; Rybski et al., 2000; Weisbin et al., 1999).

Some special issues in sensor networks from an estimation perspective are fusion of data emerging from multiple nodes, association of measurements with targets in case multiple targets are present, scheduling in case all sensors cannot take or transmit simultaneous measurements, optimal positioning of sensors and so on. In this note, we present an algorithm to solve the problem of sensor scheduling. In addition, the algorithm can also be extended for use in the problem of optimal positioning and trajectory generation.

The problem of sensor scheduling arises when one (or multiple) sensors have to be selected out of N given sensors at every time step. This might be the case if, e.g., there are echo-based sensors like sonars which can interfere with each other. If the sensors observe a schedule and thus minimize simultaneous (and hence interfering) measurements, the total sensor power consumption can also be reduced. Another situation where sensor scheduling is useful is in tracking and discrimination problems, where a radar can make different

☆ A preliminary version of the paper was presented at the 16th IFAC World Congress. This paper was recommended for publication in revised form by Associate Editor Ioannis Paschalidis under the direction of Editor Ian Petersen.
* Corresponding author.
E-mail addresses: vijay@cds.caltech.edu (V. Gupta), timothyc@caltech.edu (T.H. Chung), hassibi@caltech.edu (B. Hassibi), murray@caltech.edu (R.M. Murray).

0005-1098/$ - see front matter © 2005 Elsevier Ltd. All rights reserved.
doi:10.1016/j.automatica.2005.09.016

types of measurements by transmitting a suitable waveform each of which has a different power requirement. There might be shared communication resources (e.g., broadcast channels or a shared communication bus) that constrain the usage of many sensors at the same time. Such a situation arises, e.g., in telemetry-data aerospace systems. Because of its importance, this problem has received considerable attention in literature. The seminal work of Meier, Peschon, and Dressler (1967) showed that for linear plants and quadratic cost functions, a separation property holds between the optimal plant control policy and the measurement control policy. The measurement control problem, which is the sensor scheduling problem, was cast as a non-linear deterministic control problem and shown to be solvable by a tree-search in general. That work proposed forward dynamic programming and a gradient method for solution. To deal with the complexity of a tree-search, greedy algorithms have been proposed many times, some examples being the works of Oshman (1994), Kagami and Ishikawa (2004), Gupta, Chung, Hassibi, and Murray (2004). Allied contributions have dealt with robust sensor scheduling (Savkin, Evans, & Skafidas, 2000), a greedy algorithm with an information based cost measure (Zhao, Shin, & Reich, 2002), a numerical method for obtaining sub-optimal schedules with error bounds (Alriksson & Rantzer, 2005) and include the works of Miller and Runggaldier (1997), Krishnamurthy (2002), Rago, Willet, and Bar-Shalom (1996) etc. A different numerical approach to solve the problem was provided in Athans (1972) where the problem was cast as a two-point boundary value problem. The non-linear matrix differential equations thus obtained were solved numerically by a min-H technique. These ideas were extended to discrete-time systems in Kerr (1981–1982) and the two-point boundary value problem was converted to an initial value problem in Kerr and Oshman (1995). In Herring and Melsa (1974), the technique was generalized to consider multiple devices being chosen at the same time.

Our algorithm differs from these approaches in that it is based on the idea of letting the sensors switch randomly according to some optimal probability distribution to obtain the best *expected* steady-state performance.[1] Besides being numerically more tractable than tree-search based and similar solutions proposed in the literature, it does not rely on the sensors having considerable computational power or knowledge about other sensors. There are numerous other advantages as will be pointed out later. Our algorithm can also be applied to the problem of sensor trajectory generation for optimal coverage of an area. This problem arises when there are some specified number of mobile sensors that can each sense over a limited region but together they must monitor a given area. The problem of optimal sensor location in case there are no bounds on the range over which the sensors can sense leads to the problem of Voronoi partitioning of the space and has been solved both in a centralized framework (Du, Faber, & Gunzburger, 1999; Okabe, Boots, Sugihara, & Chiu, 2000; Okabe & Suzuki, 1997) and in a decentralized fashion (Cortes, Martinez, Karatas, & Bullo, 2004). The problem when there are range (or direction) limitations on the sensors has also been looked at in the literature. However, most of the approaches that have been proposed are very application specific (Heng, Kuno, & Shirai, 1997; Mori, 1990; Nakamoto, Ishida, & Moriizumi, 1997). The general problem of determining the optimal trajectory can again be cast as a tree-search problem and greedy approaches have often been proposed (Basir & Shen, 1995; Chung, Gupta, Burdick, & Murray, 2004; Mukai & Ishikawa, 1996). We can again obtain many advantages over such algorithms by using our method.

The paper is organized as follows. In the next section, we state our assumptions and set up the problem. Then we present our random sensor selection algorithm and briefly analyze it. We illustrate the algorithm and some of its advantages using some simple simulations. We finish with conclusions and avenues for future research.

2. Modeling and problem formulation

Consider a system evolving according to the equation

$$x[k+1] = Ax[k] + Bw[k]. \tag{1}$$

$x[k] \in \mathbf{R}^n$ is the process state at time step k and $w[k]$ is the process noise assumed white, Gaussian and zero mean with covariance matrix Q. The initial state $x[0]$ is assumed to be a Gaussian zero mean random variable with covariance Π_0. The process state is being observed by N sensors with the measurement equation for the ith sensor being

$$y_i[k] = C_i x[k] + v_i[k], \tag{2}$$

where $y_i[k] \in \mathbf{R}^m$ is the measurement. The measurement noises $v_i[k]$'s for the sensors are assumed independent of each other and of the process noise. Further the noise $v_i[k]$ is assumed to be white, Gaussian and zero mean with covariance matrix R_i. At every time step, one sensor takes a measurement which is then communicated to all the sensors (or a data sink) in an error-free manner. The assumption of one sensor per time step is without loss of generality. Since all the measurements are being shared, all sensors have the same estimate of the process state $x[k]$, denoted by $\hat{x}[k]$. Further the optimal estimate is given by a Kalman filter assuming a time-varying sensor. Assuming that the ith sensor takes the measurement at time step k, the covariance of the estimate error $P[k]$ evolves according to the Riccati recursion

$$P[k+1] = AP[k]A^{\mathrm{T}} + BQB^{\mathrm{T}} - AP[k]C_i^{\mathrm{T}}(C_i P[k] C_i^{\mathrm{T}} + R_i)^{-1} C_i P[k] A^{\mathrm{T}}, \tag{3}$$

with the initial condition given by $P[0] = \Pi_0$.

