Optimal Event Triggered Transmission of Information in Distributed State Estimation Problems

Lichun Li, Michael Lemmon and Xiaofeng Wang

Abstract— **This paper considers a distributed estimation problem in which a sensor sporadically transmits information to a remote observer. An event-triggered approach is used to trigger the transmission of information from the sensor to the remote observer. The event-trigger is chosen to minimize the mean square estimation error at the remote observer subject to a constraint on how frequently the information can be transmitted. This problem was recently studied by M. Rabi et al. [1] where the observed process was a scalar linear system over a finite time interval. This paper extends those earlier results by relaxing the prior assumption that the initial condition is zero-mean and that there is no measurement noise. This is done by adopting a dynamic programming approach to solve the problem.**

I. INTRODUCTION

Wireless networking has become an attractive method for implementing networked control systems. The emergence of standardized, low-cost, low power radios had made industrial applications of wireless networks economically attractive [2].

A major challenge encountered in using such wireless networks is that these networks have limited throughput capacity. In addition to this, a wireless link's capacity will vary over time due to changes in the external environment. These time-varying limitations in link bandwidth may have a negative impact on overall system behavior. This is particularly true in networked control systems, where the quality of the feedback data has a direct impact on the physical plant's stability and performance [3], [4], [5]. This can also be seen in embedded sensor network applications where multiple sensors transmit data over an ad hoc wireless network to a data fusion center [6]. In both cases, network artifacts (such as dropouts and delays) may adversely affect the application's performance.

Many networked control systems presume the periodic transmission of information across the network. Periodic transmission, however, may consume more network bandwidth than necessary. Since the period is chosen prior to system deployment, it must be robust over all variations in network and system behavior and this "open-loop" approach to period selection can be overly conservative in its use of network bandwidth.

Lichun Li and Michael Lemmon are with the department of Electrical Engineering, University of Notre Dame, Notre Dame, IN 46556, USA. (e-mail:lli3,lemmon@nd.edu). The authors gratefully acknowledge the partial financial support of the National Science Foundation NSF-CNS-0720457 and NSF-CNS-0931195

Xiaofeng Wang is with the department of Mechanical Science and Engineering, University of Illinois at Urbana-Champaign, Urbana, IL, 61801. email: wangx@illinois.edu

The recognition of the inherent conservatism in open-loop periodic transmission policies has led numerous researchers to move towards the *sporadic* transmission of information through *event-triggered* formalisms. Event-triggering has an agent transmit information to its neighbors when some measure of the "novelty" in that information exceeds a specified threshold. Early examples of event-triggering were used in relay control systems[7] and more recent work has looked at event-triggered PID controllers [8]. Much of the early work in event-triggered control assumed event-triggers in which the triggering thresholds were constant. Recently it was shown that state-dependent event triggers could be used to enforce stability concepts such as input-to-state stability [9] or $\mathcal{L}_2$ stability [10] in both embedded control systems and networked control systems [11]. There has been ample experimental evidence [12], [13], [14] to suggest that event-triggering can greatly reduce communication bandwidth while preserving overall system performance. Event-triggering therefore provides a useful approach for reducing an application's use of the communication network.

While much of the prior work in event-triggering has focused on control in embedded and networked control systems. This prior work, however, has usually assumed full state feedback controllers. Extending this to output feedback control is complicated by the fact that one would need to estimate the process state. This paper therefore focuses on the state estimation problem in networked systems. In particular, we consider a simple "canonical" problem that was recently studied by M. Rabi [1]. This problem considers a discrete-time scalar linear process over a finite interval of time. The process is observed by a sensor that constructs local estimates of the process state and must decide when to transmit those local estimates to a *remote observer* so that the mean square estimation error at the remote observer is minimized. To keep the problem interesting, the decision to transmit local estimates must satisfy a bandwidth constraint that limits the number of messages that the sensor can send to the remote observer. This paper extends the earlier work in [1] by dropping the earlier assumption of zero mean initial conditions with no measurement noise. The technique used to obtain these results also differs from [1]. Rather than using optimal stopping results, this paper uses stochastic dynamic programming to derive the value function and event-triggering thresholds. This approach is very similar to what was used in [15] for a related infinite horizon problem. This analysis recovers the original results in [1].

The remainder of this paper is organized as follows. Section II discusses the prior work on event-triggered state estimation. The problem statement, main results and simulation results are in section III, IV, V, respectively. Future directions are in section VI. Most of the paper's proofs will be found in section VII.

II. Prior work

It has long been recognized that the sporadic flow of information can be incorporated into Kalman filters[16], [17] and into multi-sensor networks [18]. Rather than simply analyzing the impact that nondeterministic network artifacts have on estimator performance, one can also think about "controlling" the way in which information is transmitted. In multi-sensor networks, for example, one may try to schedule sensor transmissions in a manner that achieves optimal performance while reducing overall network usage [19], [20]. If one focuses on a single remote sensor transmitting over a throughput constrained link, then one can also control the time when information was transmitted. The potential benefits of controlling transmission time were experimentally documented by Yook et al. [21]. Formal analyses of this tradeoff were later carried out by Xu and Hespanha [15] for infinite horizon estimation problems and later by M. Rabi [1], [22] for finite horizon estimation problems. This paper uses event-triggering to control transmission times across a single communication link. This section reviews the related prior work [21], [15], [1] cited above.

In the system architecture considered by J. Yook et al.[21], each node can talk with other nodes. The local node estimates the state of the other nodes. At the same time, when a local node finds that the estimation error of the local state is greater than a pre-specified threshold, the true local state will be broadcasted to its neighbors. It was shown through simulations that network bandwidth can be significantly reduced while the performance of the system is only slightly impacted. This paper therefore provides a good motivation for controlling transmission times.

Based on the same system architecture, Y. Xu and J. Hespanha [15] derived an optimal triggering event to minimize the sum of the total mean square estimation error (MSEE) and the communication cost. Dynamic programming was used to solve this problem over an infinite horizon. Computing the optimal event-triggering threshold, however, proved to be difficult and approximate thresholds were determined in [23].

