Proceedings of the
46th IEEE Conference on Decision and Control
New Orleans, LA, USA, Dec. 12-14, 2007

Sensor Scheduling using Smart Sensors

Pedram Hovareshti, Vijay Gupta and John S. Baras

Abstract— **The sensor selection problem arises when multiple sensors are jointly trying to estimate a process but only a subset of them can take and/or use measurements at any time step. In a networked estimation situation, sensors are typically equipped with some memory and processing capabilities. We illustrate that utilization of these capabilities can lead to significant performance gains in the sensor selection problem for improved estimation performance. Further, it also leads to significant pruning of the search tree that yields the optimum sensor schedule. We also present a periodicity result for the case where the decision is whether the sensor should transmit or not.**

I. 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 [9] 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.

Communication constraints, however, often impose a restriction on the maximum number of sensors that can transmit data to the estimator. Thus, there is a problem of sensor scheduling. One example when such a situation arises is when there are echo-based sensors like sonars which can interfere with each other. Another situation where sensor scheduling is relevant is in tracking and discrimination problems, where a radar can make different types of measurements by transmitting suitable waveforms, 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, the sensor scheduling problem has received considerable attention in the literature. The seminal work in [12] proved a separation property between the optimal plant control policy and the measurement control policy for LQ control. 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. It was proven that if the decision to choose a particular sensor rests with the estimator, an open-loop selection strategy is optimal for a cost based on the estimate error covariance. Forward dynamic programming and a gradient method were proposed for this purpose. To deal with the complexity of a tree-search, greedy algorithms have been proposed many times, some examples being [13], [17]. Allied contributions have dealt with robust sensor scheduling [1] and the works of [15], [16], [18] etc. A different numerical approach to solve the problem was provided in [2], [10], [14] which cast the problem as a two-point boundary value problem. A completely general framework for nonlinear systems and general nonlinear diffusion sensor signals was developed in the seminal paper [3]. The dynamic sensor scheduling problem was solved using dynamic programming methods, based on general stochastic control separation and nonlinear filtering, which involved quasi-variational inequality techniques for the analytical proofs. A stochastic algorithm that is particularly useful in situations where communication channels impose random data dropouts was proposed in [4].

However, these approaches assume that a sensor, when allowed to transmit at time step k, transmits only the latest measurement that it observed at time step k. Thus, even if all sensors are taking measurements at every time step, the estimator does not have access to all this information. A notable exception is the general framework and methods of [3], where the estimator has complete past histories of measurements, and where even simultaneous measurements by several sensors in each time step are allowed. In networked control systems, sensors are usually equipped to communicate over wireless channels or communication networks. Thus, it is reasonable to assume that they possess some storage and processing capabilities. Thus, if the sensors can execute simple recursive algorithms to process the information being collected, significant improvement in estimation (or control) performance can be expected. Such algorithms have already been demonstrated for the case of single sensor systems in [5], [6]. In a companion paper [7], we illustrate the improvement in the stability region using such pre-processing strategies for multi-sensor systems. In this paper, we use information processing algorithms along the lines of the ones proposed in [8] for the sensor scheduling problem. Using these information processing algorithms, we show that we obtain significantly better estimates. We also consider the problem of finding the optimal sensor schedule. While the general solution remains a tree-search, we show that the number of paths to be searched are significantly pruned. We also prove a periodicity result in the optimal sensor schedules.

The paper is organized as follows. The next section deals with the problem formulation. We then present a simple recursive yet optimal information processing algorithm to be followed by the sensors. In Section IV, we consider the prob-

Research supported by the U.S. Army Research Office under MURI01 Program Grant No. DAAD19-01-1-0465, under Grant No. DAAD19-02-1-0319, and under Grant No DAAD 19-01-1-0494.

The authors are with the Institute for Systems Research, University of Maryland College Park, MD 20742, USA.(hovaresp@isr.umd.edu, vijay@cds.caltech.edu, baras@isr.umd.edu)

lem of optimal scheduling. Finally, in Section V, we present a special case when the decision (selection) is between a sensor transmitting or not, and present a periodicity result. The result also applies to more general scenarios. Because of space constraints, we omit the proofs here. The detailed proofs can be found in [11].