It is obvious from (3) that error covariance is a function of the sensor schedule. We wish to find the sensor schedule that minimizes the steady-state error covariance. We can represent all the possible sensor schedule choices by a tree structure. The depth of any node in the tree represents time instants with the root at time zero. The branches correspond to choosing a particular sensor to be active at that time instant. Thus, the path from the root to any node at depth d represents a particular

[1] As is made clear later, for computational ease, we actually minimize an upper bound on the expected steady-state performance.

sensor schedule choice for time steps 0 to d. We can associate with each node the cost function evaluated using the sensor schedule corresponding to the path from the root to that node. Obviously, finding *the* optimal sequence requires traversing all the paths from the root to the leaves in the tree. If the leaves are at a depth δ, a total of N^δ schedules need to be compared. This procedure might place too high a demand on the computational and memory resources of the system. In the next section, we present an alternative algorithm that does not involve traversing the tree.

The problem of optimal sensor trajectory generation can also be cast in the above framework. For simplicity, assume that only one sensor is present. We discretize the area to be covered into a grid and assume that the discretization is fine enough so that we only need to observe the evolution of the process at these points. In other words, sampling the underlying process at these discrete points is sufficient to cover the entire area. We model the limited sensing range by assuming (for example) that if the sensor is at a particular point, it generates measurements corresponding to that point only. As a simple example consider that the area to be monitored has been discretized into N points, denoted by $l_1, l_2, \ldots, l_N$. At location l_i, the process evolves according to the equation

$$x_i[k+1] = A_i x_i[k] + \sum_{j \neq i} A_{ij} x_j[k] + B_i w_i[k],$$

where we have assumed that the process state at location l_i is also affected by the process states at other locations. Thus by stacking the process states at all these locations into a single vector $x[k]$, we see that for the entire process, the evolution is of the form given in (1). Similarly assume that if the sensor is at location l_i, its measurement is described by the equation

$$y_i[k] = H_i x_i[k] + v_i[k].$$

By defining C_i suitably, this can easily be recast in the form of (2). Clearly there are N such virtual sensors. Thus we see that the sensor trajectory problem is equivalent to the sensor scheduling problem described earlier where N sensors are present but only one can be selected to take the measurement. We can model physical constraints on the sensor motion by assuming that the sensor can move from its current location only to its immediate neighbors. In the tree approach described earlier, it imposes some constraints on the branches of the tree that can be present. In our algorithm, this constraint can be modeled by assuming that the sensors are selected with transition probabilities described by a Markov chain. The states of the Markov chain represent the location of the sensor. Thus the probability of moving from one location to a location far away in a single time step is zero. Some of the results in this paper were also applied for the special case when A is block-diagonal (the system dynamics at different points are uncoupled) in Tiwari, Jun, Jeffcoat, and Murray (2005).

3. Description of the algorithm

Our algorithm consists of choosing sensors randomly according to some probability distribution. The probability distribution is chosen so as to minimize the expected steady-state error covariance. Note that we cannot calculate the exact value of the error covariance since that will depend on the specific sensor schedule chosen. Hence we optimize the expected value of the error covariance. For obtaining the expected value given any particular probability distribution, we proceed as follows. Consider a sensor that senses a process of form (1) according to the relation

$$y[k] = Cx[k] + v[k],$$

where the noise $v[k]$ has covariance matrix R. Then we define

$$f_C(P) = APA^{\mathrm{T}} + BQB^{\mathrm{T}} - APC^{\mathrm{T}}(CPC^{\mathrm{T}} + R)^{-1}CPA^{\mathrm{T}},$$

$$f_C^k(P) = \underbrace{f_C(f_C(\cdots(f_C(P))))}_{f_C \text{ applied } k \text{ times}}.$$

Thus the error covariance of the estimate at time step $k+1$ if Kalman filter is being used is given by $f_C(P)$, where P is the error covariance at time step k. We would need the following properties of this operator.

Lemma 1. *$f_C(P)$ is concave in P provided P is positive semi-definite and R is positive definite.*

Proof. Proof follows readily from the fact (Boyd & Vandenberghe, 2003) that a function $f(x)$ is concave in x if and only if $f(x_0 + th)$ is concave in the scalar t for all x_0 and h. □

Lemma 2. *Let X and Y be two positive semi-definite matrices. If $X < Y$, then $f_C(X) < f_C(Y)$. Moreover, for any C and for any positive semi-definite matrix X, $f_C(X) \geqslant BQB^{\mathrm{T}}$.*

Proof. Following Sinopoli et al. (2004), introduce

$$\phi(K, X) = BQB^{\mathrm{T}} + (A + KC)X(A + KC)^{\mathrm{T}} + KRK^{\mathrm{T}},$$

and note that

$$K = -AXC^{\mathrm{T}}(CXC^{\mathrm{T}} + R)^{-1} \triangleq K_X$$

minimizes $\phi(K, X)$. Moreover $\phi(K_X, X) = f_C(X)$. Also note that $\phi(K, X)$ is an increasing function in the second argument. Thus, we see that

$$\phi(K_X, X) \leqslant \phi(K_Y, X) \leqslant \phi(K_Y, Y).$$

This proves the first part of the lemma. For the second part, note that

$$f_C(X) = \phi(K_X, X) \geqslant \phi(K_X, 0) = BQB^{\mathrm{T}}. \qquad \square$$

Consider the time-varying Kalman filter recursion given in (3) for the system given by (1). Suppose that the sensor at every time step k is chosen from among the choices $S_1, S_2, \ldots, S_N$ with the probability of S_i being chosen at time step k being π_i^k. The associated sensor measurement matrix is C_i and the noise covariance matrix is R_i. To begin with, assume that the choice is done independently at each time step. Denoting the

choice at time step k by $C[k]$ and $R[k]$, the evolution of the error covariance for any sensor can be written as

$$P[k+1] = f_{C[k]}(P[k]),$$

with $P[0]$ as the initial condition. Note that the error covariance $P[k+1]$ is random since it depends on the particular sequence of chosen S_i's ($0 \leqslant i \leqslant k$). We look at its expected value and try to evaluate the expectation in the limit as $k \to \infty$. Thus we are interested in

$$E[P[k+1]] = E[f_{C[k]}(P[k])]. \tag{4}$$

Explicitly evaluating this expectation appears to be intractable. We look instead for an upper bound. We have the following result.