M. Rabi et al. [1], [22] examined a problem which sought to minimize the MSEE over a finite horizon subject to a constraint on the maximum number of transmissions. This work confined its attention to discrete-time scalar linear systems with zero initial condition and no measurement noise. For the single-sample case, the problem of minimizing the MSEE was treated as an optimal stopping problem [24]. This approach led to a backward recursion that computed the minimum total cost (MSEE) for the problem. This earlier work asserted that the extension to multiple transmissions was relatively easy. But they only provide partial characterizations of the proofs and algorithms supporting this assertion.

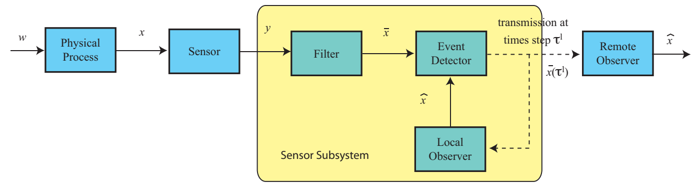

Fig. 1. Structure of event triggered networked state estimator

This paper adopts the dynamic programming methods to re-examine the finite-horizon multi-sample problem treated in [1]. Our use of dynamic programming is similar to what was used in [15] for a related infinite horizon problem. We recover the original results in [1] that determine an optimal time-varying event-triggering threshold. We also generalize the results in [1] to cases where the initial state is non-zero mean and the sensor data is corrupted by measurement noise. The backward recursion developed in this paper applies to both the scalar systems treated in [1] as well as more general vector systems. As was found in [23], however, the computation of the optimal event-triggering thresholds is computationally expensive, thereby suggesting that future work should investigate methods for approximating these optimal event-triggers.

III. Problem statement

The event-triggering problem considered in [1] assumes that a sensor is observing a scalar linear discrete-time process over a finite horizon of length M. The process state $x : [0, M] \to \Re$ satisfies the difference equation

$$x_{k+1} = ax_k + w_k$$

for $k \in [0, M]$ where a is a real constant, $w : [0, M] \to \Re$ is a zero mean white noise process with variance Q. The initial state, x_0, is assumed to be a Gaussian random variable with mean μ_0 and variance Π_0. The sensor generates a measurement $y : [0, M] \to \Re$ that is a corrupted version of the process state. The sensor measurement at time k is

$$y_k = x_k + v_k$$

for $k \in [0, M]$ and where $v : [0, M] \to \Re$ is another zero mean white noise process with variance R that is uncorrelated with the process noise w. The process and sensor blocks are shown on the left hand side of figure 1. In this figure, the output of the sensor feeds into a transmission subsystem that decides when to transmit information to a remote observer.

The subsystem consists of three components; an *event detector*, a *filter*, and a *local observer*. The event detector decides when to transmit information at $B \in [0, M+1]$ time instants to the remote observer. So B represents the total number of transmissions that the sensor is allowed to make to the remote observer. We let the $\{\tau^l\}_{l=1}^B$ denote a sequence of increasing times ($\tau^l \in [0, M]$) when information is transmitted from the sensor to the remote observer. The decision to transmit is based on estimates that are generated by the "filter" and "local observer".

Let $\mathcal{Y}_k = \{y_0, y_1, \cdots, y_k\}$ denote the measurement information available at time k. The *filter* generates a state estimate $\overline{x} : [0, M] \to \Re$ that minimizes the mean square estimation error $E\left[(x_k - \overline{x}_k)^2 \,|\, \mathcal{Y}_k\right]$ at each time step conditioned on all of the sensor information received up to and including time k. These estimates, of course, can be computed using a Kalman filter. For the scalar process under study the filter equations are,

$$\begin{aligned} \overline{x}_k &= E\left[x_k \,|\, \mathcal{Y}_k\right] = a\overline{x}_{k-1} + L_k(y_k - a\overline{x}_{k-1}) \\ \overline{P}_k &= E\left[(x_k - \overline{x}_k)^2 \,|\, \mathcal{Y}_k\right] \\ &= a^2\overline{P}_{k-1} + Q - L_k^2(a^2\overline{P}_{k-1} + Q + R), \end{aligned} \quad (1)$$

where $\overline{x}_0 = \frac{\Pi_0}{\Pi_0+R}y_0 + \frac{R}{\Pi_0+R}\mu_0$, $\overline{P}_0 = \frac{\Pi_0 R}{\Pi_0+R}$ and $L_k = \frac{a^2\overline{P}_{k-1}+Q}{a^2\overline{P}_{k-1}+Q+R}$.

The event detector uses the *filter's* state estimate, $\overline{x}$, and another estimate generated by a *local observer* to decide when to transmit the filtered state $\overline{x}$ to the *remote observer*. Given a set of transmission times $\{\tau^l\}_{l=1}^B$, let $\overline{\mathcal{X}}_k = \{\overline{x}_{\tau^1}, \overline{x}_{\tau^2}, \ldots, \overline{x}_{\tau^{\ell(k)}}\}$ denote the filter estimates that were transmitted to the remote observer by time k where $\ell(k) = \max\left\{\ell : \tau^\ell \leq k\right\}$. We can think of this as the "information set" available to the remote observer at time k. The remote observer generates an a posteriori estimate $\hat{x} : [0, M] \to \Re$ of the process state that minimizes the MSEE, $E\left[(x_k - \hat{x}_k)^2 \,|\, \overline{\mathcal{X}}_k\right]$, at time k conditioned on the information received up to and including time k. The a priori estimate of the remote observer, $\hat{x}^- : [0, M] \to \Re$, minimizes $E\left[(x_k - \hat{x}_k)^2 \,|\, \overline{\mathcal{X}}_{k-1}\right]$, the MSEE at time k conditioned on the information received up to and including time $k-1$. Due to the scalar nature of the process, these estimates take the form

$$\begin{aligned} \hat{x}_k^- &= E\left[x_k \,|\, \overline{\mathcal{X}}_{k-1}\right] = a\hat{x}_{k-1} \\ \hat{x}_k &= E\left[x_k \,|\, \overline{\mathcal{X}}_k\right] = \begin{cases} \hat{x}_k^- & \text{don't transmit at step } k \\ \overline{x}_k & \text{transmit at step } k \end{cases} \end{aligned}$$

where $\hat{x}_0^- = \mu_0$.