II. Modeling and Problem Formulation

Consider a system evolving as

$$x(k+1) = Ax(k) + w(k), \tag{1}$$

where $x(k) \in \mathbf{R}^n$ is the process state at time step k and $w(k)$ is the process noise assumed white, Gaussian and zero mean with covariance matrix R_w. The initial condition $x(0)$ is assumed independent of the process noise and Gaussian with zero mean and covariance P_0. The process state is being observed by N sensors $S_1,\ S_2,\ \cdots,\ S_N$ with the measurement equation for the i-th sensor being

$$y_i(k) = C_i x(k) + v_i(k), \tag{2}$$

where $y_i(k) \in \mathbf{R}^{s_i}$ is the measurement. The measurement noises $\{v_i(k),\ i = 1,\ \cdots,\ N\}$, for the sensors are assumed independent of each other, of the process noise and of the initial condition. Further the noise $v_i(k)$ is assumed to be white, Gaussian and zero mean with covariance matrix R_i. In this paper, we will assume $N = 2$ for ease of exposition. The ideas are applicable to the general case, at the expense of more notation. We assume that the pair (A, C) is observable and the pair $(A, R_w^{\frac{1}{2}})$ is stabilizable, where $C = \begin{bmatrix} C_1^T & C_2^T \end{bmatrix}^T$.

At every time step k, one sensor is chosen to take the measurement. The assumption of one sensor being allowed per time step is without loss of generality. If the i-th sensor is chosen at time k, we represent this event as $t(k) = i$. By a sensor schedule, we mean the choice of events $t(0)$, $t(1)$, $\cdots$. The i-th sensor then calculates a finite vector

$$s_i(k) = f(i, k, y_i(0), \cdots, y_i(k), t(0), \cdots, t(k)),$$

where $s_i(k) \in \mathbf{R}^m$ and transmits it to a central estimator (equivalently, shared with all the sensors) in an error-free manner. By abusing the notation a bit, we denote by $s(k)$ the vector received by the estimator at time step k. The estimator calculates an estimate

$$\hat{x}(k+1) = g\left(k, s(0), s(1), \cdots, s(k)\right)$$

of $x(k+1)$ that minimizes the usual mean squared error

$$P(k+1) = E\left[e(k)e^T(k)\right]$$

where $e(k)$ is the error defined as $e(k) = x(k+1) - \hat{x}(k+1)$.

To compare particular encoding functions $f()$ and decoding functions $g()$, we use the finite-horizon cost

$$J_K = \sum_{k=1}^{K} trace(P(k)),$$

or the infinite-horizon cost

$$J_\infty = \lim_{K\to\infty} \frac{1}{K} \sum_{k=1}^{K} trace(P(k)).$$

In this paper, we are concerned with the following problems:

1) What are the functions f and g that are optimal for above cost functions for any schedule of the sensors?
2) What is the optimal sensor schedule for the infinite-horizon cost? We will be interested in open loop schedules where the choice of the event $t(k)$ does not depend on the measurement values $\{y_i(k),\ i = 1,\ \cdots,\ N\}$.
3) For the special case when the sensing choices consist of transmitting a measurement by the sensor or not transmitting one, what is the optimal schedule for transmitting measurements for the finite-horizon cost?

III. Optimal Encoding and Decoding Functions

At time k, define the time-stamp for sensor i as

$$\tau_i(k) = \max\{j \mid j \le k, \quad t(j) = i\}.$$

Thus the time-stamp denotes the latest time at which transmission was possible from sensor i. Using the time-stamp, define the maximal information set $\mathcal{I}_i^{\max}(k)$ for sensor i as $\mathcal{I}_i^{\max}(k) = \{y_i(0), y_i(1), \cdots, y_i(\tau_i(k))\}$. The maximal information set is the largest set of measurements from sensor i that the controller can possibly have access to at time k. For any encoding functions f chosen by the sensors, the information available at the estimator will be a sub-set of the maximal information set. Hence, with the optimal minimum mean squared error (MMSE) estimation being chosen as the decoding function g by the decoder, the performance for any encoding functions f will be upper bounded (equivalently, the cost will be lower bounded) if the estimator had access to the maximal information sets from all the sensors.

