Stochastic sensor scheduling via distributed convex optimization

Chong Li [1], Nicola Elia

Department of Electrical and Computer Engineering, Iowa State University, Ames, IA, 50011, United States

ARTICLE INFO

ABSTRACT

In this paper, we propose a stochastic scheduling strategy for estimating the states of N discrete-time linear time invariant (DTLTI) dynamic systems, where only one system can be observed by the sensor at each time instant due to practical resource constraints. The idea of our stochastic strategy is that a system is randomly selected for observation at each time instant according to a pre-assigned probability distribution. We aim to find the optimal pre-assigned probability in order to minimize the maximal estimate error covariance among dynamic systems. We first show that under mild conditions, the stochastic scheduling problem gives an upper bound on the performance of the optimal sensor selection problem, notoriously difficult to solve. We next relax the stochastic scheduling problem into a tractable suboptimal quasi-convex form. We then show that the new problem can be decomposed into coupled small convex optimization problems, and it can be solved in a distributed fashion. Finally, for scheduling implementation, we propose centralized and distributed deterministic scheduling strategies based on the optimal stochastic solution and provide simulation examples.

1. Introduction

In this paper, we consider the problem of scheduling the observations of independent targets in order to minimize the tracking error covariance, but when only one target can be observed at a given time. This problem captures many interesting tracking/estimation application problems. As a motivational example, consider N independent dynamic targets, spatially distributed in an area, that need to be tracked (estimated) by a single (mobile) camera sensor. The camera has limited sensing range and therefore it needs to zoom in on, or be in proximity of, one of the targets for obtaining measurements. Under the assumption that the switching time among the targets is negligible, then we need to find a visiting sequence in order to minimize the estimate error.

Another case is when a set of N mobile surveillance devices need to track N geographically-separated targets, where each target is tracked by one assigned surveillance device. However, the sensing/measuring channel can only be used by one estimator at the time (e.g. sonar range-finding Cremean et al., 2002). Then, we need to design a scheduling sequence of surveillance devices for accurate tracking.

1.1. Related work and contributions of this paper

There has been considerable research effort devoted to the study of sensor selection problems, including sensor scheduling (Alriksson & Rantzer, 2005; Bitar, Baeyens, & Poolla, 2009; Gupta, Chung, Hassibi, & Murray, 2006; Joshi & Boyd, 2009; Li & Elia, 2011; Liang, Tang, & Zhu, 2007; Lin & Wang, 2013; Ny, Feron, & Dahleh, 2011; Shi, Johansson, & Murray, 2007; Srivastava, Plarre, & Bullo, 2011; Tiwari, Jun, Jeffcoat, & Murray, 2005; Vasanthi & Annadurai, 2006) and sensor coverage (Acar & Choset, 2002; Choset, 2001; Cortes, 2010; Cortes, Martinez, Karatas, & Bullo, 2004; Gupta et al., 2006; Hussein & Stipanovi, 2007). This trend has been inspired by the significance and wide applications of sensor networks. As the literature is vast, we list a few results which are relevant to this paper. The sensor scheduling problem mainly arises from minimization of two relevant costs: sensor network energy consumption and estimate error. On the one hand, (Liang et al., 2007; Vasanthi & Annadurai, 2006) and (Bitar et al., 2009), see also reference therein, have proposed various efficient sensor scheduling algorithms to minimize the sensor network energy consumption and consequently maximize the network lifetime. On the other hand, researchers have proposed many tree-search based

sensor scheduling algorithms (mostly in conjunction with Kalman filtering) to minimize the estimate error (Alriksson & Rantzer, 2005; Lin & Wang, 2013; Tiwari et al., 2005), e.g. sliding-window, thresholding, relaxed dynamic programming, etc. By taking both sensor network lifetime and estimate accuracy into account, several sensor tree-search based scheduling algorithms have been proposed in Shi et al. (2007), Shi, Capponi, Johansson, and Murray (2008). In Joshi and Boyd (2009), the authors have formulated the general sensor selection problem and solved it by relaxing it to a convex optimization problem. The general formulation therein can handle various performance criteria and topology constraints. However, the framework in Joshi and Boyd (2009) is only suitable for static systems instead of dynamic systems which are mostly considered in the literature.

In general, deterministic optimal sensor selection problems are notoriously difficult. In this paper, we propose a stochastic scheduling strategy. At each time instant, a target is randomly chosen to be measured according to a pre-assigned probability distribution. We find the optimal pre-assigned distribution that minimize an upper bound on the expected estimate error covariance (in the limit) in order to keep the actual estimate error covariance small. Compared with algorithms in the literature, this strategy has low computational complexity, it is simple to implement and provides performance guarantee on the general deterministic scheduling problem. Of course, the reduction of computational complexity comes at the expenses of degradation of the ideal performance. However, in many situations the extra computational complexity cost may not be justified. Further this strategy can easily incorporate extra constraints on the scheduling design, which might be difficult to handle in existing algorithms (e.g. tree-search based algorithms).

Our work is related to Gupta et al. (2006), Mo, Garone, Casavola, and Sinopoli (2011) and Ny et al. (2011). Gupta et al. (2006) introduces stochastic scheduling to deal with sensor selection and coverage problems, and Mo et al. (2011) extends the setting and results in Gupta et al. (2006) to a tree topology. Although we also adopt the stochastic scheduling approach, the problem formulation and proposed algorithms of this paper are different from Gupta et al. (2006), Mo et al. (2011). In particular, we consider different cost functions and design distributed algorithms that provide optimal probability distributions. Ny et al. (2011) has considered a scheduling problem in continuous-time and proposed a tractable relaxation, which provides a lower bound on the achievable performance, and an open-loop periodic switching strategy to achieve the bound in the limit of arbitrarily fast switching. However, besides the difference in the formulations, their approach does not appear to be directly extendable to the discrete-time setting. In summary, our main contributions include:

(1) We obtain a stochastic scheduling strategy with performance guarantee on the general deterministic scheduling problem by solving distributed optimization problems.
(2) For scheduling implementation, we propose both centralized and distributed deterministic scheduling strategies.

1.2. Notations and organization

Throughout the paper, A' is the transpose of matrix A. $Ones(n, n)$ implies an $n \times n$ matrix with 1 as all its entries. $Diag(V)$ denotes a diagonal matrix with vector V as its diagonal entries. $M \succeq 0$ (or $M \in S_+$) and $M \succ 0$ (or $M \in S_{++}$) respectively implies matrix M is positive semi-definite and positive definite where S_+ and S_{++} represent the positive semi-definite and positive definite cones. For a matrix A, if the block entry $A_{ij} = A'_{ji}$, we use $(\cdot)$ in the matrix to present block A_{ji}. The trace of a square matrix is denoted by $Tr(\cdot)$.

The paper is organized as follows. In Section 2, we mathematically formulate the stochastic scheduling problem. In Section 3,

we develop an approach and a distributed computing algorithm to solve the optimization problem. In Section 4, we present some further results and the extensions of our model. In Section 5, we consider the scheduling implementation problem. At last, we present simulations to support our results.

2. Sensor scheduling problem setup

Consider a set of N DTLTI systems (targets) evolving according to the equations

$$x_i[k + 1] = A_i x_i[k] + w_i[k] \quad i = 1, 2, \ldots, N \tag{1}$$

where $x_i[k] \in \mathbb{R}^{n_i}$ is the process state vector and $w_i[k] \in \mathbb{R}^{n_i}$ is assumed to be an independent Gaussian noise with zero mean and covariance matrix $Q_i \succ 0$. The initial state $x_i[0]$ is assumed to be an independent Gaussian random variable with zero mean and covariance matrix $\pi_i[0]$. In practice, each DTLTI system modeled above may represent the dynamic change of a local environment, the trajectory of a mobile vehicle, the varying states of a manufactory machine, etc. As a result of the sensor's limited range of sensing or the congestion of the sensing channel, at time instant k, only one system can be observed as

$$\tilde{y}_i[k] = \xi_i[k](C_i x_i[k] + v_i[k]) \tag{2}$$

where $\xi_i[k]$ is the indicator function indicating whether or not the system i is observed at time instant k, and accordingly we have constraint[2] $\sum_{i=1}^{N} \xi_i[k] = 1$. $v_i[k] \in \mathbb{R}^{p_i}$ is the measurement noise, which is assumed to be independent Gaussian with zero mean and covariance matrix $R_i \succ 0$.