The event-detection strategy that is used to select the transmission times τ^l is based on observing the *gap*, $e_k^- = \overline{x}_k - \hat{x}_k^-$ between the filter's estimate $\overline{x}$ and the remote observer's a priori estimate $\hat{x}^-$. Note that even though the gap is a function of the remote observer's estimate, this signal will be available to the sensor. This is because the sensor has access to all of the information, $\overline{\mathcal{X}}_k$, that it sent to the remote observer. As a result, the sensor can use another *local estimator* to construct a copy of $\hat{x}$ that can be locally accessed by the event-detector to compute the gap. This local estimator is shown as part of the detection subsystem in figure 1. The event detector's decision to transmit is triggered when the estimate's gap e_k^- goes out of the time varying trigger set $S_k^{p_k}$ where $k \in [0, M]$ and p_k is the number of transmissions that are remaining at step k. As noted in [1], this type of decision logic essentially treats the transmission time as a random variable that forms a stopping time of the stochastic process being monitored. As Rabi showed, the "optimal" sets can be computed using results from optimal stopping theory [24]. The result, however, is a backward recursion that bears great similarity to dynamic programming recursions. So in this paper we adopt stochastic dynamic programming to obtain these trigger sets.

For later convenience, the following notational conventions are used throughout this paper,

$$\hat{e}_k = x_k - \hat{x}_k \qquad \text{estimation error at step } k \quad (2)$$

$$\overline{e}_k = x_k - \overline{x}_k \qquad \text{filtered state error at step } k \quad (3)$$

$$e_k^- = \overline{x}_k - \hat{x}_k^- \qquad \text{a priori gap at step } k \quad (4)$$

$$e_k = \overline{x}_k - \hat{x}_k \qquad \text{a posteriori gap at step } k \quad (5)$$

We let p_k denote the number of transmissions that are remaining at step k. This number must be an integer between 0 and B. The apriori information available to the event detector at time k is denoted as $I_k^- = (e_k^-, p_k)$, an ordered pair consisting of the a priori gap and the number of transmissions remaining to be made. The a posteriori information available to the event-detector is $I_k = (e_k, p_{k+1})$.

We are now in a position to formally state the problem being addressed in this paper. Consider a cost function of the form

$$J_M(B; \mathcal{S}_{0|p_0=B}, \cdots, \mathcal{S}_{M|p_0=B}) = E\left[\sum_{k=0}^{M} \hat{e}_k^2 \,|\, p_0 = B\right] \quad (6)$$

where the expectation is taken over $\hat{e}_0, \ldots, \hat{e}_M$, $\tau^1, \cdots, \tau^B$and $\mathcal{S}_{k|p_r=b} = \left\{S_k^{b-max\{0,b-(k-r)\}}, \cdots, S_k^b\right\}$ is the collection of all possible trigger sets needed at time step k, given b transmissions remaining at step r. The objective is to find the optimal trigger sets minimizing the cost function:

$$J_M^*(B) = \min_{\mathcal{S}_{0|p_0=B}, \cdots, \mathcal{S}_{M|p_0=B}} J_M(B, \mathcal{S}_{0|p_0=B}, \cdots, \mathcal{S}_{M|p_0=B}). \quad (7)$$

IV. Main Results

The problem in equation (7) can be treated as the optimal control of a stochastic process. In our case, the control variable is the trigger set S_r^b, rather than some "control signal". Since this is a dynamic optimization problem, we can use a stochastic version of Bellman's principle of optimality to obtain a backward recursion that generates the value function for our problem. That value problem characterizes the cost (as measured by the MSEE at the remote observer) from any initial system state. This section introduces the recursion used to compute the value function and discusses some properties of the value function and the optimal trigger sets.

The value function for our problem is defined in a manner that is analogous to what is commonly done in stochastic dynamic programming. In particular, our value function is defined as

$$v(\zeta, b; r) = \min_{\mathcal{S}_{r|p_r=b}, \cdots, \mathcal{S}_{M|p_r=b}} E\left(\sum_{k=r}^{M} \hat{e}_k^2 \quad | \quad I_r^- = (\zeta, b)\right), \quad (8)$$

which is the minimal expected cost conditioned on the information $I_r^- = (e_r^-, p_r)$ at time r. Because the information sequence $\{I_0^-, I_0, ..., I_M^-, I_M\}$ is Markov (lemma 7.2), the value function in equation (8) is only conditioned on the current information, rather than all past information. The main result of this section is stated below. This theorem provides a backward recursion that can be used to calculate the value function defined in equation (8). The theorem's proof is given in the appendix.

Theorem 4.1: The value function (8) satisfies the backward recursive equation:

$$\begin{aligned} &v(\zeta, b; r) = \\ &\min\Big\{\bar{P}_r + \zeta^2 + E_{e_{r+1}^-}(v(e_{r+1}^-, b; r+1)|I_r = (\zeta, b)) \ , \\ &\bar{P}_r + E_{e_{r+1}^-}(v(e_{r+1}^-, b-1; r+1)|I_r = (0, b-1))\Big\}, \end{aligned} \tag{9}$$

with initial condition:

$$v(\zeta, 0; r) = \begin{cases} \frac{Q(M+1-r)}{1-a^2} + (\bar{P}_r + \zeta^2 - \frac{Q}{1-a^2})\frac{1-a^{2(M+1-r)}}{1-a^2}, & \text{if } |a| \neq 1; \\ \frac{Q(M+r)(M+1-r)}{2} + (\bar{P}_r + \zeta^2 - rQ)(M+1-r), & \text{if } |a| = 1, \end{cases} \tag{10}$$

for $r = 1, \ldots, M$ and

$$v(\zeta, b; r) = \sum_{k=r}^{M} \bar{P}_k \tag{11}$$

where $r = M + 1 - b$ for a given $b \in [1, \ldots, B]$. where $\bar{P}_r$ and e_r are defined in equations (1) and (5) respectively. The optimal triggering sets are

$$\begin{aligned} S_r^{b*} = \{\zeta : \zeta^2 + E_{e_{r+1}^-}(v(e_{r+1}^-, b; r+1)|I_r = (\zeta, b)) \leq \\ E_{e_{r+1}^-}(v(e_{r+1}^-, b-1; r+1)|I_r = (0, b-1)\}, \end{aligned} \tag{12}$$

with $S_r^{0*} = \mathcal{R}, \forall r = 0, 1, ..., M$ and $S_r^{b*} = \{0\}$,where $r = M + 1 - b$.

