Distributed Estimation and Control

Vijay Gupta

May 9, 2006

Abstract

In this lecture, we will touch upon some of the open issues when such estimation has to be carried out in a communication constrained network control environment. We will concentrate on three constraints: topology, quantization issues and packets dropped by communication channels. We will discuss the major research directions in these areas.

Contents

Estimation and control in Networked Control Systems is, by and large, an open problem. There are many effects that communication channels introduce. We will now consider some such effects and state some research directions that are being pursued. For the purpose of estimation, we will only be interested in variables that evolve in time.

1 Effect of Topology

1.1 Distributed Sensor Fusion

We saw that the so-called consensus protocol can be used to generate weighted averages in arbitrary graphs and hence for static sensor fusion. Attempts have been made to apply similar ideas for dynamic sensor fusion in networked control systems as well [15, 1]. To follow the method used, we use the following result proved in the lecture on distributed estimation.

Proposition 1. *Consider a random variable evolving in time as*

$$x(k+1) = Ax(k) + w(k).$$

Suppose it is observed through measurements of the form

$$y(k) = Cx(k) + v(k).$$

Then the measurement updates of the Kalman filter can be given by this alternate information form.

$$\begin{aligned} P^{-1}(k|k)\hat{x}(k|k) &= P^{-1}(k|k-1)\hat{x}(k|k-1) + C^T R^{-1} y(k) \\ P^{-1}(k|k) &= P^{-1}(k|k-1) + C^T R^{-1} C. \end{aligned}$$

Suppose now that the measurement $y(k)$ is obtained by stacking measurements $y_i(k)$ from N sensors, with the i-th sensor having measurement matrix C_i and noise $v_i(k)$ with covariance matrix R_i. Assume that the noises $v_i(k)$ are independent of each other, so that the matrix R is block-diagonal. Thus we obtain

$$\begin{aligned} C^T R^{-1} y(k) &= \sum_{i=1}^{N} C_i^T R_i^{-1} y_i(k) \\ C^T R^{-1} C &= \sum_{i=1}^{N} C_i^T R_i^{-1} C_i. \end{aligned}$$

Thus note that calculating the global Kalman filter estimate essentially requires two quantities: $C^T R^{-1} C$ and $C^T R^{-1} y$ which can be expressed as the sum over similar quantities from each sensor. Since the consensus algorithm yields the average of quantities at many sensors, if every sensor knows the total number N of the sensors present, it can calculate the sum and implement the global Kalman filter.

Practically, the consensus algorithm only yields the average asymptotically hence if there are n communication rounds allowed, the sum calculated is an approximation. Intuitively, the performance loss is equivalent to premultiplication of the global Kalman filter by a low-pass filter determined by the network topology and speed.

1.2 Topology Synthesis in Distributed Control

We saw that the problem of constructing a controller that respects the topology of the agents is a hard problem. However, there is an implicit assumption in that problem that the interconnection topology is known to the designer prior to the synthesis of the control law. This is equivalent to assuming that the decentralization is an external constraint. In networked control, both the communication network and the controller may be designed at the same time. So, for example, when deciding between a leader-follower (star) topology and a string topology, the choice of the information pattern is to be performed concurrently with the controller synthesis. Thus a more fundamental question is to obtain an *optimal topology.* This question has been studied for long in the realm of Team Decision theory (mainly for static variables) [16] but the distributed control aspects have started getting attention only recently [17, 18].

In the setting of [18], consider N agents with the dynamics of the i-th agent being[1]

$$x_i(k+1) = A_i x_i(k) + B_i u_i(k).$$

Let the vector $x_i(k)$ have dimension $n_i \times 1$ and $u(k)$ have dimensions $m_i \times 1$. Also let the vector $x(k)$ formed by stacking the states of all the agents have dimension $n = \sum_{i=1}^{N} n_i$ and similarly $u(k)$ have dimension $m = \sum_{i=}^{N} m_i$. The interconnection topology is depicted by a graph with an edge

[1] [18] considers continuous time dynamics, but the arguments can be carried over to the discrete-time case.

present between two vertices iff the corresponding agents can communicate. Any graph g can then be used to impose a structure on a matrix K as follows. K is a structured $m \times n$ matrix that is defined block-wise, with each block K_{ij} being a $m_i \times n_j$ matrix such that $K_{ij} = 0$ whenever i and j do not share an edge. Thus if a control law is such that

$$u(k) = Kx(k),$$

then any control law $u_i(k)$ involves the values of $x_j(k)$ for only those j that are neighbors of i. If $u(k)$ is of this form, we say that u has structure g.