Assumption 1. For all $i \in \{1, 2, \ldots, N\}$, the pair $(A_i, Q_i^{1/2})$ is controllable and the pair (A_i, C_i) is detectable.

Denote $\hat{x}_i[k]$ as the estimate at time k, obtained by a causal estimator for system i, which depends on the past and current observations $\{\tilde{y}_i[j]\}_{j=1}^{k}$. We begin by considering problem of minimizing (in the limit) the maximal estimate error. The problem can be formulated mathematically as

$$\min_{\hat{x}_i, \{\xi_i[j]\}_{j=1}^{\infty}} \max_i \left(\limsup_{T \to \infty} \frac{1}{T} \sum_{k=1}^{T} \mathbb{E}[(x_i[k] - \hat{x}_i[k])'(x_i[k] - \hat{x}_i[k])] \right)$$

$$\text{s.t.} \quad Equation : (1), (2), \quad i = 1, \ldots, N, \tag{3}$$

$$\sum_{i=1}^{N} \xi_i[k] = 1.$$

As the DTLTI systems are assumed to be evolving independently, then for a fixed $\{\xi_i[j]\}_{j=1}^{\infty}$ the optimal estimator for minimizing the estimate error covariance of system i ($i = 1, 2, \ldots, N$) is given by a Kalman filter[3] whose process of prediction and update are presented as follows (Sinopoli et al., 2004). Firstly we define

$$\hat{x}_i[k|k] \triangleq \mathbb{E}[x_i[k] | \{\tilde{y}_i[j]\}_{j=1}^{k}]$$

$$P_i[k|k] \triangleq \mathbb{E}[(x_i[k] - \hat{x}_i[k|k])(x_i[k] - \hat{x}_i[k|k])' | \{\tilde{y}_i[j]\}_{j=1}^{k}]$$

$$\hat{x}_i[k + 1|k] \triangleq \mathbb{E}[x_i[k + 1] | \{\tilde{y}_i[j]\}_{j=1}^{k}]$$

$$P_i[k + 1|k] \triangleq \mathbb{E}[(x_i[k + 1] - \hat{x}_i[k + 1|k])(x_i[k + 1] - \hat{x}_i[k + 1|k])' | \{\tilde{y}_i[j]\}_{j=1}^{k}].$$

---

[2] If we assume *at most* one target is chosen to be measured at each time instant, then we have $\sum_{i=1}^{N} \xi_i \leq 1$. Without loss of generality, in this paper we consider the case that one out of N targets must be chosen at each time instant.

[3] This indicates that N parallel estimators, i.e., Kalman filters, are used for estimating N independent DTLTI systems.

Then the Kalman filter evolves as

$$\hat{x}_i[k+1|k] = A_i\hat{x}_i[k|k]$$
$$P_i[k+1|k] = A_iP_i[k|k]A' + Q_i$$
$$\hat{x}_i[k+1|k+1]$$
$$\quad = \hat{x}_i[k+1|k] + \xi_i[k+1]K_i[k+1](y_i[k+1] - C_i\hat{x}_i[k+1|k])$$
$$P_i[k+1|k+1] = P_i[k+1|k] - \xi_i[k+1]K_i[k+1]C_iP_i[k+1|k]$$

where $K_i[k+1] = P_i[k+1|k]C_i'(C_iP_i[k+1|k]C_i' + R_i)^{-1}$ is the Kalman gain matrix. After straightforward derivation, we have the covariance of the estimate error evolving as

$$A_iP_i[k]A_i' + Q_i - \xi_i[k]A_iP_i[k]C_i'(C_iP_i[k]C_i' + R_i)^{-1}C_iP_i[k]A_i'$$
$$\quad = P_i[k+1] \tag{4}$$

where we use the simplified notation $P_i[k] = P_i[k|k-1]$. Note that the error covariance $P_i[k+1]$ is a function of sequence $\{\xi_i[j]\}_{j=1}^k$. Moreover, given $\{\xi_i[j]\}_{j=1}^k$, the evolution of the error covariance P_i is independent of the measurement values. Substituting the optimal estimator, the problem (3) is simplified into the following one.

Deterministic Scheduling Problem:

$$\mu_d = \min_{\{\xi_i[j]\}_{j=1}^\infty, i=[1..N]} \quad \max_i \left(\limsup_{T\to\infty} \frac{1}{T}\sum_{k=1}^T Tr(P_i[k]) \right)$$
$$\text{subject to} \quad Equation : (4), \quad \sum_{i=1}^N \xi_i[k] = 1. \tag{5}$$

This problem is notoriously difficult to solve. There are no known optimal alternatives to searching all possible schedules and then chosing the optimal one by complete comparison. However, the procedure is not computationally tractable in practice. Motivated by this fact, we present a stochastic scheduling strategy with advantages summarized below:

(1) The stochastic scheduling strategy provides an upper bound on the performance of the deterministic scheduling problem under mild conditions, as proved in Theorem 1 next.
(2) The stochastic scheduling problem can be easily relaxed into a convex optimization problem, which can be solved efficiently in a distributed fashion.
(3) The relaxed problem provides an open-loop strategy, which has low computing complexity and is simple to implement.
(4) Several practical constraints/considerations can be easily incorporated into the stochastic scheduling formulation, as discussed in Section 4.

2.1. Problem formulation: Stochastic scheduling strategy

First of all, we remove the dependence on time instant k and consider ξ_i as an independent and identically distribution (i.i.d) Bernoulli random variable with

$$\xi_i[k] = \begin{cases} 1 & \text{with probability } q_i \\ 0 & \text{with probability } 1 - q_i \end{cases} \quad i = 1, 2, \ldots, N \tag{6}$$

for all k, where q_i is the probability that the system i is observed at each time instant. As $\sum_{i=1}^N \xi_i = 1$, we have $\sum_{i=1}^N q_i = 1$. Then the stochastic scheduling strategy is that at each time instant a target (i.e., DTLTI system) is randomly chosen for measurements according to a pre-assigned probability distribution $\{q_i\}_{i=1}^N$.

Notice that the error covariance $P_i[k+1]$ is random as it depends on the randomly chosen sequence $\{\xi_i[j]\}_{j=1}^k$. Thus we need to evaluate the expected estimate error covariance in order to minimize the actual estimate error. Putting above together, we have the stochastic scheduling problem motivated by (5) as follows.

Stochastic Scheduling Problem

$$\mu_s = \min_{q_i, i=[1..N]} \quad \max_i Tr\left(\lim_{k\to\infty} \mathbb{E}_{\xi_{i,k}}[P_i[k]] \right)$$
$$\text{subject to} \quad Equation : (4), (6), \tag{7}$$
$$\sum_{i=1}^N q_i = 1, \quad 0 \le q_i \le 1,$$

where the expectation is w.r.t $\{\xi_i[1], \ldots \xi_i[k]\}$, which we denote by $\mathbb{E}_{\xi_{i,k}}$.

Theorem 1. *If A_i is invertible for $i = 1, \ldots, n$, the deterministic scheduling performance μ_d in (5) is almost surely upper bounded by the stochastic scheduling performance μ_s in (7).*

Proof. See Appendix. □

Remark 2. While Problem (5) provides an important motivation, Problem (7) is relevant on its own right and can be applied without the restriction on A_i being invertible.