The first term in equation (9) is the minimum conditional cost of not transmitting at step r whereas the second term is the cost incurred if the sensor does transmit at time step r to the remote observer. This first term consists of the predicted a priori cost, $E_{e_{r+1}^-}\left[v(e_{r+1}^-, b; r+1)|I_r = (\zeta, b)\right]$, at step $r+1$ plus the additional cost, $\bar{P}_r + \zeta^2$, incurred by the current step. The second term has a similar structure with some important differences since this term characterizes the cost that's incurred if the sensor transmits at step r. If such a transmission occurs, then the gap, e_r^-, equals zero and there are $b-1$ transmissions remaining. The predicted a priori cost therefore becomes, $E_{e_{r+1}^-}\left[v(e_{r+1}^-, b-1; r+1)|I_r = (0, b-1)\right]$, and the cost increment reduces to $\bar{P}$, the two terms that are seen in the second term of equation (9).

What should be apparent in examining equation (9) is that the optimal cost at time step r is based on the choice between the costs of transmitting or not transmitting at step r. The actual values that those two costs take is conditioned on the value, ζ, that the a priori gap, e_r^-, takes at time step r. This means we can use the choice in equation (9) to identify two mutually disjoint sets; the trigger set S_r^{b*} and its complement.

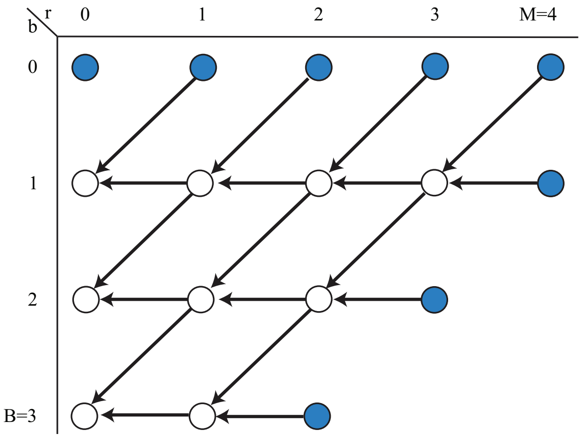

Fig. 2. Order of calculating value function with M=4, B=3

If e_r^- is not in the set S_r^{b*}, then we trigger a transmission otherwise the sensor decides not to transmit its information.

Equation (9) is a backward recursion that recurses over two set of indices; the time steps, r, and the remaining transmissions b. The value function, $v(\zeta, b; r)$, at time step r with b remaining transmissions is computed from the value functions, $v(\zeta, b; r+1)$ and $v(\zeta, b-1; r+1)$, at time step $r+1$ with b and $b-1$ remaining transmissions, respectively. The initial conditions for this recursion are given in equations 10 and 11 in the theorem. Equation 10 specifies the value function when at time step $r \in [0, M]$ there are no transmissions remaining ($b = 0$). These initial conditions are easily computed as the total MSEE assuming no further measurement updates. Equation 11 specifies the value function when there are $b \in [0, B]$ transmissions remaining at time step $M + 1 - b$. This initial condition equals the MSEE assuming an update at each remaining time step. We can picture the recursion as shown in figure 2 . This picture plots the indices (b, r) and identifies the initial conditions and the order of computation. The blue dots in the graph show the initial value functions given in equations 10 and 11. The arrows show the computational dependencies in the recursion.

If we let $R = 0$, $\mu_0 = 0$, and $\Pi_0 = 0$, then theorem 4.1's result is equivalent to the backward recursion derived by Rabi in [22]. In particular for the single sample case ($B = 1$), the cost function, $V_r^1(\zeta)$ introduced in [22] is related to our value function through the relation

$$v(\zeta, 1; r) = \frac{1 - a^{2(M+1-r)}}{1 - a^2}\zeta^2 + \sum_{k=r}^{M}\sum_{i=n}^{k-1} a^{2(i-r)}Q - V_r^1(\zeta).$$

Substituting the above equation into equation (9) yields a backward recursion for $V_r^1(\zeta)$ of the form

$$V_r^1(\zeta) = \max\left\{\zeta^2 \frac{1 - a^{2(M+1-r)}}{1 - a^2}, E\left[V_{r+1}^1(\zeta) \,|\, x_r = \zeta\right]\right\},$$

which is identical to the recursion used in [22] to compute Rabi's cost function $V_r^1(\zeta)$.

The backward recursion in theorem 4.1 is used to compute the minimum cost associated with solving our problem. That cost is formally given in the following corollary.

Corollary 4.2: $J^*_M(B) = E_{e_0^-}(v(e_0^-, B; 0))$.

Proof:

$$\begin{aligned} & E_{e_0^-}(v(e_0^-, B; 0)) \\ = \; & E_{e_0^-}\left[\min_{\mathcal{S}_{0|p_0=B},\ldots,\mathcal{S}_{M|p_0=B}} E\left(\sum_{k=0}^{M} \hat{e}_k^2 | I_0^- = (\zeta, B)\right)\right] \\ = \; & \min_{\mathcal{S}_{0|p_0=B},\ldots,\mathcal{S}_{M|p_0=B}} E_{e_0^-}\left[E\left(\sum_{k=0}^{M} \hat{e}_k^2 | e_0^- = \zeta, p_0 = B\right)\right] \\ = \; & \min_{\mathcal{S}_{0|p_0=B},\ldots,\mathcal{S}_{M|p_0=B}} E\left(\sum_{k=0}^{M} \hat{e}_k^2 | p_0 = B\right) \\ = \; & J^*_M(B). \end{aligned}$$

■

We can use equations (9) and (12) to identify properties of the value functions that help us obtain a simple form of the optimal triggering set. These properties are stated in the following corollaries which are presented without proof.

Corollary 4.3: With b fixed, the value function $v(\zeta, b; r)$ is symmetric about the y-axis and nondecreasing with respect to $|\zeta|$.

Corollary 4.4: The optimal trigger set S_r^{b*} is in the form of $[-\theta_r^b, \theta_r^b]$.

With corollary 4.4, we can change our triggering event into $|e_r^-| > \theta_r^b$. Instead of finding the optimal set S_r^{b*}, we can search for the optimal threshold θ_r^b.

V. SIMULATION RESULTS

This section presents simulation results characterizing the performance of the event-triggered estimation problem. We first introduce an algorithm to compute the value function and optimal trigger sets. We then use these triggers in a simulation and compare the performance of optimal event-triggered transmissions against periodically triggered transmissions.