For such a system, we consider the quadratic cost

$$J_g(u) = \sum_{k=0}^{\infty} \left[x^T(k)Q(g)x(k) + u^T(k)u(k) \right].$$

Note that the cost is a function of the graph g because both the matrix Q and the control law u are functions of the graph. Note that if the cost matrix Q were independent of the topology, the optimal topology would be a completely connected graph and the corresponding control law would be the usual LQ control. There are many ways in which Q can depend on the graph.

1. The map Q can satisfy
$$Q = Q_0 + \sum_{\text{all edges } (i,j)} P_{ij},$$
with each matrix $P_{ij} > 0$ being partitioned according to the agents and having all blocks zero except the (i,i)-th, (i,j)-th, (j,i)-th and (j,j)-th ones. So, every subsystem pays a price for every edge it transmits its state on.

2. Q is given as above, but P_{ij} has the same structure as Q. Thus every subsystem pays a price for any edge added to the system.

3. There is no edge-by-edge separation in Q.

For a given weight matrix Q, we refer to the value of the cost function obtained by using the optimal control law u that satisfies the topology of the graph g as $J_g^\star(Q)$.

If a graph g is a subgraph of another graph g' then we write $g \preceq g'$. The following results are obvious.

Proposition 2. *1. If $g \preceq g'$ then for a fixed Q, $J_{g'}^\star(Q) \leq J_g^\star(Q)$.*

2. If $Q \leq Q'$ then for any graph g, $J_g^\star(Q) \leq J_g^\star(Q')$.

Comparing the values of costs for arbitrary topologies is difficult chiefly because finding the optimal control law that satisfies a given topology is a difficult problem. For some special cases, however, the problem yields simplifications.

Proposition 3. *Consider a mapping of topologies into the weighting matrices Q such that for any graph g, the matrix $Q(g)$ has the same structure as the matrix K. Then if the graph g is a clique graph then $J_g^\star(Q) = x^T(0)P(g)x(0)$, where $P(g)$ is the unique positive definite solution to the Riccati equation*

$$P(g) = A^T P(g) A + Q - A^T P(g) B \left(B^T P(g) B + R \right)^{-1} B^T P(g) A.$$

A clique graph is one each of whose connected components is a complete subgraph. Thus for such a graph, if the weighting matrix follows the assumption stated above, then the system can simply be treated as a collection of subsystems with each subsystem corresponding to a sub-graph. The problem is totally decoupled and the resulting LQ optimal control law automatically satisfies the structure imposed by the topology. Thus the cost function stated in the proposition results.

If in addition the mapping is such that a graph g being a sub-graph of another graph g' implies $Q(g) \leq Q(g')$, then it is possible to compare the optimal cost obtained by a clique graph to its supergraphs.

Proposition 4. *If the mapping of topologies into weighting matrices satisfies the assumptions stated above, and g be a clique graph, then $J_g^\star(Q(g)) \leq J_{g'}^\star(Q(g'))$ whenever $g \preceq g'$.*

Proof. Proof follows by a straight-forward set of inequalities. First we note that $J_g^\star(Q(g)) \leq J_{g'}^\star(Q(g))$ since g is a clique graph and hence the optimal LQ control law has structure imposed by the graph g. But $J_{g'}^\star(Q(g)) \leq J_{g'}^\star(Q(g'))$ since $Q(g) \leq Q(g')$. Combining the inequalities yields the desired result. □

An immediate consequence of the result is that if the mapping satisfies the assumptions stated above, the fully decentralized graph is the most *efficient* graph.