Now consider an algorithm $\bar{\mathcal{A}}$ under which at every time step k, if $t(k) = i$, sensor i transmits the set $S_i(k) = \{y_i(0), y_i(1), \cdots, y_i(k)\}$. Note that the algorithm $\bar{\mathcal{A}}$ does not specify valid encoding functions since the dimension of the transmitted vectors cannot be bounded by any constant m. However, if the algorithm $\bar{\mathcal{A}}$ is followed, at any time step k, the decoder (and the controller) would have access to the maximal information sets $\mathcal{I}_i^{\max}(k)$. This implies that for any other encoding algorithm, the cost will always be higher for any given schedule than obtained by using the algorithm $\bar{\mathcal{A}}$. Thus, in particular, one way to achieve the optimal value of the cost J_K or J_∞ for a given schedule is through the combination of an encoding algorithm that makes the information sets $\mathcal{I}_i^{\max}(k)$ available to the controller and a controller that optimally utilizes the information set. Further, one such information processing algorithm is the algorithm $\bar{\mathcal{A}}$ described above. However, this algorithm requires increasing data transmission as time evolves. Surprisingly, in a lot of cases, we can achieve the same performance using a constant amount of transmission and memory.

To this end, we begin with a result proven in [5], [7]. This result identifies the optimal information processing to be done by the sensors to ensure that the estimator can calculate the estimate of state $x(k+1)$ based on the maximal information sets $\mathcal{I}_i^{\max}(k)$.

Proposition 1: Consider a process of the form (1) being observed by two sensors of the form (2). The estimate $\hat{x}(k|l,m)$ of the state based on measurements from sensor 1 till time l and sensor 2 till time m (and the corresponding error covariance $P(k|l,m)$) can be calculated using the algorithm given below. Assume, without loss of generality, that $l \leq m$. Let $\hat{x}_i(k|l)$ denote the MMSE estimate of $x(k)$ based on all the measurements of sensor i up to time l. Denote the corresponding error covariance by $P_i(k|l)$.

- At each time step $j \leq k$, the sensor 1 executes the following actions:
  1) Obtain the estimate $\hat{x}_1(j|j)$ and $P_1(j|j)$ through a Kalman filter. For $j \leq l$, use the measurement $y_1(j)$. For $j > l$, assume that the sensor 1 did not take any measurement at time step j.
  2) Calculate
  $$\lambda_1(j) = \left(P_1(j|j)\right)^{-1} \hat{x}_1(j|j) - \left(P_1(j|j-1)\right)^{-1} \hat{x}_1(j|j-1).$$
  3) Calculate global error covariance matrices $P(j|j,j)$ and $P(j|j-1,j-1)$ using the relation
  $$\left(P(j|j,j)\right)^{-1} = \begin{cases} \left(P(j|j-1,j-1)\right)^{-1} + C_1^T \left(\Sigma_{v,1}\right)^{-1} C_1 \\ +C_2^T \left(\Sigma_{v,2}\right)^{-1} C_2 & \text{if } j \leq l \\ \left(P(j|j-1,j-1)\right)^{-1} \\ +C_2^T \left(\Sigma_{v,2}\right)^{-1} C_2 & \text{if } l < j \leq m \\ \left(P(j|j-1,j-1)\right)^{-1} & \text{otherwise,} \end{cases}$$
  $$P(j|j-1,j-1) = AP(j-1|j-1,j-1)A^T + \Sigma_w.$$
  4) Obtain
  $$\gamma(j) = \left(P(j|j-1,j-1)\right)^{-1} A \, P(j-1|j-1,j-1).$$
  5) Finally calculate
  $$I_{1,l,m}(j) = \lambda_1(j) + \gamma(j) I_{1,l,m}(j-1),$$
  with $I_{1,l,m}(-1) = 0$.
- The quantity $I_{2,l,m}(k)$ is calculated by a similar algorithm except using the local estimates $\hat{x}_2(j|j)$ and covariance $P_2(j|j)$.
- Finally, the estimate $\hat{x}(k|l,m)$ is calculated as
$$\left(P(k|k,k)\right)^{-1} \hat{x}(k|l,m) = I_{1,l,m}(k) + I_{2,l,m}(k).$$

The above result identifies the quantities that need to be transmitted by the two sensors to calculate the MMSE estimate of $x(k)$. The quantities depend only on local measurements at the sensors; however, an implicit assumption is that each sensor is informed about the times l and m. We now present an algorithm according to which the sensors can calculate these optimal vectors with constant memory and processing for any given schedule. We present the algorithm $\mathcal{A}_1$ that the 1st sensor needs to implement. The algorithm $\mathcal{A}_2$ for the second sensor is similar.