The value functions characterized in theorem 4.1 cannot usually be computed in closed form. We must therefore determine the value function (and associated event-triggers) using algorithmic methods. We now describe that algorithm.

From theorem 4.1 we know that the value function $v(\zeta, b; r)$ may be computed as the minimum of two functions

$$f_r^b(\zeta) = \overline{P}_r + \zeta^2 + E_{e_{r+1}^-}\left[v(e_{r+1}^-, b; r+1) \,|\, I_r(\zeta, b)\right]$$

and

$$g_r^b = \overline{P}_r = E_{e_{r+1}^-}\left[v(e_{r+1}^-, b-1; r+1) \,|\, I_r = (0, b-1)\right].$$

For the scalar systems under consideration, these two functions may be rewritten as

$$\begin{aligned} f_r^b(\zeta) &= \bar{P}_r + \zeta^2 + E_{e_{r+1}^-}(v(e_{r+1}^-, b; r+1) | I_r = (\zeta, b)) \\ &= \bar{P}_r + \zeta^2 + g_{r+1}^b \\ &\quad - \int_{-\theta_{r+1}^b}^{\theta_{r+1}^b} (g_{r+1}^b - f_{r+1}^b(x)) p_{e_{r+1}^-|e_r=\zeta}(x) dx \end{aligned}$$

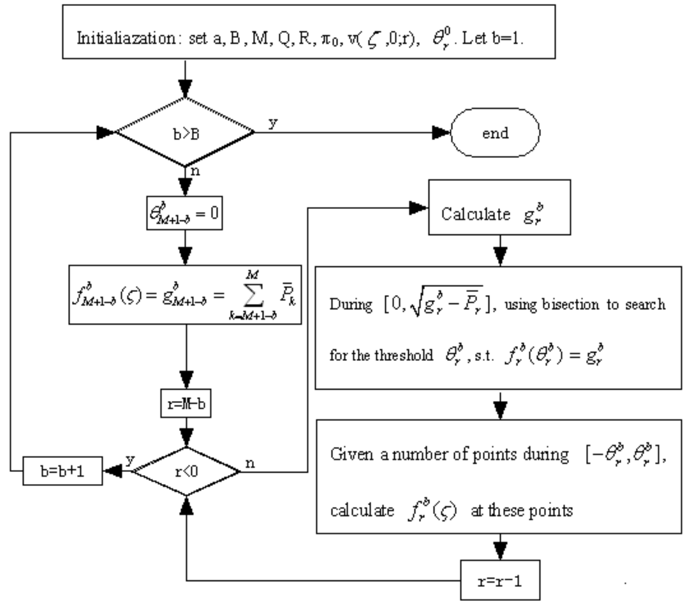

Fig. 3. flowchart of calculating the value function and optimal threshold

and

$$\begin{aligned} & g_r^b = \bar{P}_r + E_{e_{r+1}^-}\left[v(e_{r+1}^-, b-1; r+1) | I_r = (0, b-1)\right] \\ & = \bar{P}_r + g_{r+1}^{b-1} - \int_{-\theta_{r+1}^{b-1}}^{\theta_{r+1}^{b-1}} \left(g_{r+1}^{b-1} - f_{r+1}^{b-1}(x)\right) p_{e_{r+1}^-|e_r=0}(x) dx. \end{aligned}$$

where $p_{x|y}$ denotes the probability density function of x conditioned on y. At step r with b samples remaining, we first calculate g_r^b, and then use a bisection algorithm to search for the optimal threshold θ_r^b such that $f_r^b(\theta_r^b) = g_r^b$. We then evaluate the value function at a number of points between $[-\theta_r^b, \theta_r^b]$.

The flowchart in figure 3 illustrates the algorithmic steps used in evaluating the value function and event thresholds. With remaining samples b fixed, we first initialize the value functions and the optimal thresholds for $r = M + 1 - b$, and then calculate the value functions and the optimal threshold for step $M - b$ backward to step 0. After finishing the calculation for b remaining samples at all steps, we go on with the calculation for the $b + 1$ remaining samples until $b > B$.

Consider a scalar system with $a = 1.2$, $\mu_0 = 1$, $\Pi_0 = 2$ and $Q = R = 1$. Fix the terminal step M to be 8. Figure 4 plots the MSEE for optimal event-triggered and periodically triggered transmissions as a function of the total number of transmissions, B. The plot shows that the experimentally observed MSEE equals the predicted MSEE, thereby validating the correctness of our analysis. The plot also shows that optimal event-triggered transmissions always generate a smaller total MSEE than comparable periodically triggered systems.

Figure 5 shows two plots for the single sample case where $B = 1$. The top plot graphs the event threshold θ_r^1 as a function of the time step r for different initial covariances,

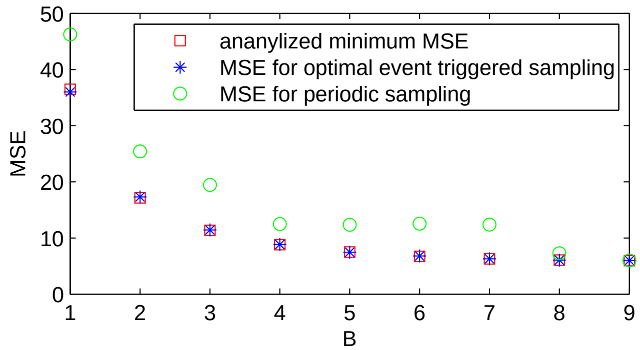

Fig. 4. Event-triggered versus Periodic Transmission MSEE

Π_0. This plot shows that larger initial covariances will initially result in larger optimal thresholds. At later time steps, however, the thresholds asymptotically approach each other. The bottom plot in figure 5 graphs the mean transmission time τ^1 as a function of the initial covariance, Π_0. This plot shows that large initial covariances cause the system to transmit (on average) more quickly than systems with small initial covariances.