The above result does not enable us to compare two arbitrary topologies. In [18] the authors provide some LMI conditions to find out the critical price at which adding an edge to a given topology becomes detrimental. However the LMIs assume that the control law is fixed (say a consensus type law, e.g.). If the LMIs have to be solved both for the control law and the price, they turn into bilinear matrix inequalities that are NP-hard to solve in general.

Thus the problem of synthesizing the optimal topology has only been solved in some specific cases. The general problem remains open. It is of interest in many networked control systems to obtain the best possible topology in which to connect the different agents. Solving the topology synthesis question will also shed light on this problem.

2 Effect of Quantization

In last week's lectures, the following result was proven.

Proposition 5. *(See, e.g., [2, 3, 4]) Consider a system of the form*

$$x(k+1) = Ax(k) + w(k)$$

being observed by a single sensor of the form

$$y(k) = Cx(k) + v(k).$$

If (A, C) is observable, then a necessary condition for asymptotic observability over a noiseless digital channel is that the rate[2] *R satisfies*

$$R \geq \sum_{\lambda(A)} \max\{0, \log \mid \lambda(A) \mid\}.$$

[2] This is unfortunate notation since earlier R was being used for a covariance matrix and now for the rate vector. The context hopefully makes the meaning clear.

A sufficient condition is that

$$R > \sum_{\lambda(A)} \max\{0, \log \mid \lambda(A) \mid\}.$$

For a system being observed by two sensors with observation matrices C_1 and C_2, a similar result holds. Define $\mathcal{N}_n$ to be the unobservable subspace associated with (A, C_n) and $\mathcal{O}_n$ to be the subspace orthogonal to $\mathcal{N}_n$[3]. Assume that $\cap_{n=1}^{N} \mathcal{N}_n = \phi$. Thus any $x \in \mathcal{O}_n$ can be observed by encoder n.

Associate with each n the set

$$\Lambda_n = \{\lambda(A) : \text{ those eigenvalues of } A \text{ that correspond to the subspace } \mathcal{O}_n\}.$$

Thus every eigenvalue of A is in at least one such set. Since the eigenvalue may be in multiple sets, encoders can jointly transmit information about it. This freedom is captured in the following result. For the n-th sensor, define its rate vector to be

$$R_n = \left(R_{n,1}, R_{n,2}, \cdots, R_{n,\dim(\mathcal{O}_n)}\right).$$

Proposition 6. *(From [5]) A necessary and sufficient condition on the rate for asymptotic observability is that the vector of rates used by all the sensors lies in the region $\mathcal{R}$ defined by*

$$\mathcal{R} = \{(R_1, \cdots, R_N) : \sum_{n:\lambda(A)\in\Lambda_n} R_{n,j_{\lambda(A)}} \geq \max\{0, \log \mid \lambda(A) \mid\}, \forall \lambda(A)\}$$

where $j_{\lambda(A)}$ represents the index of the rate vector associated with that eigenvalue.

Proof. Proof is straightforward. For any eigenvalue that only one sensor observes, the rate from that sensor should be enough to guarantee observability of that subspace. If multiple sensors observe some eigenvalue, then the rate required for that subspace can be split among them. □

This result solves the problem of one source and multiple sensors for digital noiseless channels. The result can be extended when the channels are replaced by networks of digital noiseless channels. The quantity of interest then becomes the min-cut capacity of the network. The problem for multiple sources can be solved using the above results if we simply stack the states on top of one another. Above results were for the case of a noiseless system. When noise is present, the situation is much less well-understood [6]. Similarly the control performance of the system in presence of quantization is largely an open problem.

3 Effect of Packet Drops

This has also become a region of active research lately. Results that are known are limited in nature. There are two main directions of research:

[3] As an interesting exercise, prove that $\mathcal{N}_n$ is indeed a subspace.