Algorithm $\mathcal{A}_1$ to be followed by sensor 1: The sensor maintains two vectors $I^1_{1,k,\alpha_2(k)}(k)$ and $I^2_{1,k,k}(k)$.

1) *Initialization:* Initialize both the vectors $I^1_{1,-1,\alpha_2(-1)}(-1)$ and $I^2_{1,-1,-1}(-1)$ to be zero.
2) *Update and Transmission:* At every time step $k \geq 0$, there are two cases:
   - Sensor 1 transmits at time step k: It takes the following actions:
     – It updates vector $I^1_{1,k-1,\alpha_2(k-1)}(k-1)$ to calculate $I^1_{1,k,\alpha_2(k)}(k)$ using an algorithm of the form mentioned in Proposition 1, where $\alpha_2(k) = \alpha_2(k-1)$. It then transmits this vector.
     – It updates the vector $I^2_{1,k,k}(k)$ from $I^2_{1,k-1,k-1}(k-1)$ using an algorithm of the form mentioned in Proposition 1.
   - Sensor 2 transmits at time step k: Sensor 1 takes the following actions:
     – It updates the vector $I^2_{1,k,k}(k)$ from $I^2_{1,k-1,k-1}(k-1)$ using an algorithm of the form mentioned in Proposition 1.
     – It resets $I^1_{1,k,\alpha_2(k)}(k) = I^2_{1,k,k}(k)$.

For this algorithm, it can be verified that

1) The index $\alpha_2(k)$ is always equal to the last time $m \leq k$ where sensor 2 was able to transmit.
2) All the update steps at time k require only the knowledge of the latest measurement from sensor 1 $y_1(k)$. Thus, constant memory and processing are required.

These two observations allow us to state the following result.

Proposition 2: Consider the problem formulation stated in Section II. Using the transmitted vectors $I^1_{1,k,\alpha_2(k)}(k)$ and $I^2_{2,\alpha_1(l),l}(l)$ from the two sensors, the estimator can construct the MMSE estimate of $x(k+1)$ using all the measurements from sensor 1 till time k and from sensor 2 till time l. Further, the vectors can be calculated by the sensors using constant amount of processing, memory and transmission at every time step using algorithms $\mathcal{A}_1$ and $\mathcal{A}_2$.

The algorithm we have outlined is optimal among all other causal encoding algorithms, in the sense that for any given schedule of transmission, the cost J_K achieved at any time K is minimum for this algorithm. It can also be extended to consider the effect of stochastic packet drops by communication channels from the sensors to the estimator.

IV. Optimal Scheduling

In this section, we look at designing an optimal schedule, i.e., the choice of the events $t(k)$ at every time step k. We begin by considering the finite horizon cost J_K. We first note that for the optimal encoding and decoding functions that we have identified in Section II, the proof of optimality of open loop schedules [12] can directly be carried over. In other words, the optimal open loop schedule, in which the choice of $t(k)$ depends only on the system parameters, yields the same performance as the optimal closed loop schedule, in which $t(k)$ can additionally depend on the choice of events $t(0), t(1), \cdots, t(k-1)$. Thus, from now on, we will consider obtaining the optimal open loop schedule.

All the possible sensor schedule choices can be represented by a tree. The depth of a node in the tree represents time instants with the root representing time zero. The branches correspond to choosing a particular sensor to be active at that time instant. Each node is associated with the cost evaluated using the sensor schedule corresponding to the path from the root to that node. 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 d, a total of 2^d schedules need to be compared. This procedure might place too high a demand on the computational and memory resources of the system. We will now see that with the optimal encoding and decoding functions, we can prune the tree significantly.

Consider a time k when the estimation error covariance incurred in estimating $x(k+1)$ using the measurements of *both* the sensors till time step k has reached a steady state value $P^\star$. The steady-state exists and is reached exponentially because of our observability assumptions. For simplicity, we will assume that the horizon K is long enough so that the cost incurred in the transient phase is small and can be ignored during the optimization. Equivalently, we can assume that the covariance of the initial state $P(0) = P^\star$. Thus, we can carry out the optimization by assuming that the steady-state has been reached.