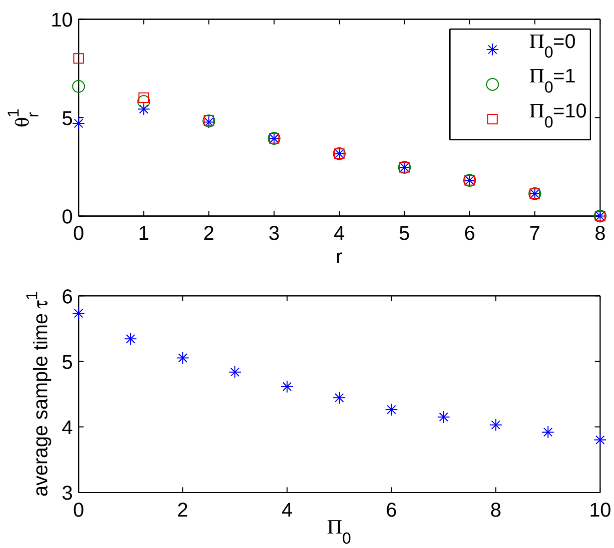

Fig. 5. Influence of Π_0 on optimal threshold and transmission time

VI. Future Work

This paper discussed the design of optimal event-triggers for distributed state estimation problems with finite terminal time and a fixed number of transmissions. This paper solves the problem using stochastic dynamic programming and recovers the earlier results in [1] that were derived using optimal stopping arguments. This paper extends the results in [1] to account for nonzero mean initial conditions with measurement noise. Such extensions could have also been achieved using optimal stopping methods.

Future work will try to extend our approach to vector systems. Backward recursions similar to those in theorem 4.1 may be obtained for vector systems. This recursion takes the following form

$$\begin{aligned}&v(\zeta, b; r) = \\ &\min\{trace(\bar{P}_r) + \zeta^T\zeta + E_{e_{r+1}^-}(v(e_{r+1}^-, b; r+1)|I_r = (\zeta, b)), \\ &\quad trace(\bar{P}_r) + E_{e_{r+1}^-}(v(e_{r+1}^-, b-1; r+1)|I_r = (0, b-1))\},\end{aligned}$$

and the optimal set is

$$\begin{aligned}S_r^{b*} = \{\zeta : \zeta^T\zeta + E_{e_{r+1}^-}(v(e_{r+1}^-, b; r+1)|I_r = (\zeta, b)) \leq \\ E_{e_{r+1}^-}(v(e_{r+1}^-, b-1; r+1)|I_r = (0, b-1))\}.\end{aligned}$$

Using this recursion to compute the value function and triggering sets is much more difficult than the scalar case. As the system's dimension increases, the number of points used to evaluate the $v(\zeta, b; r)$ increases in an exponential manner. Moreover, the triggering set can no longer be described by a simple one dimensional interval. The only practical approach may be to compute good approximations to the triggering thresholds, in much the same way as was done in [23].

VII. Appendix

The following two lemmas are used in the proof of theorem 4.1. These lemmas are presented without proof.

Lemma 7.1: $\bar{e}_k$ is independent with e_j^- and e_j for any $j \leq k$.

Proof: This follows direction from the fact that e_j^-, e_j, and $\bar{e}_k$ are Gaussian and our use of a MSE estimator. ∎

Lemma 7.2: The sequence $\{I_0^-, I_0, I_1^-, I_1, ..., I_{k-1}, I_k^-, I_k, ..., I_M\}$ is a Markov chain.

Proof: The dynamics of e_k^- and e_k are summarized below

$$\begin{aligned} e_k^- &= \overline{x}_k - \hat{x}_k^- \\ &= a e_{k-1} + L_k a \bar{e}_{k-1} + L_k(w_k + v_k) \qquad (13) \\ e_k &= \begin{cases} e_k^- & e_k^- \in S_k^b \\ 0 & \text{otherwise} \end{cases} \qquad (14)\end{aligned}$$

From lemma 7.1, we know that $\bar{e}_{k-1}$ is independent of $e_{k-1}, e_{k-1}^-, \ldots, e_0^-$. We also know that w_k and v_k are independent of $e_{k-1}, e_{k-1}^-, \ldots, e_0^-$. Therefore

$$u_k = L_k a \bar{e}_{k-1} + L_k(w_k + v_k)$$

is also independent of $e_{k-1}, e_{k-1}^-, \ldots, e_0^-$. Note that the number of transmissions

$$p_{k+1} = \begin{cases} p_k - 1 & \text{if } e_k^- \notin S_k^{p_k} \\ p_k & \text{otherwise} \end{cases}$$

with $p_0 = B$. So p_{k+1} is a function of p_0 and e_j^- for $j \leq k$. This means that u_k is also independent of p_j for $j \leq k+1$. So we can conclude u_k is independent of the information sets, $I_{k-1}, I_{k-1}^-, \ldots, I_0^-$.

Consider the conditional probability density function $f(I_k^- \mid I_{k-1}, I_{k-1}^-, I_{k-2}, \ldots, I_0^-)$. From equation (13) and since $I_k^- = (e_k^-, p_k)$, we see that

$$\begin{aligned} & f(I_k^- \mid I_{k-1}, I_{k-1}^-, \ldots, I_0^-) \\ = \;& f(e_k^-, p_k \mid I_{k-1}, I_{k-1}^-, \ldots, I_0^-) \\ = \;& f(ae_{k-1} + u_k, p_k \mid I_{k-1}, I_{k-1}^-, \ldots, I_0^-) \end{aligned}$$

This shows that $I_k^- = (ae_{k-1} + u_k, p_k)$ is a linear combination of $I_{k-1} = (e_{k-1}, p_k)$ and u_k. We showed above that u_k is independent of $I_{k-1}, I_{k-1}^-, \ldots, I_0^-$. So the conditional probability may be written as

$$f(I_k^- \mid I_{k-1}, I_{k-1}^-, \ldots, I_0^-) = f(I_k^- \mid I_{k-1})$$

which implies that $I_k^-, I_{k-1}, I_{k-1}^-, \ldots, I_0^-$ are Markov.