3.1 Measurements being Transmitted

It may be assumed a priori that the quantity to be transmitted is the measurement. This case was considered for the single sensor-estimator case in the class a couple of weeks ago by Bruno. Essentially the average error covariance for i.i.d. packet drops is characterized through an upper and a lower bound [7]. These provide sufficient and necessary conditions for convergence. That result has been extended in two directions. For the case of multiple sensors observing the same process which can all drop measurements independently of each other, the situation is again described by a rate region [8]. The analysis is very similar to the single sensor case and is left as an exercise.

If multiple sensors observe the same process such that some sensors may never drop measurements, the analysis (especially of the lower bound) becomes slightly more involved. However this case has also been analyzed [9]. This approach can also extend the upper and lower bounds for the average error covariance for the case when packet drops are not i.i.d. Let us look into this analysis in a little more detail.

Consider a system of the form

$$x(k+1) = Ax(k) + w(k)$$

that is observed at every time step by one among N different sensors

$$y_i(k) = C_i x(k) + v_i(k).$$

Let the noises $w(k)$ and $v_i(k)$ be zero mean, white, Gaussian, independent of each other and with variances Q and R_i respectively. Suppose sensor i is chosen with probability q_i at every time step. The following questions may be asked.

1. What is the performance of the optimal estimator?

2. What is the optimal schedule for choosing the various sensors?

It is obvious that the optimal estimator is a Kalman filter. For ease of notation, we denote the Riccati operator

$$f_i(P) = APA^T + Q - APC_i^T \left(C_i P C_i^T + R_i \right)^{-1} C_i P A^T.$$

Thus $f_i(P)$ denotes the estimate error covariance at time step $k+1$ is the i-th sensor was used at that time and the covariance at time step k was P. It is obvious that the error covariance $P(k)$ evolves in a stochastic fashion. To characterize it, we look at its expected value. The following properties of the Riccati operator are easy to prove.

Proposition 7. *1. For a positive semi-definite P, $f_i(P)$ is concave in P.*

2. For two positive semi-definite matrices X and Y, if $X < Y$, then $f_i(X) < f_i(Y)$.

3. For any positive semi-definite matrix X, $f_i(X) \geq Q$.

Explicit calculation of the expected error covariance seems hard. We look instead for upper and lower bounds.

The upper bound is obtained by using the concavity of the Riccati operator.

Proposition 8. *For the problem set-up given above, if the sensors are chosen i.i.d., then the expected error covariance of the estimate at time step $k+1$ is upper-bounded by $\Delta(k+1)$ where*

$$\Delta(k+1) = Q + A\Delta(k)A^T - \sum_{i=1}^{N} q_i A\Delta(k)C_i^T \left(C_i\Delta(k)C_i^T + R_i\right)^{-1} C_i\Delta(k)A^T,$$

with the initial condition $\Delta(0) = P(0)$, the covariance of $x(0)$.

Proof. Proof is simple. First note that $P(k)$ and $C(k)$ (the sensor chosen at time step k) are independent. Thus we can explicitly take expectation w.r.t. $C(k)$ to obtain

$$E\left[P(k+1)\right] = \sum_{i=1}^{N} q_i E\left[f_i\left(P(k)\right)\right],$$

where the expectation on the right hand side is now only over $C(0)$, $\cdots$, $C(k-1)$. Now owing to the concavity of the Riccati operator, we can use Jensen's inequality on the right hand side to obtain

$$\begin{aligned} E\left[P(k+1)\right] &= \sum_{i=1}^{N} q_i E\left[f_i\left(P(k)\right)\right] \\ &\leq \sum_{i=1}^{N} q_i f_i\left(E\left[P(k)\right]\right). \end{aligned}$$

Since the Riccati operator is an increasing operator, the upper bound is immediate. □

The convergence of the recursion for the upper bound is also considered in [9]. The lower bound can be obtained in multiple ways. We present the method in [9]. Define the repeated Riccati recursion operator as