Theorem 3 (*Upper bound and its convergence*). *Let there be N sensors out of which one sensor is randomly chosen per time step for taking measurements. If the ith sensor is chosen at time step k with probability π_i^k independently at each time step, then the expected error covariance of the estimate is upper bounded by $\Delta[k+1]$ where $\Delta[k]$ is given by the recursion*

$$\begin{aligned}\Delta[k+1] &= BQB^{\mathrm{T}} + A\Delta[k]A^{\mathrm{T}} \\ &\quad - \sum_{i=1}^{N} \pi_i^k [A\Delta[k]C_i^{\mathrm{T}}(R_i + C_i\Delta[k]C_i^{\mathrm{T}})^{-1} \\ &\quad \times C_i\Delta[k]A^{\mathrm{T}}],\end{aligned} \tag{5}$$

with the initial condition $\Delta[0] = P[0]$. Further suppose that the sensor probabilities π_i^k tend to constants q_i as $k \to \infty$. If there exist matrices $K_1, K_2, \ldots, K_N$ and a positive definite matrix P such that

$$P > BQB^{\mathrm{T}} + \sum_{i=1}^{N} q_i((A + K_iC_i)X(A + K_iC_i)^{\mathrm{T}} + K_iR_iK_i^{\mathrm{T}}),$$

then the iteration in (5) *converges for all initial conditions $P[0] \geqslant 0$ and the limit $\bar{P}$ is the unique positive semi-definite solution of the equation*

$$\begin{aligned}X &= BQB^{\mathrm{T}} + AXA^{\mathrm{T}} \\ &\quad - \sum_{i=1}^{N} q_i A[XC_i^{\mathrm{T}}(R_i + C_iXC_i^{\mathrm{T}})^{-1}C_iX]A^{\mathrm{T}}.\end{aligned} \tag{6}$$

Proof. First note that the quantities $P[k]$ and $C[k]$ are independent. Thus we can explicitly take the expectation in (4) with respect to the probability distribution of $C[k]$ and write

$$E[P[k+1]] = \sum_{i=1}^{N} \pi_i^k E[f_{C_i}(P[k])],$$

where the expectation on the right-hand side is now over $C[0], \ldots, C[k-1]$. Now we use Jensen's inequality (see, e.g., Gradshteyn & Ryzhik, 2000) on account of Lemma 1. Thus we immediately obtain

$$\begin{aligned}E[P[k+1]] &= \sum_{i=1}^{N} \pi_i^k E[f_{C_i}(P[k])] \\ &\leqslant \sum_{i=1}^{N} \pi_i^k f_{C_i}(E[P[k]]).\end{aligned} \tag{7}$$

Since $f_{C_i}(.)$ is an increasing operator, we obtain the required upper bound. For the convergence, proof is similar to the one of Theorem 1 in Sinopoli et al. (2004). We redefine the quantities

$$\mathscr{L}(Y) = \sum_{i=1}^{N} q_i(A + K_iC_i)Y(A + K_iC_i)^{\mathrm{T}},$$

$$\begin{aligned}&\phi(K_i, P) \\ &\quad = BQB^{\mathrm{T}} + \sum_{i=1}^{N} q_i((A + K_iC_i)X(A + K_iC_i)^{\mathrm{T}} + K_iR_iK_i^{\mathrm{T}})\end{aligned}$$

and follow the arguments given in that proof. □

The convergence of the upper bound implies boundedness of the recursion in (4). As an example, if all eigenvalues of A are strictly less than unity in magnitude, we can always find matrices K_i's and P satisfying the above conditions by choosing K_i's as the zero matrices and P as $2\bar{P}$ where $\bar{P}$ is the positive definite solution of the Lyapunov equation

$$\bar{P} = A\bar{P}A^{\mathrm{T}} + BQB^{\mathrm{T}}.$$

Thus as long as A is stable, the recursion in (7) converges. The case when A is stable (and thus the process to be estimated does not grow unbounded) is very important in a large number of practical applications of estimation.

The algorithm thus consists of choosing q_i's so as to optimize the upper bound as a means of optimizing the expected steady-state value of P_k itself. The problem is solved under the constraint of probabilities being non-negative and summing up to 1. The optimization problem can be solved by a gradient search algorithm or even by brute force search for a reasonable value of N. After determining the probability values, the sensors are turned on and off with their corresponding probabilities. Note that the implementation does assume some shared randomness and synchronization among the sensors so that two sensors are not turned on at the same time. This can readily be achieved, e.g., through a common seed for a pseudo-random number generator available to all the sensors. Alternatively a token-based mechanism for the scheme can be implemented. Also note that the algorithm is run off-line and it has to be reapplied everytime the number of sensors changes. However, if a sensor is stochastically failing with a known probability, we can model that in the algorithm.

For the cases when A is not stable, we need to find out if (4) diverges. We now obtain a lower bound for the recursion. If the lower bound does not converge, it will imply the non-convergence of the expected steady-state error covariance. We note the following result.

Theorem 4 (*Lower bound and its convergence*). *Suppose there are N sensors out of which one sensor is randomly chosen per time step for taking measurements and the ith sensor is chosen with probability π_i^k at time step k independently at each time step. Define*

$$\bar{\pi}_j^{k,t} = \pi_j^k \pi_j^{k-1} \cdots \pi_j^{k-t+1}.$$

Then the expected error covariance of the estimate at time step k is lower bounded by X[k] where X[k] is obtained from the equation

$$\begin{aligned} X[k] = {} & \bar{\pi}_j^{k-1,k} f_{C_j}^k(P[0]) + (1-\pi_j^{k-1})BQB^{\mathrm{T}} \\ & + \bar{\pi}_j^{k-1,1}(1-\pi_j^{k-2}) f_{C_j}(BQB^{\mathrm{T}}) \\ & + \bar{\pi}_j^{k-1,2}(1-\pi_j^{k-3}) f_{C_j}^2(BQB^{\mathrm{T}}) \\ & + \cdots + \bar{\pi}_j^{k-1,k-1}(1-\pi_j^0) f_{C_j}^{k-1}(BQB^{\mathrm{T}}), \end{aligned} \tag{8}$$

where P[0] is the initial error covariance used in calculating the expected error covariance through (4). *Note that one such lower bound results for each value of $j = 1, \ldots, N$. Further, suppose that the sensor probabilities π_i^k tend to constants q_i as $k \to \infty$. Then the condition for X[k] given in* (8) *to stay bounded as $k \to \infty$ is*

$$q_j |\lambda_{\max}(\bar{A}_j)|^2 \leqslant 1, \tag{9}$$

where q_j is the probability of choosing the jth sensor while $\lambda_{\max}(\bar{A}_j)$ refers to the eigenvalue with the maximum magnitude of the unobservable part of A when the pair (A, C_j) is put in the observable canonical form (Dullerud & Paganini, 2000).