Notice that $S_k^{p_k}$ is not a function of e_j^- or e_j for all j. Eq.(14) shows that e_k is only a function of I_k^-. p_{k+1} is also a function of I_k^-. So the conditional PDF of $I_k = (e_k, p_{k+1})$ over $I_k^-, I_{k-1}, ..., I_0^-$, can be written as $f(I_k | I_k^-, I_{k-1}, ..., I_0^-) = f(I_k | I_k^-)$. ■

Theorem 4.1 Proof The value function satisfies

$$v(\zeta, b; r) = \min_{\mathcal{S}_{r|p_r=b}} \min_{\mathcal{S}_{r+1,M|p_r=b}} E\left[\sum_{k=r}^{M} \hat{e}_k^2 | I_r^- = (\zeta, b)\right]$$

where $\mathcal{S}_{\alpha,\beta|p_k=b} = \{\mathcal{S}_{\alpha|p_r=b} \cdots \mathcal{S}_{\beta|p_r=b}\}$. Notice that $\mathcal{S}_{r|p_r=b} = \{S_r^b\}$. If $e_r^- \in S_r^b$, $\bar{x}_r$ isn't transmitted to the remote estimator, so that $e_r = e_r^- = \zeta$ and $p_{r+1} = b$. Otherwise, $\bar{x}_r$ will be transmitted, and $e_r = 0$, $p_{r+1} = b-1$. Therefore,

$$\begin{aligned} & v(\zeta, b; r) = \\ & \quad \min_{S_r^b} \left\{ \left[\min_{S_{r+1,M|p_{r+1}=b}} E\left[\sum_{k=r}^{M} \hat{e}_k^2 | I_r = (\zeta, b)\right]\right] 1_{\zeta \in S_r^b} \right. \\ & \left. + \left[\min_{S_{r+1,M|p_{r+1}=b-1}} E\left[\sum_{k=r}^{M} \hat{e}_k^2 | I_r = (0, b-1)\right]\right] 1_{\zeta \notin S_r^b} \right\} \end{aligned} \quad (15)$$

Let's consider the first term in the above equation. This is the term associated with "not" transmitting at step r. Recall that $\hat{e}_r$ is independent of $\mathcal{S}_{r+1,M|p_{r+1}=b}$ so that

$$\begin{aligned} & \min_{S_{r+1,M|p_{r+1}=b}} E\left[\sum_{k=r}^{M} \hat{e}_k^2 | I_r = (\zeta, b)\right] \\ = \;& E\left[\hat{e}_r^2 | I_r = (\zeta, b)\right] \\ & + \min_{S_{r+1,M|p_{r+1}=b}} E\left[\sum_{k=r+1}^{M} \hat{e}_k^2 | I_r = (\zeta, b)\right] \\ = \;& E\left[\hat{e}_r^2 | I_r = (\zeta, b)\right] \\ & + \min_{S_{r+1,M|p_{r+1}=b}} E_{e_{r+1}^-}\left[h | I_r = (\zeta, b)\right], \end{aligned}$$

where

$$h = E\left[\sum_{k=r+1}^{M} \hat{e}_k^2 | I_{r+1}^- = (e_{r+1}^-, b), I_r = (\zeta, b)\right].$$

From equations (2), (3) and (5), we can see that $\hat{e}_r = \bar{e}_r + e_r$. Lemma 7.1 asserts that $\bar{e}_r$ is independent with e_r. We also know that $\bar{e}_r$ is independent from p_{r+1}, so that

$$\begin{aligned} E\left[\hat{e}_r^2 | I_r = (\zeta, b)\right] &= E\left[\bar{e}_r^2 + e_r^2 | I_r = (\zeta, b)\right] \quad (16) \\ &= \bar{P}_r + \zeta^2. \quad (17) \end{aligned}$$

Expanding $\hat{e}_k^2$ in our expression for h yields

$$\begin{aligned} h &= E\left[\sum_{k=r+1}^{M} \bar{e}_k^2 + e_k^2 + 2e_k\bar{e}_k | I_{r+1}^- = (e_{r+1}^-, b), I_r\right] \\ &= E\left[\sum_{k=r+1}^{M} \bar{e}_k^2 | I_{r+1}^- = (e_{r+1}^-, b), I_r\right] \quad (18) \\ & \quad + E\left[\sum_{k=r+1}^{M} e_k^2 | I_{r+1}^- = (e_{r+1}^-, b), I_r\right] \quad (19) \end{aligned}$$

The Markov nature of the information sets then yields

$$h = E\left[\sum_{k=r+1}^{M} \hat{e}_k^2 | I_{r+1}^- = (e_{r+1}^-, b)\right] \quad (20)$$

The first term of eq.(15) can therefore be rewritten as

$$\begin{aligned} & \min_{S_{r+1,M|p_{r+1}=b}} E\left[\sum_{k=r}^{M} \hat{e}_k^2 | I_r = (\zeta, b)\right] \\ & = \bar{P}_r + \zeta^2 + E_{e_{r+1}^-}\left[\min_{S_{r+1,M|p_{r+1}=b}} h | I_r = (\zeta, b)\right] \\ & = \bar{P}_r + \zeta^2 + E_{e_{r+1}^-}\left[v(e_{r+1}^-, b; r+1) | I_r = (\zeta, b)\right]. \end{aligned}$$

where h is given in equation 20.

Following almost the same steps, we show that the second term of eq.(15) equals

$$\begin{aligned} & \min_{S_{r+1,M|p_{r+1}=b-1}} E\left[\sum_{k=r}^{M} \hat{e}_k^2 | I_r = (0, b-1)\right] \\ = \;& \bar{P}_r + E_{e_{r+1}^-}\left[v(e_{r+1}^-, b-1; r+1) | I_r = (0, b-1)\right], \end{aligned}$$

thereby yielding the backward recursive equation with the specified optimal trigger sets

The initial condition given in equation 10 is obtained as follows. $v(\zeta, 0; r)$ means that no transmissions occur between step r to step M. Since $\hat{e}_k = a\hat{e}_{k-1} + w_k$ then for all $k \geq r+1$ we can see that

$$\begin{aligned} & E\left[\hat{e}_k^2 | I_r^- = (\zeta, 0)\right] \\ = \;& E\left[a^2\hat{e}_{k-1}^2 + w_k^2 + 2aw_k\hat{e}_{k-1} | I_r^- = (\zeta, 0)\right] \\ = \;& a^2 E\left[\hat{e}_{k-1}^2 | I_r^- = (\zeta, 0)\right] + Q \end{aligned}$$

Using equation (17) and some algebra, we see that

$$E\left[\hat{e}_k^2 | I_r^- = (\zeta, 0)\right] = a^{2(k-r)}(\bar{P}_r + \zeta^2) + \sum_{i=r+1}^{k} a^{2(k-i)} Q.$$

Note that $v(\zeta, 0; r) = E\left[\sum_{k=r}^{M} \hat{e}_k^2 | I_r^- = (\zeta, 0)\right]$. So with some algebra, we obtain the initial condition in equation (10) .