2.2. Relaxation

Problem (7) involves the evolution of

$$\mathbb{E}_{\xi_{i,k}}[P_i[k]] = A_i\mathbb{E}_{\xi_{i,k-1}}[P_i[k-1]]A_i' + Q_i$$
$$\quad - q_i\mathbb{E}_{\xi_{i,k-1}}[A_iP_i[k-1]C_i'(C_iP_i[k-1]C_i' + R_i)^{-1}C_iP_i[k-1]A_i'].$$

Unfortunately, the right-hand side of the above expression is not easily computable, as it involves the expectation w.r.t. $\xi_{i,k-1}$ of a nonlinear recursive expression of $P_i[k-1]$. However, Sinopoli et al. (2004) has nicely shown that $\lim_{k\to\infty} \mathbb{E}_{\xi_{i,k}}[P_i[k]]$ is upper bounded by the fixed point $X_i \succeq 0$ of the following associated MARE,

$$MARE : X_i = A_iX_iA_i' + Q_i - q_iA_iX_iC_i'(C_iX_iC_i' + R_i)^{-1}C_iX_iA_i'. \tag{8}$$

This result motivates us to minimize $Tr(X_i)$ as a means to keep $Tr(\lim_{k\to\infty} \mathbb{E}_{\xi_{i,k}}[P_i[k]])$ itself small. Specifically, we consider the following optimization problem, denoted by OP, in the rest of the paper.

$$OP : \quad \min_{q_i, i=[1..N]} \quad \max_i Tr(X_i) \tag{9}$$
$$\text{subject to:} \quad \sum_{i=1}^N q_i = 1, \quad q_i^c < q_i \le 1, \quad i = 1, 2, \ldots, N$$

where X_i is an implicit function of q_i, defined by $X_i = g_{q_i}(X_i)$ and

$$g_{q_i}(X_i) = A_iX_iA_i' + Q_i - q_iA_iX_iC_i'(C_iX_iC_i' + R_i)^{-1}C_iX_iA_i'. \tag{10}$$

Remark 3. q_i^c is the critical value depending on the unstable eigenvalues of A_i, where the fixed point X_i exists if and only if the assigned probability $q_i > q_i^c$. We refer interested readers to Sinopoli et al. (2004), Mo and Sinopoli (2008) for the details on q_i^c. In this paper, we assume $\sum_{i=1}^N q_i^c < 1$, otherwise the above optimization problem has no feasible solution, i.e., the upper bound turns out to be infinity. For stable systems, we always have $q_i^c = 0$.

Note that the searching space of the above optimization problem is continuous, and it will be shown that the problem is (quasi)-convex. In addition, based on Theorem 1, the objective value of the above problem is an upper bound on the performance of the deterministic scheduling problem (5). In the rest of the paper, we provide efficient distributed algorithms to obtain the optimal solutions of OP.

3. Minimization of the maximal estimate error among targets

In this section, we first show that OP can be decoupled into N convex optimization problems, which can be solved separately. Then by utilizing the classical consensus algorithm, we propose a

distributed computing algorithm to obtain the optimal solution of OP. First of all, we recall some results on the MARE in Sinopoli et al. (2004) derived under the assumption $Q \succeq 0$.

Lemma 4. *Fix $q \in \mathbb{R}_{(q^c, 1]}$, for any initial condition $X_0 \succeq 0$,*

$$\lim_{k \to \infty} g_q^{(k)}(X_0) = \lim_{k \to \infty} \underbrace{g_q(g_q(\cdots g_q(X_0)))}_{k} = X$$

where X is the unique positive-semidefinite fixed point of the MARE, namely, $X = g_q(X)$.

Lemma 5. *For a given scalar q and a DTLTI system (A, C, Q, R) as described in (1) and (2), the fixed point X of the MARE presented in the form of (8) can be obtained by solving the following Linear Matrix Inequality (LMI) problem.*

$$\begin{array}{ll} \underset{X}{\operatorname{argmax}} & Tr(X) \\[2mm] \text{subject to} & \begin{bmatrix} AXA' - X + Q & \sqrt{q}AXC' \\ \sqrt{q}CXA' & CXC' + R \end{bmatrix} \succeq 0 \\[4mm] & X \succeq 0. \end{array} \tag{11}$$

Lemma 6. *Assume $X, Q \in S_+$, $R \in S_{++}$ and $(A, Q^{\frac{1}{2}})$ is controllable. Then the following facts are true.*

(1) $g_q(X) \succeq g_q(Y)$ if $X \succeq Y$.
(2) $g_{q_1}(X) \succeq g_{q_2}(X)$ if $q_1 \leq q_2$.
(3) $g_q(X) = \Phi_q(K, X)$ if $K = -AXC'(CXC' + R)^{-1}$.
(4) $g_q(X) = \min_K \Phi_q(K, X) \preceq \Phi_q(K, X), \forall K$.
(5) $g_q(\alpha X + (1 - \alpha)Y) \succeq \alpha g_q(X) + (1 - \alpha)g_q(Y), \forall \alpha \in [0, 1]$.
(6) *Define the linear operator*

$$L_q(Y) = (1 - q)(AYA') + qFYF'.$$

Suppose there exists $\bar{Y} \succ 0$ such that $\bar{Y} \succ L_q(\bar{Y})$. Then for all $W \succeq 0$

$$\lim_{k \to \infty} L_q^{(k)}(W) = 0$$

where

$$\begin{aligned} \Phi_q(K, X) = {} & (1 - q)(AXA' + Q) + q(A + KC)X(A + KC)' \\ & + qQ + qKRK'. \end{aligned} \tag{12}$$

Now we prove the monotonicity of the fixed point of the MARE, which will facilitate us to analyze OP.

Definition 7 (*Matrix-Monotonicity*). A function $f: \mathbb{R} \to \mathbb{S}_+$ is matrix-monotonic if for all $x, y \in \mathbf{dom}f$ with $x \leq y$, we have either $f(x) \preceq f(y)$ or $f(x) \succeq f(y)$ in the positive semidefinite cone $\mathbb{S}_+$.

Theorem 8 (*Matrix-Monotonicity of the MARE*). *For $\forall q \in \mathbb{R}_{(q^c, 1]}$, the fixed point of the MARE is matrix-monotonically decreasing w.r.t the scalar q.*

Proof. Assume $q^c < q_1 \leq q_2 \leq 1$ and X_1, X_2 satisfying $X_1 = g_{q_1}(X_1)$ and $X_2 = g_{q_2}(X_2)$. The existence of the fixed points X_1 and X_2 is guaranteed according to Lemma 4. We need to show $X_1 \succeq X_2$. Since $q_1 \leq q_2$, by using Lemma 6(2) we have

$$X_1 = g_{q_1}(X_1) \succeq g_{q_2}(X_1).$$

By Lemma 6(1), we have

$$\begin{aligned} X_1 & \succeq g_{q_2}(g_{q_2}(X_1)) \\ & \succeq g_{q_2}(g_{q_2}(g_{q_2}(X_1))) \\ & \succeq \cdots \\ & \succeq g_{q_2}^{(k)}(X_1). \end{aligned}$$

By the convergence property of the MARE (i.e., Lemma 4), we have $X_1 \succeq X_2$ by taking $k \to \infty$. $\square$

Remark 9. This theorem reveals an important message that, for any two different scalar q_1 and q_2, the corresponding fixed points can be ordered in the positive semidefinite cone. In other words, for a given system model (A, C, Q, R), the fixed points of the MARE w.r.t variable q are comparable. As we will see, this property is the foundation for deriving algorithms to solve OP.

Now, we are ready to analyze and solve OP. For the ease of reading, we occasionally use notation $X_i(q_i)$ to stress that X_i is a function of q_i, i.e., X_i is the fixed point of the MARE w.r.t the scalar q_i.

Corollary 10. *Problem OP is a quasi-convex optimization problem.*

Proof. Consider the cost function of OP. From Theorem 8, $Tr(X_i(q_i))$ is monotonically non-increasing in q_i, as the trace function is linear. Therefore, $Tr(X_i(q_i))$ is a quasi-convex function, for each $i = 1, \ldots, N$, due to the fact that any monotonic function is quasi-convex. Next, based on the fact that non-negative weighted maximum of quasi-convex functions preserves quasi-convexity, the result follows. $\square$

Next, we can rewrite problem OP in the following equivalent form

$$\begin{aligned} & \min_{q_i, i=[1..N], \gamma > 0} \quad \gamma \\ & \text{subject to} : \ Tr(X_i(q_i)) \leq \gamma, \\ & \sum_{i=1}^{N} q_i = 1, \quad q_i^c < q_i \leq 1, \ i = 1, 2, \ldots, N. \end{aligned} \tag{13}$$

The problem is in principle solved by bisecting γ and checking the feasible set is not empty. However the constraint set is not in a useful form yet. For any fixed γ, the feasible set is convex but not easy to work with, given the implicit functions $Tr(X_i(q_i))$.