Proof. The event space for the sensor schedule till time step $k-1$ (which determines the value of $P[k]$) can be partitioned into $k+1$ disjoint events E_i of the form: sensor S_j was chosen consecutively for the last i time steps and in the time step just before that, S_j was not chosen, $0 \leqslant i \leqslant k$. Thus the expected error covariance is given by

$$E[P[k]] = \sum_{i=0}^{k} p(E_i) V(E_i),$$

where $p(E_i)$ refers to the probability of E_i occurring and $V(E_i)$ refers to the value of error covariance under the event E_i. Now consider the ith term in the summation, where $i < k$. Note that

(1) When the sensor S_j is chosen at time step m, the error covariance at the time step $m+1$ is given by $f_{C_j}(\Sigma)$ where Σ was the error covariance at the present time step.
(2) When any other sensor is chosen the corresponding error covariance at time step $m+1$ is lower bounded by BQB^{T}. Moreover if, then, at time step $m+1$, the jth sensor is chosen, the error covariance at time step $m+2$ is lower bounded by $f_{C_j}(BQB^{\mathrm{T}})$.

By combining these two facts, we see that

$$V(E_i) \geqslant f_{C_j}^i(BQB^{\mathrm{T}}).$$

Thus we obtain

$$p(E_i)V(E_i) \geqslant \bar{\pi}_j^{k-1,i}(1-\pi_j^{k-1-i}) f_{C_j}^i(BQB^{\mathrm{T}}).$$

For the term E_k, from the definition we obtain

$$p(E_k)V(E_k) = \bar{\pi}_j^{k-1,k} f_{C_j}^k(P[0]).$$

By adding together the terms $p(E_i)V(E_i)$, we obtain that $X[k]$ as given in (8) is indeed a lower bound for the expected error covariance.

For the purpose of studying the boundedness of estimation error, we will denote the sensor used for the calculation in (8), C_j, by C. Note that we can assume without loss of generality that the pair (A, C) is in the observer canonical form. (If not, an invertible linear transformation can convert it to the companion form. This transformation will not affect the boundedness of estimation error.) Denote the matrices in the form

$$A = \begin{bmatrix} \bar{A}_{11} & 0 \\ \bar{A}_{21} & \bar{A}_{22} \end{bmatrix}, \quad C = [\bar{C}_1 \quad 0].$$

Denote by $e[k]$ the error in estimating $x[k]$. Split $e[k]$ in accordance with the observer canonical form so that with a suitable choice of the Kalman filter gain matrix $K[k]$, the error evolves as

$$e[k+1] = (A - K[k]C)e[k] + Bw[k] + K[k]v[k],$$

where

$$e[k] = \begin{bmatrix} e_1[k] \\ e_2[k] \end{bmatrix}, \quad K[k] = \begin{bmatrix} K_1[k] \\ K_2[k] \end{bmatrix},$$

$e_1[k]$ corresponds to the observable part of the system and $e_2[k]$ to the unobservable part. Denote

$$\begin{aligned} P[k] &= \begin{bmatrix} E[e_1[k]e_1[k]^{\mathrm{T}}] & E[e_1[k]e_2[k]^{\mathrm{T}}] \\ E[e_2[k]e_1[k]^{\mathrm{T}}] & E[e_2[k]e_2[k]^{\mathrm{T}}] \end{bmatrix} \\ &= \begin{bmatrix} P_{11}[k] & P_{12}[k] \\ P_{21}[k] & P_{22}[k] \end{bmatrix}. \end{aligned}$$

Since $e_1[k]$ corresponds to the observable part of the system, $P_{11}[k]$ remains bounded. $P_{22}[k]$ evolves as

$$P_{22}[k+1] = \bar{A}_{22} P_{22}[k] \bar{A}_{22}^{\mathrm{T}} + \Sigma[k],$$

where $\Sigma[k]$ depends on noise terms, $P_{11}[k]$ and $P_{12}[k]$ but not on $P_{22}[k]$. Note that $\Sigma[k]$ remains bounded as $k \to \infty$ if $P_{11}[k]$ and $P_{22}[k]$ remain bounded. Now for time-invariant probabilities of sensor usage, the ith term of the lower bound in (8) is of the form

$$T_i = q_j^{i-1}(1-q_j) f_{C_j}^{i-1}(BQB^{\mathrm{T}}).$$

Because of the observability assumption, the Riccati recursion for the observation error covariance matrix when only sensor j is used will converge to a constant value irrespective of the initial condition. Thus for large enough i, $f_{C_j}^{i-1}(BQB^{\mathrm{T}}) = P[i-1]$ and the ith term can be rewritten as

$$T_i = q_j^{i-1}(1-q_j) P[i-1].$$

Thus $X[k]$ is obtained through a summation of the form

$$X[k] = \sum_{i=0}^{k-1} q_j^{i-1}(1-q_j)P[i-1] = \begin{bmatrix} \sum_{i=0}^{k-1}\gamma[i]P_{11}[i-1] & \sum_{i=0}^{k-1}\gamma[i]P_{12}[i-1] \\ \sum_{i=0}^{k-1}\gamma[i]P_{21}[i-1] & \sum_{i=0}^{k-1}\gamma[i]P_{22}[i-1] \end{bmatrix},$$

where $\gamma[i] = q_j^{i-1}(1-q_j)$. There are four terms here whose boundedness needs to be considered. Now, as already stated, the $(1,1)$ term is bounded because of the observability assumption. The $(2,2)$ term is bounded if and only if (9) holds. Also if both $(1,1)$ and $(2,2)$ terms are bounded, the off-diagonal terms of $P[k]$ are bounded by the Cauchy–Schwarz inequality. □

Note that the bounds we have obtained can be specialized to cases when one sensor pertains to a data loss situation similar to that considered in Sinopoli et al. (2004) and Liu and Goldsmith. If the sensors are being chosen according to a Markov chain, we can still obtain upper and lower bounds in a similar form as the ones discussed above. We give the bounds below. The convergence conditions can be derived in a similar way as before. Let $[q_{ij}]$ be the transition probability matrix of the Markov chain. Also let π_k^j be the probability of being in Markov state j at time step k.

Theorem 5. *If the sensors are chosen according to a Markov chain, $E[P_k]$ is upper bounded by X_k and lower bounded by Y_k where*

$$X_{k+1} = \sum_{j=1}^{N} \pi_k^j X_{k+1}^j, \quad \pi_k^j X_{k+1}^j = \sum_{i=1}^{N} f_{C_j}(X_k^i) q_{ij} \pi_k^i,$$

$$Y_k = \sum_{i=1}^{k} q_{jj}^{i-1}(\pi_{k+1-i}^j - q_{jj}\pi_{k-i}^j) f_{C_j}^i(BQB^{\mathrm{T}}) + q_{jj}^{k-1}\pi_0^j f_{C_0}^k(P_0).$$