The initial condition in equation 11 is obtained as follows. This condition is the MSEE from time step r assuming there is a transmission in each of the remaining time steps. This means that $\hat{x}_k = \overline{x}_k$ for these remaining time steps. Since the covariance of $\overline{x}_k$ is $\overline{P}_k$, the total MSEE from r to the last time step is simply $\sum_{k=r}^{M} \overline{P}_k$ as given in equation 11. ■

Corollary 4.3 Proof: This result can be established through mathematical induction. ■

Corollary 4.4 Proof: $S_r^{0*} = \mathcal{R}, \forall r = 0, 1, ..., M$, and $\theta_r^0 = \infty$. $S_r^{b*} = \{0\}, \forall r = M+1-b, ..., M$, and $\theta_r^b = 0$, $\forall r = M+1-b, ..., M$. For the other cases,

$$S_r^{b*} = \{\zeta : \zeta^2 + E_{e_{r+1}^-}(v(e_{r+1}^-, b; r+1)|I_r = (\zeta, b)) \leq E_{e_{r+1}^-}(v(e_{r+1}^-, b-1; r+1)|I_r = (0, b-1))\}.$$

From corollary 4.3 we can show that the second term is a constant and the first term is symmetric about y-axis and increasing about $|\zeta|$. So S_r^{b*} must be in the form of $[-\theta_r^b, \theta_r^b]$. ■

REFERENCES

[1] M. Rabi, G. Moustakides, and J. Baras, "Multiple sampling for estimation on a finite horizon," in *Decision and Control, 2006 45th IEEE Conference on*, 2006, pp. 1351–1357.

[2] K. Koumpis, L. Hanna, M. Andersson, and M. Johansson, "Wireless industrial control and monitoring beyond cable replacement," in *2nd Profibus International Conference*, 2005.

[3] G. Walsh, H. Ye, and L. Bushnell, "Stability analysis of networked control systems," in *American Control Conference, 1999. Proceedings of the 1999*, vol. 4, 1999.

[4] W. Zhang, M. Branicky, and S. Phillips, "Stability of networked control systems," *IEEE Control Systems Magazine*, vol. 21, no. 1, pp. 84–99, 2001.

[5] D. Nesic and A. Teel, "Input-output stability properties of networked control systems," *IEEE Transactions on Automatic Control*, vol. 49, no. 10, pp. 1650–1667, 2004.

[6] B. Sinopoli, L. Schenato, M. Franceschetti, K. Poolla, M. Jordan, and S. Sastry, "Kalman filtering with intermittent observations," *IEEE Transactions on Automatic Control*, vol. 49, no. 9, pp. 1453–1464, 2004.

[7] Y. Tsypkin, *Relay Control Systems*. Cambridge University Press, 1984.

[8] K. E. Arzen, "A simple event based pid controller," in *Proc. 14th IFAC World Congress*, 1999.

[9] Tabuada, "Event-triggered real-time scheduling of stabilizing control tasks," *IEEE transactions on automatic control*, vol. 52, no. 9, p. 1680, 2007.

[10] X. Wang and M. Lemmon, "Self-triggered feedback control systems with finite-gain l 2 stability," *IEEE transactions on automatic control*, vol. 54, p. 452, 2009.

[11] ——, "Event-triggering in distributed networked systems with data dropouts and delays," in *Proceedings of Hybrid Systems: computation and control*, 2009.

[12] K. Astrom and B. Bernhardsson, "Comparison of riemann and lebesgue sampling for first order stochastic systems," in *Proceedings of the IEEE Conference on Decision and Control*, vol. 2, 2002, pp. 2011–2016.

[13] J. Sandee, W. Heemels, and P. van den Bosch, "Case studies in event-driven control," in *Hybrid Systems: Computation and Control*. Springer, 2007, pp. 762–765.

[14] P. Wan and M. D. Lemmon, "Distributed network utility maximization using event-triggered augmented lagrangian methods," in *Submitted to American Control Conference*, 2009.

[15] Y. Xu and J. Hespanha, "Optimal communication logics in networked control systems," in *43rd IEEE Conference on Decision and Control, 2004. CDC*, vol. 4, 2004.

[16] A. Matveev and A. Savkin, "The problem of state estimation via asynchronous communication channels with irregular transmission times," *IEEE transactions on automatic control*, vol. 48, no. 4, pp. 670–676, 2003.

[17] B. Sinopoli, L. Schenato, M. Franceschetti, K. Poolla, M. Jordan, and S. Sastry, "Kalman filtering with intermittent observations," *Automatic Control, IEEE Transactions on*, vol. 49, no. 9, pp. 1453–1464, Sept. 2004.

[18] X. Liu and A. Goldsmith, "Kalman filtering with partial observation losses," in *Decision and Control, 2004. CDC. 43rd IEEE Conference on*, vol. 4, Dec. 2004, pp. 4180–4186 Vol.4.

[19] B. Zhu, B. Sinopoli, K. Poolla, and S. Sastry, "Estimation over wireless sensor networks," in *American Control Conference, 2007. ACC '07*, July 2007, pp. 2732–2737.

[20] V. Gupta, T. H. Chung, B. Hassibi, and R. M. Murray, "On a stochastic sensor selection algorithm with applications in sensor scheduling and sensor coverage," *Automatica*, no. 2, pp. 251–260, 2 2006.

[21] J. Yook, D. Tilbury, N. Soparkar, E. Syst, and E. Raytheon, "Trading computation for bandwidth: reducing communication indistributed control systems using state estimators," *IEEE transactions on control systems technology*, vol. 10, no. 4, pp. 503–518, 2002.

[22] M. Rabi, G. Moustakides, and J. Baras, "Adaptive sampling for linear state estimation," Submitted to SIAM Journal on Control and Optimization, `http://www.ee.kth.se`, 2008.

[23] R. Cogill, S. Lall, and J. Hespanha, "A constant factor approximation algorithm for event-based sampling," in *American Control Conference, 2007. ACC'07*, 2007, pp. 305–311.

[24] S. Dayanik and I. Karatzas, "On the optimal stopping problem for one-dimensional diffusions," *Stochastic processes and their applications*, vol. 107, no. 2, pp. 173–212, 2003.