It is then convenient to consider the following related problem for a given $\gamma > 0$.

$$\begin{aligned} \mu(\gamma) = {} & \min_{q_i, i=[1..N]} \quad \sum_{i=1}^{N} q_i \\ & \text{subject to:} \quad Tr(X_i(q_i)) \leq \gamma, \\ & \qquad\qquad q_i^c < q_i \leq 1, \quad i = 1, 2, \ldots, N. \end{aligned} \tag{14}$$

Lemma 11. *$\gamma > 0$ is feasible for Problem (13) if and only if Problem (14) is feasible and $\mu(\gamma) \leq 1$.*

Proof. Let $\mathcal{S}_\gamma$ denote the set of q_i's feasible for Problem (13). Assume $\mathcal{S}_\gamma$ is not empty. Then feasible set of Problem (14) is not empty, and since we know that there are q_i's such that $\sum_{i=1}^{N} q_i = 1$, then $\mu(\gamma) \leq 1$. For the other direction, assume that $\mu(\gamma) = 1$. Then $\mathcal{S}_\gamma$ is not empty, and therefore γ is feasible for Problem (13). If $\mu(\gamma) < 1$, then let $\alpha > 1$ such that $\alpha\mu(\gamma) = 1$, and consider $\tilde{q}_i = \alpha q_i$, for $i = 1, \ldots, N$. Then, $\sum_{i=1}^{N} \tilde{q}_i = 1, \tilde{q}_i > q_i^c$, and $Tr(X_i(\tilde{q}_i)) \leq Tr(X_i(q_i)) \leq \gamma$ because of Theorem 8. Thus, $\mathcal{S}_\gamma$ is not empty. Hence γ is feasible for Problem (13). $\square$

The cost of (14) is separable and the constraints are independent for each i. Thus, for any $\gamma > 0$, (14) can be solved by minimizing N independent problems, namely:

$$\begin{aligned} \mu(\gamma) = {} & \sum_{i=1}^{N} \quad \min_{q_i} \quad q_i \\ & \text{subject to:} \quad Tr(X_i(q_i)) \leq \gamma, \quad q_i^c < q_i \leq 1. \end{aligned} \tag{15}$$

It is easy to infer that $\mu(\gamma)$ in (15) is decreasing w.r.t. performance γ. Thus, OP can be solved by (15) using bisection on γ until $\mu(\gamma) = 1$.

We next concentrate on the subproblems:

$$\begin{aligned} q_i^{opt}(\gamma) = {} & \min_{q_i} \quad q_i \\ & \text{subject to:} \quad Tr(X_i(q_i)) \leq \gamma, \quad q_i^c < q_i \leq 1. \end{aligned} \tag{16}$$

Based on Theorem 8, we see that the optimal solution $q_i^{opt}(\gamma)$ of the problem (16) implies the smallest probability required for measuring system i for achieving the pre-assigned estimate performance γ. If the problem (16) is not feasible (e.g. γ is too small), we set $q_i^{opt}(\gamma) = 1$. Next, we show that the optimization problem (16) can be reformulated as the iteration of a Linear Matrix Inequality (LMI) feasibility problem. Without abuse of notation, we remove the subscript i since the following results apply to all DTLTI dynamic systems.

Lemma 12. *Assume that $(A, Q^{1/2})$ is controllable and (A, C) is detectable. For any given $q \in (q^c, 1]$ and invertible matrices Q and R, the following statements are equivalent:*

(1) $\exists \bar{X} \in S_{++}$ such that $\bar{X} = g_q(\bar{X})$.
(2) $\exists K$ and $X \in S_{++}$ such that $X \succ \Phi_q(K, X)$ (defined in (12)).
(3) $\exists H$ and $G \in S_{++}$ such that $\Gamma_q(H, G) \succ 0$.

where

$$\Gamma_q(G, H) = \begin{bmatrix} G & \sqrt{1-q}GA & \sqrt{q}(GA+HC) & \sqrt{q}H & G \\ (\cdot)' & G & 0 & 0 & 0 \\ (\cdot)' & (\cdot)' & G & 0 & 0 \\ (\cdot)' & (\cdot)' & (\cdot)' & R^{-1} & 0 \\ (\cdot)' & (\cdot)' & (\cdot)' & (\cdot)' & Q^{-1} \end{bmatrix} \quad (17)$$

Proof. (1) $\Rightarrow$ (2). According to Lemma 6(3), we have $\bar{X} = g_q(\bar{X}) = \Phi_q(K_{\bar{X}}, \bar{X})$ or $\bar{X} = (1-q)A\bar{X}A' + q(A+KC)\bar{X}(A+KC)' + Q + qKRK'$ with $K_{\bar{X}} = -A\bar{X}C'(C\bar{X}C' + R)^{-1}$. Then $\bar{X} \succ 0$ since $Q \succ 0$ and the other terms are $\succeq 0$. Moreover,

$$\begin{aligned} 2\bar{X} &= 2\Phi_q(K_{\bar{X}}, \bar{X}) \\ &= (1-q)A(2\bar{X})A' + q(A + K_{\bar{X}}C)(2\bar{X})(A + K_{\bar{X}}C)' \\ &\quad + 2Q + 2qK_{\bar{X}}RK'_{\bar{X}} \\ &\succ (1-q)A(2\bar{X})A' + q(A + K_{\bar{X}}C)(2\bar{X})(A + K_{\bar{X}}C)' \\ &\quad + Q + qK_{\bar{X}}RK'_{\bar{X}} \\ &= \Phi_q(K_{\bar{X}}, 2\bar{X}). \end{aligned} \quad (18)$$

The inequality follows from the fact that $Q \succ 0$ and $K_{\bar{X}}RK'_{\bar{X}} \succeq 0$. The proof is complete.

(2) $\Rightarrow$ (1). If $X \succ \Phi_q(K, X)$, the proof follows from Theorem 1 in Sinopoli et al. (2004) with $Q \succ 0$.

(2) $\Leftrightarrow$ (3).

$$\begin{aligned} &X \succ \Phi_q(K, X) \\ \Leftrightarrow\quad &X \succ (1-q)AXA' + q(A+KC)X(A+KC)' + qKRK' + Q. \end{aligned}$$

Let $G = X^{-1} \succ 0$ and $H = X^{-1}K$. Left and right multiply the above inequality by G we have

$$\begin{aligned} \Leftrightarrow\quad &GXG \succ (1-q)GAXA'G + q(GA + HC)X(GA + HC)' \\ &\quad + qHRH' + GQG. \end{aligned}$$

By using Schur complement this is equivalent to (17). $\square$

Theorem 13. *If $(A, Q^{1/2})$ is controllable and (A, C) is detectable, the solution of the optimization problem (16) can be obtained by solving the following quasi-convex optimization problem in the variables (q, G, H, Y).*

$$\min_{q, H, G \succ 0, Y \succ 0} \quad q$$

$$\begin{aligned} \text{subject to} \quad &Tr(Y) \le \gamma, \quad i = 1, 2, \ldots, N \\ &\begin{bmatrix} Y & I \\ I & G \end{bmatrix} \succeq 0 \\ &\Gamma_q(G, H) \succ 0, \quad q^c < q \le 1, \end{aligned} \quad (19)$$

where $\Gamma_q(G, H)$ is given by (17).

Proof. From the substitution of $G = X^{-1}$ in Lemma 12, it is straightforward to obtain the following equivalent statements by the Schur complement.

(a) $\exists X \in S_{++}$ such that $Tr(X) \le \gamma$.
(b) $\exists Y \in S_{++}$ such that $Y - X \succeq 0$ and $Tr(Y) \le \gamma$
(c) $\exists Y, G \in S_{++}$ such that $Tr(Y) \le \gamma$ and $\begin{bmatrix} Y & I \\ I & G \end{bmatrix} \succeq 0$.