We define the following Riccati operator for $i = 1, 2$:

$$h_i(P) = R_w + A\left(P - PC_i^T\left(C_iPC_i^T + R_i\right)^{-1}C_iP\right)A^T$$

The operator acts on a positive semi-definite matrix P and results in a value that equals the estimate error covariance at time step $k+1$ assuming that sensor i was used at time step k and the initial error covariance at time step k was P. We also define $h_i^t(P)$ as the operator h_i applied t times on P. We note that $h_i^t(P)$ is an increasing function in the index t for any positive semi-definite matrix P.

The key observation that allows us to prune the tree is the following. When the optimal encoding and decoding functions are employed by the sensors, the effect on the error covariance at the estimator is the same as if all previous measurements were also transmitted by each sensor whenever it was allowed to transmit. That is, if $t(k) = i$, the i-th sensor could be considered to be transmitting all measurements $y_i(0)$, $y_i(1)$, $\cdots$, $y_i(k)$. Thus, in the steady state, the error covariance at the estimator resets to $h_i(P^\star)$ whenever a switching from sensor j to sensor i happens. Moreover, if no further switching happens in an interval of length t the error covariance at the end of this interval will be $h_i^t(P^\star)$. This observation allows us to discard many sequences in the search tree and prune it significantly.

Proposition 3: Consider the problem formulation stated in Section II. Suppose that the optimal encoding and decoding functions, as identified in Section IV are being followed. Further, assume that the steady-state has been reached, so that the error covariance in estimating the state $x(m+1)$ based on all the measurements from both the sensors till time m is $P^\star$. Let the sensors be denoted by i and j. Suppose there exists $k > 0$ such that

- $\forall m = 1, ..., k-1, Trace(h_i^m(P^\star)) \leq Trace(h_j(P^\star))$
- $Trace(h_i^k(P^\star) > Trace(h_j(P^\star))$

Define two sub-sequences for selecting the sensors

$$\begin{aligned} S_1 &= \{t(n) = i, t(n+1) = i, \cdots, t(n+k-1) = i\} \\ S_2 &= \{t(m) = j, t(m+1) = j\}, \end{aligned}$$

for arbitrary times m and n. Then, the sub-sequences S_1 and S_2 can not appear in the optimal schedule.

The above result assumes the existence of the parameter k. If such a k does not exist, using sensor i at every time step is optimal. Such a case arises, e.g., when sensor i corresponds to a successful transmission and sensor j corresponds to an unsuccessful one. The issue of optimal sensor scheduling in that case is trivial, unless a bound on the number of times sensor i can be used is given. We shall consider the latter case in the next section.

Thus, we can prune all the branches that include the sequences S_1 and S_2 from the search tree. This gives us a significant decrease in the search space. However, the number of branches still remains exponential in the horizon length K. For a very large value of the horizon K, the complexity is still prohibitive. However, the case for a large enough K is practically identical to considering an infinite horizon cost. For the infinite-horizon cost, we have the following periodicity result that allows us to bypass the tree-search process altogether.

Proposition 4: Consider the problem formulation stated in Section II. Suppose that the optimal encoding and decoding functions, as identified in Section IV are being followed. Further, assume that the steady-state has been reached, so that the error covariance in estimating the state $x(m+1)$ based on all the measurements from both the sensors till time m is $P^\star$. Let the sensors be denoted by i and j. Suppose there exists $k > 0$ satisfying the two conditions in Proposition 3. Consider the optimal schedule for the infinite horizon case. Suppose that at time step m, sensor j is used. Further, let $n > 0$ be the smallest value such that at time $m+n$, sensor j is used again. Then the optimal schedule after time m is given by

$$t(l) = \begin{cases} j & \text{if } l = m + kn, \qquad k = 0, 1, 2, \cdots \\ i & \text{otherwise.} \end{cases}$$

V. Single sensor with bounded transmissions

The general framework considered in the previous sections facilitates the analysis of a single sensor scheduling in the presence of a bound on the number of transmissions. As argued in the previous section, in the case of a single sensor the issue of scheduling is trivial, unless there is a bound on the number of transmissions. Considering such bounds are important in applications which involve a trade-off between the accuracy of the estimate and the costs of using the sensors and communicating the information to the estimator.