Proof. The proof of the lower bound is similar to the one for the i.i.d. case. For the upper bound, let $C_k = j$ denote that jth sensor was chosen at time step k. Then $E[P_k] = \sum \pi_k^j E[P_k | C_{k-1} = j]$. Also

$$\begin{aligned} &\pi_k^j E[P_{k+1}|C_k = j] \\ &= \pi_k^j \sum_{i=1}^{N} E[P_{k+1}|C_k = j, C_{k-1} = i] \Pr(C_{k-1} = i | C_k = j) \\ &= \sum_{i=1}^{N} E[f_{C_j}(P_k)|C_{k-1} = i] q_{ij} \pi_{k-1}^i \\ &\leqslant \sum_{i=1}^{N} f_{C_j}(E[P_k|C_{k-1} = i]) q_{ij} \pi_{k-1}^i. \end{aligned}$$

Since $f_{C_j}(.)$ is an increasing operator, we obtain the required bound. In the above derivation, we have used the fact that $f_{C_j}(P_k)$ and C_k are conditionally independent given C_{k-1}. □

4. Simulation results

In this section, we apply our algorithm to a few sample problems and show that the algorithm offers a new interesting and powerful tool in several problems. Our first example is in the domain of sensor scheduling. Assume a vehicle moving in 2-D space according to the standard constant acceleration model.This model assumes that the vehicle has constant acceleration equal to zero except for a small perturbation. We assume that the vehicle moves in two dimensions. Denoting the position of the vehicle in the two dimensions by p_x and p_y, the velocities by v_x and v_y and with a discretization step size of h, the dynamics of the vehicle are of form (1) where

$$A = \begin{bmatrix} 1 & 0 & h & 0 \\ 0 & 1 & 0 & h \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}, \quad B = \begin{bmatrix} h^2/2 & 0 \\ 0 & h^2/2 \\ h & 0 \\ 0 & h \end{bmatrix}, \quad X = \begin{bmatrix} p_x \\ p_y \\ v_x \\ v_y \end{bmatrix}.$$

The term $w[k]$ is the perturbation term in acceleration and is modeled as a zero mean white Gaussian noise. In the numerical example, $h = 0.2$. The process noise is considered to have covariance matrix Q given by

$$Q = \begin{bmatrix} 1 & 0.25 \\ 0.25 & 1 \end{bmatrix}.$$

We assume two sensors with the measurements taken by the two sensors, y_1 and y_2 being described by

$$y_i[k] = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{bmatrix} X[k] + v_i[k]. \tag{10}$$

The terms $v_i[k]$ model the measurement noise, again assumed white, zero mean and Gaussian and also independent from each other and from $w[k]$. We consider values of the sensor noise covariances as

$$R_1 = \begin{bmatrix} 2.4 & 0 \\ 0 & 0.4 \end{bmatrix}, \quad R_2 = \begin{bmatrix} 0.7 & 0 \\ 0 & 1.4 \end{bmatrix}. \tag{11}$$

The plot given in Fig. 1 illustrates that choosing any one sensor at all time steps is not optimal. The figure plots the cost measured as the sum of the traces of the error covariance matrices of the estimates of the two sensors when they adopt the strategy of choosing only sensor 1 or only sensor 2 or when they choose an arbitrarily generated schedule over 50 time steps. For comparison, the cost achievable by the optimal sensor strategy found by a sliding window approach to the tree-search (Gupta et al., 2004) is also given. We see that the even an arbitrary sensor switching strategy can help to bring down the cost.

Next we apply our random choice algorithm to find the optimal probability distribution. On optimizing the upper bound in (6) over q_1 and q_2, the optimal probability for sensor 1 turns out to be $q_1 = 0.395$. Indeed, if we find the optimal sequence by a complete tree-search, it turns out that in the steady state, the percentage of sensor 1 in the sequence is about 37%. For this probability distribution, the steady-state value of the upper bound of the sum of the traces of the expected error covariance matrices for the two sensors turns out to be 2.3884, which compares well with the value of about 2.3 obtained by the optimal

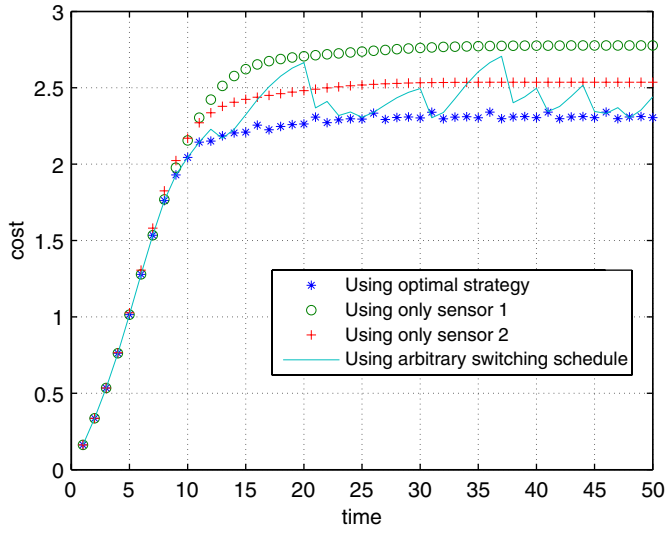

Fig. 1. Sensor switching helps to bring the cost down.

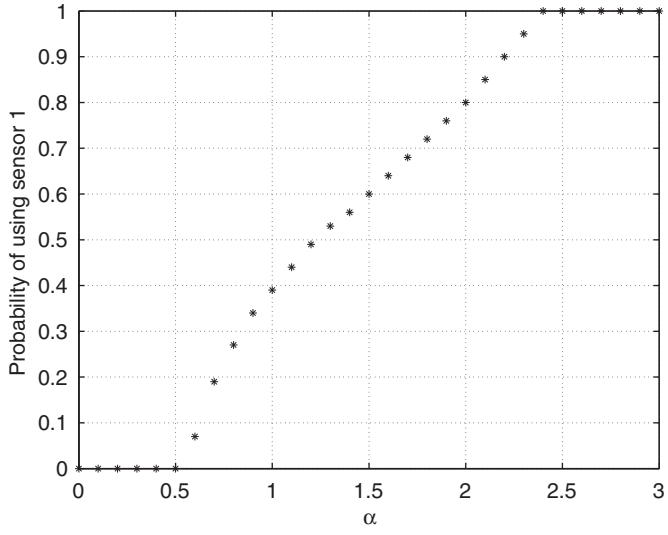

Fig. 2. Optimal probability of use of sensor 1 as sensor 2 gets noisier.

strategy. Note that our algorithm results in orders of magnitude less calculation than tree-search algorithms and finds a near-optimal schedule in the steady state. The computational savings can be very significant if we need to study the optimal sensor schedule as some sensor parameter is varied. As an example, let the measurement noise covariance of the second sensor be given by

$$R_2 = \alpha \begin{bmatrix} 0.7 & 0 \\ 0 & 1.4 \end{bmatrix}.$$

Fig. 2 plots the optimal percentage of use of sensor 1, as the parameter α is varied. The plot shows that there is a threshold phenomenon such that increasing the noise of one sensor beyond that level results in that particular sensor never being used. The reduction in computation needed makes our algorithm much more scalable in the number of sensors than the tree-based algorithms.