From Lemma 12, we have the equivalence between $X = g_q(X)$ and $\Gamma_q(G, H) \succ 0$ in terms of feasibility. For fixed q, $\Gamma_q(G, H) \succ 0$ is a LMI in variables (G, H). Therefore, the problem (19) can be solved as a quasi-convex optimization problem by using bisection for variable q. $\square$

In summary, we have shown the following

Theorem 14. *Problem OP is equivalent to a quasi-convex optimization problem that can be solved by solving (15) using bisection on γ until $\mu(\gamma) = 1$ within the desired accuracy. For each level γ, each of the N independent subproblems (16) can be solved by solving (19) also using bisection for variable q_i and iterating LMI feasibility problems.*

3.1. Distributed solutions

Note that the steps of the outer bisection iteration are straightforward and can be done either by a centralized scheduler or in a distributed fashion. If a centralized scheduler/computing-unit is available, it can collect the $q_i^{opt}(\gamma)$'s from the estimators, check that their sums is less than or equal to one, and send back to the estimator an updated value of γ based on a bisection algorithm.

Alternatively, the estimators need to cooperate and agree on an optimal feasible γ. This can be done assuming the estimators are strongly connected via a network where the communications between any two estimators are error-free.[4] In this case, each estimator needs to obtain the value of $\mu(\gamma) = \sum_{i=1}^{N} q_i^{opt}(\gamma)$ by communicating with its neighbors. Under the assumption that N is known to the estimators, $\frac{1}{N}\sum_{i=1}^{N} q_i^{opt}(\gamma)$ can be obtained by a distributed averaging process in finite steps as shown in Sundaram and Hadjicostis (2008). Then, by increasing or decreasing γ under a common bisection rule among estimators, the value of $\sum_{i=1}^{N} q_i^{opt}(\gamma)$ can be driven to 1 and consequently OP is solved.

The above argument, leads to the following distributed computing algorithm, Algorithm 1, to solve OP. The inputs of the algorithm are global information assumed to be known by each estimator in prior. Denote γ^{opt} as the optimal objective value of OP. To avoid cumbersome details and to save space, we present the algorithm under the assumption that the interval $[l, u]$ is selected to contain γ^{opt}. i.e., we have $l \le \gamma^{opt} \le u$ at each step. Then the algorithm is guaranteed to converge to the optimal objective value γ^{opt} within the desired tolerance.

4. Extensions and special cases

In practical scenarios, some conditions/constraints might be of interest in the sensor scheduling problem. However, adding extra constraints on scheduling design is problematic in many existing scheduling strategies. Our stochastic scheduling approach can easily incorporate extra conditions/constraints, as shown through two specific examples next.

4.1. Prioritization of certain targets

For some reason, specific targets may require extra attention, (i.e., more precise estimation) from the sensor. Our model can

---

[4] Note that the decentralized computing units are allowed to be allocated in a single fusion center or to be physically distributed in an area.

Algorithm 1 Distributed algorithm for solving OP

Input: N, $l \leq \gamma^{opt}$, $u \geq \gamma^{opt}$, tolerance $\epsilon \geq 0$.
Output: $\{q_i^{opt}\}_{i=1}^N$, J^*
1: **for all** $i \in \{1, 2, \cdots, N\}$ **do**
2:    $u_i \leftarrow u$ and $l_i \leftarrow l$.
3: **end for**
4: **for all** $i \in \{1, 2, \cdots, N\}$ **do**
5:    **while** $u_i - l_i > \epsilon$ {Operations in this loop are synchronized among estimators.} **do**
6:      $\gamma \leftarrow \frac{l_i + u_i}{2}$.
7:      Obtain $q_i^{opt}(\gamma)$ by solving problem (19).
8:      Obtain $\mu(\gamma) = \sum_i q_i^{opt}$ via the distributed averaging algorithm.
9:      **if** $\mu(\gamma) \leq 1$ **then**
10:        $u_i \leftarrow \gamma$
11:      **else**
12:        $l_i \leftarrow \gamma$
13:      **end if**
14:    **end while**
15: **end for**
16: Objective value of OP $J^* \leftarrow \gamma$

---

incorporate this requirement by adding constraint $q_j \geq \alpha_j$ where α_j represents the assigned attention weight for target j. Then we need to solve the following problem,

$$\min_{q_i, i=[1..N]} \quad \max_i Tr(X_i)$$

$$\text{subject to} \quad X_i = g_{q_i}(X_i), \quad \sum_{i=1}^N q_i = 1, \quad q_j \geq \alpha_j$$
$$q_i^c < q_i \leq 1 \quad i = 1, 2, \ldots, N. \tag{20}$$

4.2. Measurement loss in sensing

In practice, measurement loss is a common phenomena due to various sources, e.g., shadowing, weather condition, large delay, etc. If the measurement loss probability τ_i for sensing ith target is known in prior, this extra condition can be easily incorporated in our model. Assume that q_i's are pre-assigned to each target. Then the actual probability of reliably receiving measurements from ith target is $q_i(1 - \tau_i)$ because of measurement loss. Therefore, OP can be modified as follows,

$$\min_{q_i, i=[1..N]} \quad \max_i Tr(X_i)$$

$$\text{subject to} \quad X_i = g_{q_i(1-\tau_i)}(X_i)$$
$$\sum_{i=1}^N q_i = 1 \tag{21}$$
$$q_i(1 - \tau_i) > q_i^c, \quad q_i \leq 1, \ i = 1, 2, \ldots, N.$$

With simple modifications, these two extended problems can be solved by the proposed distributed algorithms as well.

4.3. Closed-form solutions to the MARE for special cases

For the following special class of systems, the underlying MARE has a closed-form solution. Although this cannot be expected in general, it increases the computational efficiency of the proposed algorithm. Incidentally, this appears to the first non-trivial closed-form solution of the MARE in the literature.

Consider a set of N DTLTI single-state systems to be measured evolving according to the equation

$$x_i[k + 1] = a_i x_i[k] + w_i[k] \tag{22}$$

where $x_i[k]$, $v_i[k]$, $w_i[k] \in \mathbb{R}$ and the covariance of w and v are $Q_i \in \mathbb{R}^+$ and $R_i \in \mathbb{R}^+$, respectively. The measurement taken by the sensor at each time instant is formulated as follows,

$$\tilde{y}_i[k] = \xi_i[k](x_i[k - d_i] + v_i[k])$$

where d_i represents the delay in measurement, which we assume to be fixed and known in this paper. By using augmented states to deal with delays, it is straightforward to have the following compact form for system i with measurement delays,

$$X_i[k + 1] = A_i X_i[k] + B_i w_i[k]$$
$$\tilde{y}_i[k] = \xi_i[k](C_i X_i[k] + v_i[k]), \tag{23}$$

where X, A, B, C has the following structure

$$X_i[k] = \begin{bmatrix} x_i^1[k] \\ x_i^2[k] \\ \vdots \\ x_i^{d_i}[k] \\ x_i[k] \end{bmatrix}, \qquad A_i = \begin{bmatrix} 0 & 1 & 0 & \cdots & 0 \\ 0 & 0 & 1 & \cdots & 0 \\ \vdots & & & \ddots & \vdots \\ 0 & 0 & 0 & \cdots & 1 \\ 0 & 0 & 0 & \cdots & a_i \end{bmatrix},$$

$$B_i = \begin{bmatrix} 0 \\ 0 \\ \vdots \\ 0 \\ 1 \end{bmatrix} \qquad C_i = \begin{bmatrix} 1 & 0 & \cdots & 0 \end{bmatrix}. \tag{24}$$

Note that only $x_i[k]$ is the true state of system i at time instant k while other states included in vector $X_i[k]$ are dummy variables for handling delays. By exploiting the special structure of above model, we are able to obtain the closed-form fixed point of the MARE. We present this result in the following theorem.