The problem set up is as before except that now we only consider a single sensor observing the process. As before we assume that the steady-state has been reached. For the finite horizon case, denote the length of the horizon by K and the number of allowed transmissions by $c(K) < K$. Therefore the frequency of transmission is defined as $q_K = c(K)/K$. We consider the finite horizon problem of selecting the $c(K)$ time instants such that $t(k) = 1$. We denote the choice of 'not to transmit' at time k by $t(k) = \emptyset$. The algorithm for optimal encoding in this case reduces to the sensor maintaining and transmitting an estimate $\hat{x}(k)$ of the state $x(k)$ based on the measurements $y(0)$, $y(1)$, $\cdots$, $y(k)$. The process estimator updates its estimate $\hat{x}_{dec}(k)$ as

$$\hat{x}_{dec}(k) = \begin{cases} A\hat{x}(k) & \text{if } t(k) = 1, \\ A\hat{x}_{dec}(k-1) & \text{if } t(k) = \emptyset. \end{cases}$$

Consequently, the error covariance at the decoder evolves as:

$$P(k) = \begin{cases} P^\star & \text{if } t(k) = 1, \\ AP(k-1)A^T + Q & \text{if } t(k) = \emptyset, \end{cases}$$

where $P^\star$ is the steady state error covariance of the optimal estimate of $x(k)$ using all the measurements till time $k-1$.

We are interested in the following problem: Starting from an arbitrary time m when the last update happened, find the schedule that minimizes the cost function $\sum_{k=1}^{K} trace(P(m+k))$ subject to the fact that the maximum number of channels used is limited to $n = c(K)$. The following statement indicates that periodic transmission minimizes the cost function.

Proposition 5: Consider the problem formulation as stated above. Further, suppose that $j = \frac{K-n}{n+1}$ is an integer. Then, the optimal schedule is the periodic schedule

$$t(k) = \begin{cases} 1 & \text{if } k = m + i(j+1), \quad i = 1, 2, ..., n \\ \emptyset & \text{Otherwise.} \end{cases}$$

Remark: If j is not an integer, the time intervals between the sensors cannot be all made equal to j. However, by choosing the intervals as close to periodic as possible we can get the lowest possible cost.

VI. Simulation Results

In this section we illustrate the results, starting with the improvement in estimation cost using preprocessing. We consider the case of a simple model of two sensors trying to locate a noncooperative vehicle moving in a plane. The acceleration is equal to zero except for a small perturbation. Let p denote position and v denote speed. Then $x = [p_x \quad p_y \quad v_x \quad v_y]^T$ is the state and we consider a discretization step $h = 0.2$. Following the framework of Section II the state space model parameters are:

$$A = \begin{pmatrix} 1 & 0 & h & 0 \\ 0 & 1 & 0 & h \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}, B = \begin{pmatrix} h^2/2 & 0 \\ 0 & h^2/2 \\ h & 0 \\ 0 & h \end{pmatrix},$$

$$C = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{pmatrix}.$$

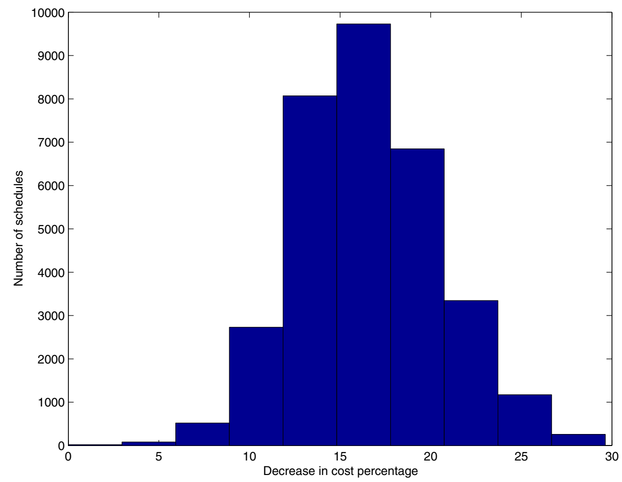

Fig. 1. Percentage decrease in J_K due to preprocessing. ($K = 15$)

The process and sensor covariances are considered to be

$$R_w = \begin{pmatrix} 0.0100 & 0 \\ 0 & 0.0262 \end{pmatrix},$$

$$R_1 = \begin{pmatrix} 0.0003 & 0 \\ 0 & 0.0273 \end{pmatrix}, \quad R_2 = \begin{pmatrix} 0.0018 & 0 \\ 0 & 0.0110 \end{pmatrix}.$$

Our first observation is that for *all* schedules, preprocessing lowers the cost. The amount of such decrease depends on the particular choice of a sensor schedule. Figure 1 shows a histogram of the distribution of this decrease for a small time horizon $K = 15$. It can be seen that more than half of the schedules will incur an improvement of 15% or more.