In addition, there are several unique advantages that our algorithm offers over the conventional algorithms. A very important one is the issue of sensor costs. Frequently, there are other considerations beyond estimation accuracy in using one sensor over another. As an example, it might be more costly to use a very accurate sensor at every time step. Similarly, we might want some sort of fairness such that one sensor is not used all the time such that all its power is drained. Usually, it is not clear how to appropriately weight the sensor costs with estimation costs. Thus it is not clear how to even generate a tree for the sensor schedule choices. However, it is easy to take sensor costs into account with our algorithm. As an example we consider three sensors of the form of (10) being present with the measurement noise covariances being given by

$$R_1 = \begin{bmatrix} 3.24 & 0 \\ 0 & 1.04 \end{bmatrix}, \quad R_2 = \begin{bmatrix} 0.25 & 0 \\ 0 & 1.36 \end{bmatrix},$$
$$R_3 = \begin{bmatrix} 0.56 & 0 \\ 0 & 0.56 \end{bmatrix}.$$

Suppose the three sensors are transmitting to a single data sink so that the only energy consumption is in taking a measurement and then broadcasting it. If we try to optimize the probability distribution, we obtain that sensor 2 should be chosen with a probability of 0.2 and sensor 3 with a probability of 0.8. However, such a strategy would lead to sensor 3 draining away all its power very quickly and thus we might want to impose an additional constraint that on average, no sensor is used more than twice as much as any other sensor. We restrict our search to the relevant $q_1 - q_2$ space and come up with the optimal probabilities satisfying the additional constraint as sensor 1 being used with a probability of 0.2 and sensors 2 and 3 being used each with a probability of 0.4.

Another situation in which our algorithm is much more easily used is when there is some randomness imposed on the system. As an example, consider the case of two sensors with measurement noise covariances given by the values in (11). Suppose that the sensors are communicating with a data sink over a communication channel that randomly drops packets with probability λ. Compared to the conventional methods, it is easy to take the channel into account while using our algorithm. We set up Eq. (5) assuming that there are three sensors present. The first two sensors have covariance matrices given above and they are chosen with probabilities $q_1(1-\lambda)$ and $q_2(1-\lambda)$. The third sensor corresponds to the packet being dropped (and hence no measurement being taken) and it is chosen with a probability of $(q_1+q_2)\lambda$. Then we optimize this bound over the parameters q_1 and q_2. Fig. 3 shows the change in the optimal probability of choosing sensor 1 as the packet drop probability λ is varied. The plot shows that the packet drop probability plays a big role in determining the optimal sensor schedule.

The lower bound derived in Theorem 4 is useful for obtaining the region in the sensor usage probability space where the expected error covariance in (4) diverges. We illustrate the lower bound with an example now. First consider the same system as in (1) being measured by two sensors of the form of (10). The measurement noise covariances are given by the values

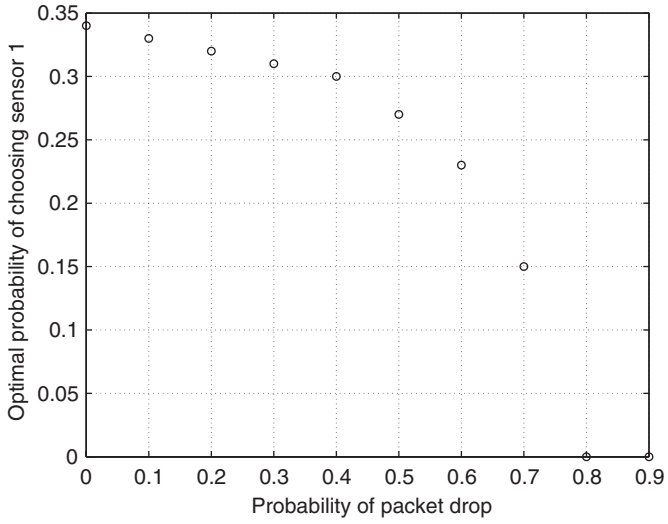

Fig. 3. Optimal probability of use of sensor 1 varies if the channel is dropping packets.

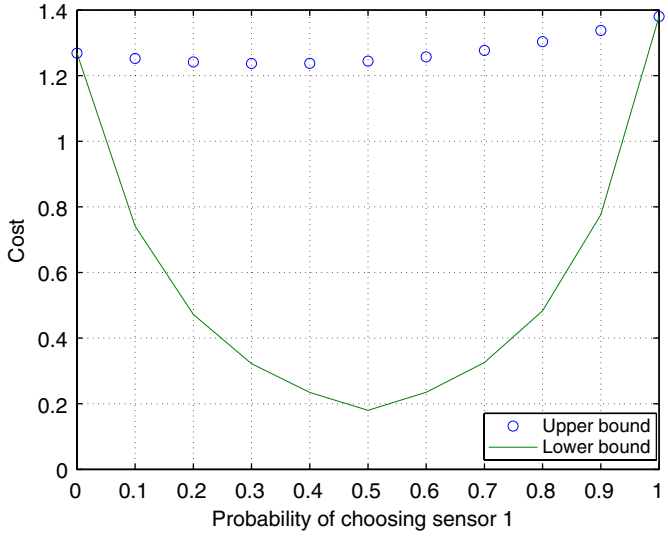

Fig. 4. Bounds may not be very tight.

in (11). The upper bound and the lower bounds are plotted in Fig. 4. We can see that the lower bounds may not be very tight. However, the main utility of the lower bounds is in predicting when the expected error covariance necessarily diverges. We consider the same example with the second sensor replaced by a sensor of the form

$$y[k] = \begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} X[k] + v[k],$$

with the sensor noise covariances given by (11). We see that the plant is unobservable while using the second sensor alone and hence as the probability of using the second sensor increases, the error covariance would diverge. It can be shown that although there is a huge gap between the lower and upper bounds, both bounds diverge at $q_1 = 0.56$ which is thus the critical probability for error divergence. This value also matches the value given in Theorem 4 since the largest eigenvalue of the unobservable part of A is 1.5. It may be noted that in general, the probabilities when the bounds diverge will not match and they serve as lower and upper bounds on the critical probability. An important special case is when all but one of the matrices C_i are invertible which renders the condition of divergence of the lower bound both necessary and sufficient for the divergence of the expected error covariance. This is the case, e.g., in scalar systems.