Theorem 15. *For a given $0 \leq q \leq 1$, consider the MARE described as $(A, C, R, \tilde{Q})$, where (A, C) have the structure presented in* (24), *$R \in \mathbb{R}^+$ and $\tilde{Q} = BQB'$ with $Q \in \mathbb{R}^+$. Then the MARE has a unique positive-semidefinite fixed point X as follows, if $a = 1$ and $q \neq 0$,*

$$X = \begin{bmatrix} x_1 & x_1 & \cdots & x_1 \\ x_1 & x_2 & \cdots & x_2 \\ \vdots & \vdots & \ddots & \vdots \\ x_1 & x_2 & \cdots & x_n \end{bmatrix} \tag{25}$$

where

$$x_j = \frac{Q + \sqrt{Q^2 + 4qQR}}{2q} + (j - 1)Q \quad j = 1, 2, \ldots, n;$$

if $0 < a < \sqrt{\frac{1}{1-q}}$ and $a \neq 1$,

$$X = \begin{bmatrix} x_1 & ax_1 & \cdots & a^{n-1}x_1 \\ ax_1 & x_2 & \cdots & a^{n-2}x_2 \\ \vdots & \vdots & \ddots & \vdots \\ a^{n-1}x_1 & a^{n-2}x_2 & \cdots & x_n \end{bmatrix} \tag{26}$$

where

$$x_1 = \frac{Ra^2 - R + Q + \sqrt{(Ra^2 - R + Q)^2 - 4(a^2 - 1 - a^2 q)QR}}{2(1 + a^2 q - a^2)}$$

$$x_j = a^{2(j-1)}x_1 + \frac{1 - a^{2(j-1)}}{1 - a^2}Q \quad j = 1, 2, \ldots, n;$$

if $a \geq \sqrt{\frac{1}{1-q}}$, MARE fails to converge to a steady state value.

The proof is tedious but straightforward by plugging in the above closed-form solution into the MARE. If $a \geq \sqrt{\frac{1}{1-q}}$, the results directly follows from Mo and Sinopoli (2008).

Note that the stochastic upper bound in Theorem 1 may not hold since A is not invertible. However, a deterministic scheduling sequence can be constructed based on the optimal stochastic solution in the next section, providing an upper bound on the original deterministic scheduling problem with good performance.

5. Scheduling implementation

For completeness, in this section we consider the problem of scheduling implementation and present a simple approach to implement the scheduling sequence. For stochastic scheduling implementation, a central scheduler is required to construct a scheduling sequence by randomly selecting targets (via a random seed) according to the optimal probability distribution. Note that this construction process can be performed efficiently either off-line or on-line.

We next turn our attention back to a deterministic scheduling and look for one consistent with the optimal stochastic solution. Note that the optimization problems are minimizing the average costs of all possible stochastic sequences. With the optimal stochastic solutions, we are able to randomly construct a sequence compatible with the distribution. However, in practice, such random-constructed scheduling sequence may result in undesirable performance. For example, one target may not be measured for a long consecutive time instants and its error covariance is temporarily built up. Thus, we would like to identify and use, among all possible stochastic sequences, those that have low costs. Motivated by the sensor scheduling literature, which suggests periodic solutions (Hovareshti, Gupta, & Baras, 2007; Zhang, Vitus, Hu, Abate, & Tomlin, 2010), we define and look for deterministic sequences of minimal consecutiveness, defined next. These sequences are periodic and switch among targets most often compatibly with the optimal scheduling distribution. We remark that the approach we propose in this section is heuristic but it can be implemented in a distributed fashion and leads to good performance in simulations. We leave the analysis of this and other approaches to future research.

Definition 16. Let $\{s[k]\}_{k=1}^{L}$ be a set of sequences with length L, where each element $s[k]$ in the sequence takes value from an element set $\mathbb{K} = \{a_1, a_2, \ldots, a_N\}$ and the number of occurrences of each value a_i in the sequence is n_i. Under these assumptions, then the sequence of minimal consecutiveness is the solution of the following optimization problem

$$\min_{\{s[k]\}_{k=1}^{L}} \max_{i,j\in\{1,2,\ldots,N\}} \{j - i | j \geq i, s[i] = s[i+1] = \cdots = s[j]\}.$$

Note that the minimal consecutiveness sequence may not be unique. The intuition for concentrating on sequences of minimal consecutiveness is that, under this class of sequences, each target is visited in the shortest time instants compatible with the optimal probability distribution. We next provide a heuristic algorithm with objective to construct a periodic minimal consecutive sequence.

Remark 17. First of all, the complexity of running Algorithm 2 is $\mathcal{O}(L)$. Next, in our stochastic scheduling strategy, the integer value n_i is obtained from the optimal probability distribution, i.e., $n_i = \lfloor q_i L \rfloor$. In order to generate a scheduling sequence precisely matching the distribution $q_i's$ within certain precision, L should be chosen such that $q_i L$ ($i = 1, 2, \ldots, N$) is an integer. The rare cases where L is supposed to be infinity, are approximated by

---

Algorithm 2 Construction of a scheduling sequence of minimal consecutiveness

Input: L: the sequence length; n_i: the number of occurrences of a_i ($i \in \{1, 2, \cdots, N\}$) in a L-length sequence. Without loss of generality, we assume that $n_1 \geq n_2 \geq \cdots \geq n_N$.

Output: $\{s[k]\}_{k=1}^{L}$

1: $\{s[k]\}_{k=1}^{n_1} \leftarrow$ Generate a n_1-length sequence with all entries as a_1.

2: **for all** $i \in \{2, \cdots, N\}$ **do**

3:     Interpolate an element a_i for every m_i's a_1 with equal interpolation interval where $m_i = \lceil \frac{n_1}{n_i+1} \rceil$. This can be done by using a backoff counter for each a_i. From the beginning of the sequence, the counter is reduced by one whenever an a_1 is found and reset to be $\lceil \frac{n_1}{n_i+1} \rceil$ as long as a_i is placed.

4: **end for**

---

taking L large, and a viable sequence is then obtained by periodic continuation. Therefore, besides practical requirements on the length of measuring period, we should choose L by taking into account both the computing capability of the centralized scheduler and the probability matching precision.

Remark 18. Several other relevant deterministic scheduling strategies can be found in the literature, e.g. Round Robin, Pinwheel Scheduling (Chan & Chin, 1992; Chan, 1993). However, a careful review finds that they are not suitable/extendable to our specific settings.

### 5.1. A distributed scheduling implementation

The above algorithm lends itself to a distributed implementation. We assume that the estimators are strongly connected through a network where the communications between any two estimators are error-free. Under the assumption that each estimator is capable of sensing the channel state, i.e., idle or occupied, we propose a heuristic distributed scheduling mechanism to approximate the result of Algorithm 2. This distributed scheduling scheme is derived based on the well-known mechanism − Carrier Sense Multiple Access with Collision Avoidance (CSMA/CA). As in CSMA/CA, our proposed multiple access mechanism relies on backoff timers. Specifically, each estimator generates a backoff timer as $T_i = \frac{\alpha}{q_i}$ where α is chosen such that $T_i \ll \tau$ (τ refers to the sampling period of DTLTI systems). Then each estimator regularly senses the transmission channel during its backoff timer. If the channel is sensed to be "idle" and the timer of estimator i goes off, estimator i begins to use the channel for observing the ith DTLTI system. If the channel is sensed to be busy, then the backoff timer must be frozen until the channel becomes free again. Remark that no synchronization is necessary among backoff timers. With small probability, collision may happen among estimators. That is, backoff timers of two or more estimators go off at the same moment. To deal with this problem, the backoff timer of ith estimator (subject to collision) can be adjusted as $T_i = \frac{\alpha}{q_i - \epsilon_i}$ where $\epsilon_i > 0$ is randomly chosen and $\epsilon_i \ll q_i$. After estimator i uses the channel once, $T_i = \frac{\alpha}{q_i - \epsilon_i}$ is set to be $T_i = \frac{\alpha}{q_i}$ immediately. We remark that, for a large period T, the number of observations on ith DTLTI system is approximately $q_i \frac{T}{\alpha}$ where $\frac{T}{\alpha}$ can be treated as a normalizing factor. Thus the optimal probability distribution $q_i's$ is preserved.

6. Examples and simulations

In this section, we present some simulation results to verify our stochastic scheduling strategy and algorithm.

Table 1
Results of the example.

	Opt. q_i's	Opt. Cost	Emp. Cost	MC Cost
OP	0.674, 0.326	59.1	58.7	55.7
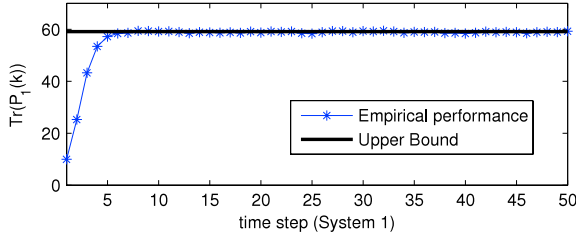
Fig. 1. Empirical estimate error covariances for DTLTI system (27). The performance for system 2 has the same feature. The empirical performance curve is obtained by averaging 5000 Monte Carlo simulations.