We also compared the optimal schedules determined with and without preprocessing for different time horizons. The optimal schedule using preprocessing always has a lower cost. Figure 2 shows the percentage of the decrease in optimal estimation cost due to preprocessing. We can see that even in this simple system, preprocessing results in more than 18% decrease in estimation cost.

It is worthwhile to note that the optimal schedule has a periodic structure as the horizon increases. The optimal schedules for different horizons are given in Table I. The trend remains the same for the values of $K \geq 20$.

K	$Optimal Schedule$
10	2212212212
11	22122122122
12	221221221222
13	2212212212212
14	22122122122122
15	221221221221222
16	2212212212212212
17	22122122122122122
18	221221221221221222
19	2212212212212212212
20	22122122122122122122

TABLE I
OPTIMAL SCHEDULES

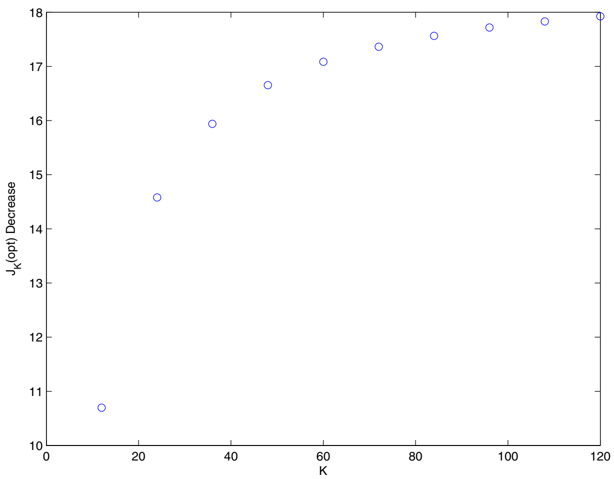

Fig. 2. Percentage of decrease in J_K for optimal schedule $(k \leq 120)$

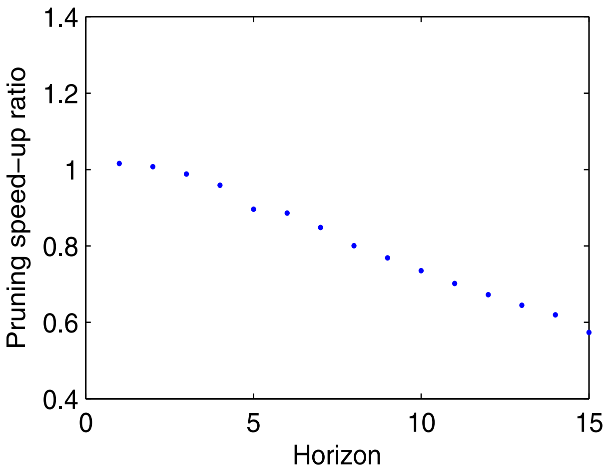

Fig. 3. CPU time reduction by pruning for $K \leq 15$

The proposed pruning method of section IV results in speed up in the search associated with the scheduling problem. We have measured this by the MATLAB commands 'tic' and 'toc' for the corresponding tree search routines. This is illustrated in Figure 3, where the ratio of the reduction in the CPU time is plotted for the range of horizon $K \leq 15$.

Figure 4 illustrates the case of a single sensor S_2. Here a time horizon of $K = 59$ is considered and the optimal cost is plotted as a function of utilization frequency. $K = 59$ is selected since this particular K results in j being integer for many choices of n. The estimation cost (error) is a decreasing function of sensor utilization. Therefore, the frequency of sensor utilization is determined by the trade off between the communication and estimation costs.

References

[1] A. Savkin, R. Evans and E. Skafidas, "The Problem of Optimal Robust Sensor Scheduling", *Proceedings of the 39th IEEE Conference on Decision and Control*, Vol. 4, pp. 3791-3796, Sydney, Australia, 2000.