To consider a representative example for sensor coverage, we consider an area gridded into $N = 4$ points being surveyed by one sensor. Each point is associated with a scalar process that tries to find the average of the neighboring points. Thus denoting the process at the ith point by x_i and the state of the entire system by $X = [x_1\ x_2\ x_3\ x_4]^{\mathrm{T}}$, we see that the system evolves as

$$X[k+1] = \begin{bmatrix} 0.5 & 0.5 & 0 & 0 \\ 0.33 & 0.33 & 0.33 & 0 \\ 0 & 0.33 & 0.33 & 0.33 \\ 0 & 0 & 0.5 & 0.5 \end{bmatrix} X[k] + w[k],$$

where $w[k]$ is white noise with mean zero and covariance equal to the identity matrix. When the sensor is at point i, it can measure the value of x_i corrupted by a Gaussian zero mean noise. Thus, as explained earlier, we have four virtual sensors taking measurements according to

$$y_i[k] = x_i[k] + v_i[k],$$

where $v_i[k]$ are all independent of each other and their variances are given by $R_1 = 1$, $R_2 = 3.1$, $R_3 = 0.5$ and $R_4 = 1.5$. We model the sensors as switching according to the transition probability matrix

$$\begin{bmatrix} 1-\lambda_1 & \lambda_1 & 0 & 0 \\ \lambda_2 & 1-2\lambda_2 & \lambda_2 & 0 \\ 0 & \lambda_2 & 1-2\lambda_2 & \lambda_2 \\ 0 & 0 & \lambda_1 & 1-\lambda_1 \end{bmatrix}.$$

We want the Markov chain to reach a stationary distribution irrespective of the initial sensor probabilities. Thus we constrain the constants λ_1 and λ_2 to be non-zero. Optimizing the upper bound yields the values $\lambda_1 = 1$ and $\lambda_2 = 0.45$ with the optimal upper bound being 7.63. 100 000 simulations with these values of λ_i's yielded an average steady-state covariance of 7.62. Using a greedy algorithm leads to only sensor 3 being used at all time steps with the steady-state covariance of 8.72. As with the sensor scheduling example, we can illustrate various different cases when our algorithm offers advantages over the conventional methods. These can correspond to, e.g., constraining the probability that the sensor be located at some points that are more dangerous than others.

5. Conclusions and future work

In this paper, we described an algorithm for stochastically selecting sensors to minimize the expected error covariance. We presented upper and lower bounds on the error covariance. This algorithm offers many advantages over conventional algorithms for sensor selection. We applied the algorithm to the

problems of sensor scheduling and sensor coverage. Some simple numerical examples were also presented.

The work can potentially be extended in many ways. Finding out how tight the bounds are and coming up with tighter bounds is one avenue. Moreover we have talked only about expected error covariance with no indication of the spread of the actual value of the covariance. Coming up with an understanding of that would also be interesting.

Acknowledgments

The authors would like to thank Prof. Joel Burdick for helpful discussions. Research supported in part by NSF Grant CCR-0326554 for the first author and by the Engineering Research Centers Program of the National Science Foundation under Award Number EEC-9402726 and also a Grant from NASA for the second author.

References

Alriksson, P., & Rantzer, A. (2005). Sub-optimal sensor scheduling with error bounds. In *Proceedings of the 16th IFAC world congress.*

Athans, M. (1972). On the determination of optimal costly measurement strategies. *Automatica*, *18*, 397–412.

Basir, O. A., & Shen, H. C. (1995). Informational maneuvering in dynamic environment. In *IEEE international conference on systems, man and cybernetics* (Vol. 2, pp. 999–1004).

Boyd, S., & Vandenberghe, L. (2003). *Convex optimization*. Cambridge: Cambridge University Press.

Chung, T. H., Gupta, V., Burdick, J. W., & Murray, R. M. (2004). On a decentralized active sensing strategy using mobile sensor platforms in a network. In *Proceedings of the IEEE conference on decision and control* (Vol. 2, pp. 1914–1919).

Cortes, J., Martinez, S., Karatas, T., & Bullo, F. (2004). Coverage control for mobile sensing networks. *IEEE Transactions on Robotics and Automation*, *20*(2), 243–255.

Curtin, T. B., Bellingham, J. G., Catipovic, J., & Webb, D. (1993). Autonomous oceanographic sampling networks. *Oceanography*, *6*(3), 86–94.

Du, Q., Faber, V., & Gunzburger, M. (1999). Centroidal Voronoi tessellations: Applications and algorithms. *SIAM Review*, *41*(4), 637–676.

Dullerud, G. E., & Paganini, F. (2000). *A course in robust control theory: A convex approach*. New York: Springer.

Estrin, D., Govindan, R., Heidemann, J., & Kumar, S. (1999). Next century challenges: Scalable coordination in sensor networks. In *Proceedings of the fifth annual international conference on mobile computing and networks.* MobiCOM.

Gradshteyn, I. S., & Ryzhik, I. M. (2000). *Tables of integrals, series, and products*. New York: Academic Press.

Gupta, V., Chung, T., Hassibi, B., & Murray, R. M. (2004). Sensor scheduling algorithms requiring limited computation. In *Proceedings of the international conference on acoustics, speech and signal processing*.

Hall, D. L., & Llinas, J. (1997). An introduction to multisensor data fusion. *Proceedings of the IEEE*, *85*(1), 6–23.

Heng, T. C. H., Kuno, Y., & Shirai, Y. (1997). Active sensor fusion for collision avoidance. In *Proceedings of the IEEE/RSJ international conference on intelligent robots and systems* (Vol. 3, pp. 1244–1249).

Herring, K., & Melsa, J. (1974). Optimum measurements for estimation. *IEEE Transactions on Automatic Control*, *19*(3), 264–266.

Kagami, S., & Ishikawa, M. (2004). A sensor selection method considering communication delays. In *Proceedings of the 2004 IEEE international conference on robotics and automation* (Vol. 1, pp. 206–211).

Kahn, J., Katz, R., & Pister, K. (1999). Next century challenges: Mobile networking for "Smart Dust". In *Proceedings of ACM MobiCom conference.* MobiCom. Seattle, WA.

Karl, H. (2002). Making sensor networks useful: Distributed services—the EYES project. ESF Workshop, La Spezia, Italy.

Kerr, T. H. (1981–1982). Modeling and evaluating an empirical INS difference monitoring procedure used to sequence SSBN navaid fixes. *Journal of the Institute of Navigation*, *28*(4), 263–285.

Kerr, T. H., & Oshman, Y. (1995). Further comments on 'optimal sensor selection strategy for discrete-time estimators' [and reply]. *IEEE Transactions on Aerospace and Electronic Systems*, *31*, 1159–1167.

Krishnamurthy, V. (2002). Algorithms for optimal scheduling and management of hidden Markov model sensors. *IEEE Transactions on Signal Processing*, *60*(6), 1382–1397.