6.1. Example A

Consider a single sensor for measuring two DTLTI systems with the following state space representations,

$$A_1 = \begin{bmatrix} 0 & 1 \\ -0.49 & 1.4 \end{bmatrix}, \qquad C_1 = \begin{bmatrix} 1 \\ 0 \end{bmatrix}',$$
$$Q_1 = \begin{bmatrix} 5 & 0 \\ 0 & 5 \end{bmatrix}, \qquad R_1 = 0.5 \tag{27}$$

$$A_2 = \begin{bmatrix} 0 & 1 \\ -0.72 & 1.7 \end{bmatrix}, \qquad C_2 = \begin{bmatrix} 1 \\ 0 \end{bmatrix}',$$
$$Q_2 = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}, \qquad R_2 = 1. \tag{28}$$

The results of applying Algorithm 1 are reported in Table 1. In the table, we also report the empirical cost obtained by pre-assigning the optimal probability distribution to the sensor and computing the empirical error state covariance of each system by accordingly generating random switching sequences. This simulation result verifies that OP provides an effective upper bound on the expected estimate error covariances of our stochastic scheduling strategies. This is further exemplified in Fig. 1, which shows the empirical expected estimate error covariance of system (27).

6.1.1. Minimal consecutiveness (MC) vs. stochastic scheduling (SS)

As shown in Fig. 2, the blue solid curves show the evolution of the error covariance w.r.t time k. We see several high peaks as a result of consecutive loss of observations. The dash lines show that the MC sequence provides much smoother error covariance which is quite necessary in certain applications. Furthermore, the objective value provided by MC sequence, 55.7, is smaller than the empirical expected estimate error covariances, 58.7, provided by the SS sequences. This performance enhancement is the result of "smoothness" provided by the MC sequence. In a word, the MC sequence provides a better upper bound than that provided by the SS sequences (i.e. the optimal objective value μ_s in (7)).

Finally, for completeness, we compare the MC cost with the cost of the (tree search based) sliding window algorithm (see Gupta, Chung, Hassibi, & Murray, 2004). It is worth noting that the optimal cost of the original deterministic scheduling problem can be approximated by sliding window algorithm with sufficiently large window size, at the expense of huge computing cost. In this example, the MC cost is $\simeq 55.7$ (in Table 1), while the sliding window algorithm has the best cost of $\simeq 57.9$ with window size up to 15. In this case, we see that MC not only has better performance than the sliding window performance, but also can be determined off-line, has low computing complexity and is simple to implement, and scales much better with the number of systems to schedule.
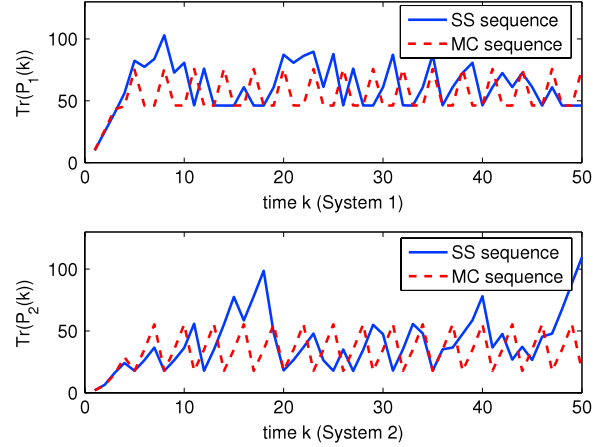
Fig. 2. Performance of minimal consecutiveness sequence vs. stochastic scheduling sequence.

6.2. Example B

In this example, we consider three random-walk vehicles in an area and a single sensor equipped with a camera is used for tracking their $1-D$ positions. The dynamics of their positions are evolving as (22) with $a_i = 1$. But they are subject to different process noises, measurement noises and delays. Here we assume that these vehicles have 1, 2 and 2 time-step measurement delays, respectively. Then we have expanded state space systems

$$A_1 = \begin{bmatrix} 0 & 1 \\ 0 & 1 \end{bmatrix}, \qquad B_1 = \begin{bmatrix} 0 \\ 1 \end{bmatrix}, \qquad C_1 = \begin{bmatrix} 1 \\ 0 \end{bmatrix}'$$
$$A_i = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ 0 & 0 & 1 \end{bmatrix}, \qquad B_i = \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} \qquad C_i = \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix}' \quad i = 2, 3$$
$$Q_1 = 1 \quad Q_2 = 2 \quad Q_3 = 5, \qquad R_1 = R_2 = R_3 = 1.$$

By running the proposed distributed algorithms, the solutions of OP is $[q_1, q_2, q_3] = [0.0649, 0.1612, 0.7739]$. We assume the existence of a centralized scheduler which constructs a random scheduling sequence accordingly. The tracking performances are shown in Fig. 3. The tracking paths (red curves) are shown in comparison with actual time-varying positions (black curves) of random-walk vehicles. The flat segments of red curves imply that no measurement is taken in this time slot and the estimator simply propagates the state estimate of the previous time-step. In Fig. 3, it is shown that the actual path of vehicle 1 changes slowly, correspondingly the attention given to this sensor is small and the estimate path is updated rarely. In comparison, the tracking paths of vehicle 2 and vehicle 3 match the actual positions much better even though the flat segments in tracking path of vehicle 2 is occasionally visible.

7. Conclusion

In this paper, we have proposed a stochastic sensor scheduling problem where the states of N targets (i.e., DTLTI systems) need to be estimated. Due to some practical sensing constraints, at each time instant, only one target can be measured by the sensor. The basic idea of stochastic scheduling is that at each time instant a target is randomly chosen for observation according to a pre-assigned probability distribution. We have relaxed the problem to a convex optimization problem and proposed a distributed algorithm to obtain the optimal probability distribution with the objective to minimize the maximal estimate error among all targets. Then we proposed centralized and distributed scheduling implementation schemes. Finally, we presented simulation results to verify our stochastic strategy.
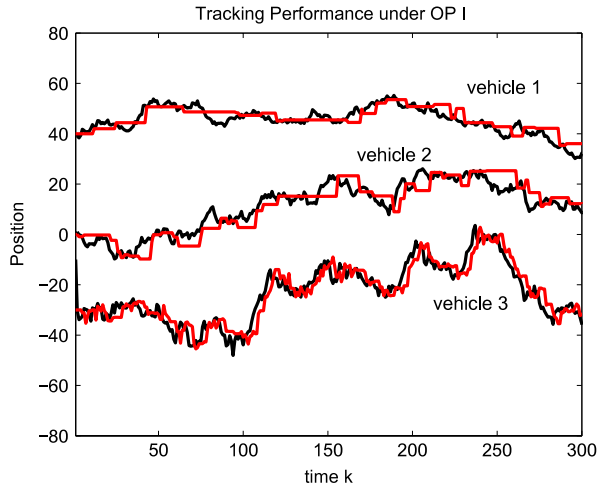
Fig. 3. Positions of random-walk vehicles and tracking paths. The flat segments imply that no measurement is taken in the time slot and the estimator simply propagates the state estimate of the previous time-step. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

Appendix. Proof of Theorem 1

Proof. Firstly, we define an upper bound $\bar{\mu}_d$ on the performance μ_d of the deterministic scheduling problem (5). For a given $q_1, q_2, \ldots, q_N$ ($\sum_{i=1}^N q_i = 1$), we generate switching sequences with the distribution (6). Clearly, these sequences are not necessarily optimal. Let

$$\bar{\mu}_d(q_1, q_2, \ldots, q_N) = \min_{q_i, i=[1..N]} \max_i \left(\limsup_{T \to \infty} \frac{1}{T} \sum_{k=1}^T Tr(P_i[k]) \right) \quad (29)$$

with constraints (4) and (6). Note that given the optimal distribution of $q_1, q_2, \ldots, q_N$ this upper bound still holds since the scheduling sequence is chosen randomly according to (6). Furthermore, under the assumption, the limit value in (29) exists almost surely. This is because for a stochastic process $\{P_i[k]\}$ ($P_i[0] = \Sigma$) satisfying (4), there exists an ergodic stationary process $\{\bar{P}_i[k]\}$ satisfying (4) with

$$\lim_{k \to \infty} \|P_i[k] - \bar{P}_i[k]\| = 0. \quad a.s. \quad (30)$$

See Bougerol (1992) Theorem 3.4. That is,

$$\lim_{T \to \infty} \frac{1}{T} \sum_{k=1}^T Tr(P_i[k]) = \lim_{T \to \infty} \frac{1}{T} \sum_{k=1}^T Tr(\bar{P}_i[k]). \quad a.s. \quad (31)$$