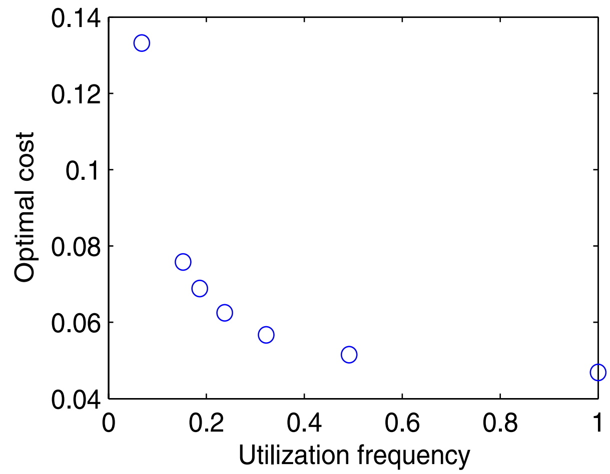

Fig. 4. Optimal cost in the single sensor case as a function of transmission frequency

[2] M. Athans, "On the Determination of Optimal Costly Measurement Strategies," *Automatica* **18**, pp. 397–412, 1972.

[3] J.S. Baras and A. Bensoussan, "Optimal Sensor Scheduling in Nonlinear Filtering of Diffusion Processes," *SIAM Journal on Control and Optimization*, Vol. 27(4), pp 786-813, 1989.

[4] V. Gupta, T. H. Chung, B. Hassibi and R. M. Murray, "A Stochastic Sensor Selection Algorithm with Applications in Sensor Scheduling and Sensor Coverage," *Automatica*, v. 42 (2), Feb. 2006, pp. 251-260.

[5] V. Gupta, B. Hassibi and R. M. Murray, "Optimal LQG Control Across Packet-Dropping Links," *System and Control Letters*, 56(6), June 1 2007, pp. 439-446.

[6] V. Gupta, A. F. Dana, J. Hespanha, R. M. Murray and B. Hassibi, "Data Transmission over Networks for Estimation and Control," *IEEE Transactions on Automatic Control*, submitted, December 2006.

[7] V. Gupta, N. Martins and J.S. Baras, "Observing a Linear Process over Packet Erasure Channels using Multiple Sensors: Necessary and Sufficient Conditions for Mean-square Stability," to appear in *Proceedings of 46th IEEE Conference on Decision and Control*, Dec. 2007.

[8] V. Gupta, N. Martins and J.S. Baras, "Stabilization over Erasure Channels using Multiple Sensors," *IEEE Transactions on Automatic Control*, submitted, February 2007.

[9] D.L. Hall and J. Llinas, "An Introduction to Multisensor Data Fusion," *Proceedings of the IEEE*, **85**(1), pp.6–23, 1997.

[10] K. Herring, and J. Melsa, "Optimum Measurements for Estimation," *IEEE Transactions on Automatic Control*, **19**(3),pp. 264–266, 1974.

[11] P. Hovareshti, V. Gupta and J. Baras, "On Sensor Scheduling using Smart Sensors," ISR Technical Report, University of Maryland, available at http://www.isr.umd.edu/research/TR_gateway.htm

[12] L. Meier, J. Peschon and R. Dressler, "Optimal Control of Measurement Subsystems," *IEEE Tr. on Aut. Control*, **12**,pp. 528–536, 1967.

[13] S. Kagami and M. Ishikawa, "A Sensor Selection Method Considering Communication Delays," *Proc. of the 2004 IEEE Intl. Conf. on Robotics and Automation*, Vol. 1, pp. 206–211, 2004.

[14] T.H. Kerr and Y. Oshman, "Further Comments on 'Optimal Sensor Selection Strategy for Discrete-time Estimators'[and reply]," *IEEE Trans. on Aerospace and Electronic Systems* **31**,pp. 1159 – 1167, 1995.

[15] V. Krishnamurthy, "Algorithms for Optimal Scheduling and Management of Hidden Markov Model Sensors," *IEEE Transactions on Signal Processing*, **60**(6),pp. 1382–1397, 2002.

[16] B.M. Miller, and W. J. Runggaldier, "Optimization of Observations: a Stochastic Control Approach," *SIAM Journal of Control and Optimization*, **35**(5), pp. 1030–1052, 1997.

[17] Y. Oshman, "Optimal Sensor Selection Strategy for Discrete-time State Estimators," *IEEE T. on Aer. and Electr. Sys.*, **30**, pp. 307–314, 1994.

[18] C. Rago, P. Willet and Y. Bar-Shalom, "Censoring Sensor: a Low Communication Rate Scheme for Distributed Detection," *IEEE Trans. on Aerospace and Electronic Systems*, **28**(2), pp. 554–568, 1996.