Liu, X., & Goldsmith, A. J. Kalman filtering with partial observation losses. In Proceedings of the *IEEE Transactions on decision and Control* (Vol. 4, pp. 4180–4186).

Meier, L., III, Peschon, J., & Dressler, R. (1967). Optimal control of measurement subsystems. *IEEE Transactions on Automatic Control*, *12*, 528–536.

Miller, B. M., & Runggaldier, W. J. (1997). Optimization of observations: A stochastic control approach. *SIAM Journal of Control and Optimization*, *35*(5), 1030–1052.

Mori, H. (1990). Active sensing in vision-based stereotyped motion. In *Proceedings of the IEEE international workshop on intelligent robots and systems '90* (Vol. 1, pp. 167–174).

Mukai, T., & Ishikawa, I. (1996). An active sensing method using estimated errors for multisensor fusion systems. *IEEE Transactions on Industrial Electronics*, *43*, 380–386.

Nakamoto, T., Ishida, H., & Moriizumi, T. (1997). Active odor sensing system. In *Proceedings of the IEEE international symposium industrial electronics*, *1997* (Vol. 1, pp. SS128–SS133).

Okabe, A., Boots, B., Sugihara, K., & Chiu, S. N. (2000). *Spatial tessellations: Concept and applications of Voronoi diagrams.* Wiley series in probability and statistics (2nd ed.). New York: Wiley.

Okabe, A., & Suzuki, A. (1997). Locational optimization problems solved through Voronoi diagrams. *European Journal of Operational Research*, *98*(3), 445–456.

Oshman, Y. (1994). Optimal sensor selection strategy for discrete-time state estimators. *IEEE Transactions on Aerospace and Electronic Systems*, *30*, 307–314.

Rago, C., Willet, P., & Bar-Shalom, Y. (1996). Censoring sensor: A low communication rate scheme for distributed detection. *IEEE Transactions on Aerospace and Electronic Systems*, *28*(2), 554–568.

Roumeliotis, S. I., & Bekey, G. A. (2002). Distributed multi-robot localization. *IEEE Transactions on Robotics and Automation*, *18*(5), 781–795.

Rybski, P. E., Papanikolopoulos, N. P., Stoeter, S. A., Krantz, D. G., Yesin, K. B., Gini, M., Voyles, R., Hougen, D. F., Nelson, B., & Erickson, M. D. (2000). Enlisting rangers and scouts for reconnaissance and surveillance. *IEEE Robotics and Automation Magazine*, *7*(4), 14–24.

Savkin, A., Evans, R., & Skafidas, E. (2000). The problem of optimal robust sensor scheduling. In *Proceedings of the 39th conference on decision and control* (Vol. 1). CDC. Sydney, Australia.

Sinopoli, B., Schenato, L., Franceschetti, M., Poolla, K., Jordan, M., & Sastry, S. (2004). Kalman filtering with intermittent observations. *IEEE Transactions on Automatic Control*, *49*(9), 1453–1464.

Tiwari, A., Jun, M., Jeffcoat, D. E., & Murray, R. M. (2005). The dynamic sensor coverage problem. In *Proceedings of the 2005 IFAC world congress.*

Viswanathan, R., & Varshney, P. K. (1997). Distributed detection with multiple sensors: Part I—fundamentals. *Proceedings of the IEEE*, *85*(1), 54–63.

Weisbin, C. R., Blitch, J., Lavery, D., Krotkov, E., Shoemaker, C., Matthies, L., & Rodriguez, G. (1999). Miniature robots for space and military missions. *IEEE Robotics and Automation Magazine*, *6*(3), 9–18.

Zhao, F., Shin, J., & Reich, J. (2002). Information-driven dynamic sensor collaboration for tracking applications. *IEEE Signal Processing Magazine*, *19*(2), 61–72.

Vijay Gupta received his B.Tech degree from Indian Institute of Technology, Delhi, India in 2001 and his M.S. degree from California Institute of Technology in 2002, both in Electrical Engineering. He is currently a Ph.D. candidate at the California Institute of Technology, Pasadena. His research interests include networked control systems, distributed estimation and detection, usage-based value of information and the interaction of communication, computation and control.

Timothy H. Chung received the B.S. degree in Mechanical and Aerospace Engineering from Cornell University, Ithaca, NY in 2001, and the M.S. degree in 2002 in Mechanical Engineering from the California Institute of Technology, Pasadena, CA, where he is currently a doctoral candidate.

His current research interests include distributed sensing, cooperative robotics, intelligent systems and decision-making.

Babak Hassibi was born in Tehran, Iran, in 1967. He received the B.S. degree from the University of Tehran in 1989, and the M.S. and Ph.D. degrees from Stanford University in 1993 and 1996, respectively, all in Electrical Engineering. From October 1996 to October 1998 he was a research associate at the Information Systems Laboratory, Stanford University, and from November 1998 to December 2000 he was a Member of the Technical Staff in the Mathematical Sciences Research Center at Bell Laboratories, Murray Hill, NJ. Since January 2001 he has been with the Department of Electrical Engineering at the California Institute of Technology, Pasadena, CA, where he is currently an associate professor. He has also held short-term appointments at Ricoh California Research Center, the Indian Institute of Science, and Linkoping University, Sweden. His research interests include wireless communications, robust estimation and control, adaptive signal processing and linear algebra. He is the coauthor of the books *Indefinite Quadratic Estimation and Control*: *A Unified Approach to* H^2 *and* H^∞ *Theories* (New York: SIAM, 1999) and *Linear Estimation* (Englewood Cliffs, NJ: Prentice-Hall, 2000). He is a recipient of an Alborz Foundation Fellowship, the 1999 O. Hugo Schuck best paper award of the American Automatic Control Council, the 2002 National Science Foundation Career Award, the 2002 Okawa Foundation Research Grant for Information and Telecommunications, the 2003 David and Lucille Packard Fellowship for Science and Engineering and the 2003 Presidential Early Career Award for Scientists and Engineers (PECASE). He is currently an Associate Editor for Communications of the IEEE Transactions on Information Theory, as well as an Editor for Foundations and Trends in Communications and Information Theory.

Richard M. Murray received the B.S. degree in Electrical Engineering from California Institute of Technology in 1985 and the M.S. and Ph.D. degrees in Electrical Engineering and Computer Sciences from the University of California, Berkeley, in 1988 and 1991, respectively. He is currently a Professor of Control and Dynamical Systems at the California Institute of Technology, Pasadena. Murray's research is in the application of feedback and control to mechanical, information, and biological systems. Current projects include integration of control, communications, and computer science in multi-agent systems, information dynamics in networked feedback systems, analysis of insect flight control systems, and synthetic biology using genetically encoded finite state machines.