Thus, (29) can be rewritten as

$$\bar{\mu}_d(q_1, q_2, \ldots, q_N) = \min_{q_i, i=[1..N]} \max_i \left(\lim_{T \to \infty} \frac{1}{T} \sum_{k=1}^T Tr(P_i[k]) \right). \quad (32)$$

Next, for every sensor, its selection process can be viewed as a binary tree where the variance of observation noise on one branch is infinity if the sensor is not selected at this time instant. This model is a special case of the one (tree topology) considered in Mo et al. (2011). Under the current assumptions it is straightforward to check that all assumptions in Theorem 1 of Mo et al. (2011) hold. As a result, inequality (10) in Mo et al. (2011) holds in our case, i.e.,

$$\lim_{T \to \infty} \frac{1}{T} \sum_{k=1}^T Tr(P_i[k]) \le Tr(\lim_{k \to \infty} \mathbb{E}_{\xi_{i,k}}[P_i[k]]). \quad a.s.$$

Recalling that $\mu_s = \min_{q_i, i=[1..N]} \max_i Tr(\lim_{k \to \infty} \mathbb{E}_{\xi_{i,k}}[P_i[k]])$, it follows that $\mu_d \le \bar{\mu}_d \le \mu_s$. Namely, the performance μ_d of the deterministic scheduling problem (5) is upper bounded by the performance μ_s of the stochastic scheduling problem (7) □

References

Acar, E. U., & Choset, H. (2002). Sensor-based coverage of unknown environments: incremental construction of morse decompositions. *International Journal of Robotics Research*, 21(4), 345–366.

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

Bitar, E., Baeyens, E., & Poolla, K. (2009). Energy-aware sensor selection and scheduling in wireless sensor networks. In *1st IFAC workshop on estimation and constrol of networked systems*.

Bougerol, P. (1992). Some results on filtering riccati equation with random parameters. In *Applied Stochastic Analysis, ser. Lecture Notes in Control and Information Sciences*. Berlin, Germany: Springer.

Chan, M. Y., & Chin, F. Y. L. (1992). General schedulers for the pinwheel problem based on double-integer reduction. *IEEE Transactions on Computing*, 41(6), 755–768.

Chan, M. Y., & Chin, F. Y. L. (1993). Schedulers for larger classes of pinwheel instances. *Algorithmica*, 9(5), 425–462.

Choset, H. (2001). Coverage for robotics a survey of recent results. *Annals of Mathematics and Artificial Intelligence*, 31(1–4), 113–126.

Cortes, J. (2010). Coverage optimization and spatial load balancing by robotic sensor networks. *IEEE Transactions on Automatic Control*, 55, 749–754.

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

Cremean, L., Dunbar, W., van Gogh, D., Hickey, J., Klavins, E., Meltzer, J., & Murray, R.M. (2002). The caltech multi-vehicle wireless testbed. In *Proceedings of 41th IEEE Conference on Decision and Control*.

Gupta, V., Chung, T., Hassibi, B., & Murray, R. M. (2004). Sensor scheduling algorithms requiring limited computation. *IEEE International Conference on Acoustics, Speech, and Signal Processing*, 3, 825–828.

Gupta, V., Chung, T. H., Hassibi, B., & Murray, R. M. (2006). On a stochastic sensor selection algorithm with applications in sensor scheduling and sensor coverage. *Automatica*, 42, 251–260.

Hovareshti, P., Gupta, V., & Baras, J.S. (2007). Sensor scheduling using smart sensors, In *Proceedings of 46th IEEE Conference Decision and Control* (pp. 494–499).

Hussein, I. I., & Stipanovi, D. M. (2007). Effective coverage control for mobile sensor networks with guaranteed collision avoidance. *IEEE Transactions on Control System Technology*, 15(4), 642–657.

Joshi, S., & Boyd, S. (2009). Sensor selection via convex optimization. *IEEE Transactions on Signal Processing*, 57, 451–462.

Li, C., & Elia, N. (2011). A sub-optimal sensor scheduling strategy using convex optimization. In *American control conference* (pp. 3603–3608).

Liang, S., Tang, Y., & Zhu, Q. (2007). Passive wake-up scheme for wireless sensor networks. In *2nd international conference on innovative computing, information and control* (p. 507).

Lin, Z., & Wang, C. (2013). Scheduling parallel kalman filters for multiple processes. *Automatica*, 49(1), 9–16.

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

Mo, Y., & Sinopoli, B. (2008). A characterization of the critical value for kalman filtering with intermittent observations. In *Proceedings of 47th IEEE Conference Decision and Control* (pp. 2692–2697).

Ny, J., Feron, E., & Dahleh, M. A. (2011). Scheduling continuous-time kalman filters. *IEEE Transactions on Automatic Control*, 56(6), 1381–1394.

Shi, L., Capponi, A., Johansson, K. H., & Murray, R. M. (2008). Network lifetime maximization via sensor trees construction and scheduling. In *3rd international workshop on feedback control implementation and design in computing systems and networks*.

Shi, L., Johansson, K. H., & Murray, R. M. (2007). Change sensor topology when needed: How to efficiently use system resources in control and estimationover wireless networks. In *Proceedings of 46th IEEE Conference on Decision and Control* (pp. 5478–5485).

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

Srivastava, V., Plarre, K., & Bullo, F. (2011). Randomized sensor selection in sequential hypothesis testing. *IEEE Transactions on Signal Processing*, 59(5), 2342–2354.

Sundaram, S., & Hadjicostis, C. N. (2008). Distributed function calculation and consensus using linear iterative strategies. *IEEE Journal on Selected Areas in Communications*, 26(4), 650–660.

Tiwari, A., Jun, M., Jeffcoat, D., & Murray, R. M. (2005). Analysis of dynamic sensor coverage problem using kalman filters for estimation. In *Proceedings of 16th IFAC world congress*.

Vasanthi, N. A., & Annadurai, S. (2006). Sleep schedule for fast and efficient control of parameters in wireless sensor−actor networks. In *1st international conference on communication system sofrware and middelware* (pp. 1–6).

Zhang, W., Vitus, M. P., Hu, J., Abate, A., & Tomlin, C. J. (2010). On the optimal solutions of the infinite-horizon linear sensor scheduling problem. In *Proceedings of 49th IEEE Conference Decision and Control* (pp. 396–401).
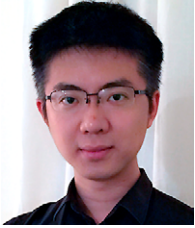
Chong Li received the B.S. degree from Harbin Institute of Technology, Harbin, China in 2008 and the Ph.D. degree from Iowa State University, Ames, IA in 2013, both in Electrical Engineering. He is currently a research engineer at Qualcomm Research and Development (R&D), NJ. His research interests include networked control systems, information and coding theory and the interaction of communication, control and computation.

Nicola Elia received the Laurea degree in electrical engineering from the Politecnico of Turin, Turin, Italy, in 1987, and the Ph.D. degree in electrical engineering and computer science from the Massachusetts Institute of Technology (MIT), Cambridge, in 1996. He was a Control Engineer with Fiat Research Center from 1987 to 1990. He was a Postdoctoral Associate at the Laboratory for Information and Decision Systems, MIT from 1996 to 1999. Presently, he is Professor of Electrical and Computer Engineering at Iowa State University, Ames. Dr. Elia received the NSF CAREER Award in 2001. His research interests include computational methods for controller design, communication systems with access to feedback, control with communication constraints, and networked systems.