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

Proposition 9. *For the problem setup given above, if the sensors are chosen in an i.i.d. fashion, then the expected error covariance at time step k is lower bounded by $X(k)$ where*

$$X(k) = q_j^k f_j^k(P(0)) + \sum_{i=1}^{k-1} q_j^i \left(1 - q_j^{k-i-1}\right) f_j^i(Q).$$

One such lower bound results for each $j =, 1, 2, \cdots, N$. Further, the lower bound diverges (hence a sufficient condition for the divergence of the expected error covariance) is

$$q_j \mid \lambda_{\max}\left(\bar{A}_j\right) \mid^2 \leq 1,$$

where $\lambda_{\max}\left(\bar{A}_j\right)$ is the eigenvalue with the maximum magnitude of the unobservable part of A when the pair (A, C_j) is put in the observer canonical form.

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

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

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

1. When the sensor j is chosen at time step m, the error covariance at the time step $m+1$ is given by $f_j(\Sigma)$ where Σ was the error covariance at the present time step.

2. When any other sensor is chosen the corresponding error covariance at time step $m+1$ is lower bounded by Q. Moreover if, then, at time step $m+1$, the j-th sensor is chosen, the error covariance at time step $m+2$ is lower bounded by $f_j(Q)$.

By combining these two facts, we see that

$$V\left(E_i\right) \geq f_j^i\left(Q\right).$$

Thus we obtain

$$p(E_i) V\left(E_i\right) \geq q_j^i \left(1 - q_j^{k-1-i}\right) f_j^i\left(Q\right).$$

For the term E_k, from the definition we obtain

$$p(E_k) V\left(E_k\right) = q_j^k f_j^k(P(0)).$$

By adding together the terms $p(E_i)V\left(E_i\right)$, we obtain the lower bound.

The proof for the divergence condition is given in [9] and is omitted here. □

Similar bounds can be obtained [9] when the sensors are chosen in a Markovian fashion. If one of the sensors corresponds to measurements being dropped, these bounds and conditions for divergence reduce to the ones that were presented by Bruno. These conditions can also be used to construct an optimal schedule for choosing sensors. Obviously the optimal schedule can only be obtained using a tree search but optimizing the expected error covariance over the probabilities of choosing various sensors can yield a stochastic schedule that performs well *on an average.*

3.2 Optimal Encoding for Estimation

A more general approach is to find coding strategies, that is find the optimal quantities to be transmitted by each sensor so that the estimator can obtain the optimal estimate (given that packets are being dropped). It is obvious that the problem of estimation (and control) in a Networked system is partly that of information transmission. While the usual setting in information theory that deals with the questions of information transmission is insensitive to questions of delay, the general idea of encoding information before transmission over a communication channel is relevant to estimation and control. We saw some strategies that dealt with digital noiseless channels in the last section when we considered quantization strategies that achieved the minimum bit-rate for stability and also when we considered distributed estimation to reduce the computation at a central node when the agents were connected by a given topology. In this section, we will look at encoding strategies with respect to channels that drop packets.

Let us begin with the simplest case: there is one sensor present that transmits over a single packet dropping channel to an estimator. If the sensor transmits measurements, the performance has been looked at in [7]. The optimal encoding and decoding for this case has been looked at in [12]. Consider a system of the form

$$x(k+1) = Ax(k) + Bu(k) + w(k)$$

being observed by a sensor of the form

$$y(k) = Cx(k) + v(k),$$

with the usual assumptions on the noises. The sensor transmits its information to the controller over a channel that drops packets. We assume that the packets have enough bits so that quantization error is not an issue. For this system consider the usual quadratic cost

$$J = E\left[\sum_{k=0}^{K}\left(u^T(k)Q^c u(k) + x^T(k)R^c x(k)\right) + x^T(k+1)P(k+1)x(k+1)\right].$$

We wish to minimize this cost function over the choice of the controller, the encoding done at the sensor and the decoding done at the controller. The following result holds.

Proposition 10. *For the single sensor, estimator and channel setting considered above, an optimal solution is the following:*

- *The encoder is a Kalman filter that calculates the estimate $\hat{x}(k|k)$ of the state $x(k)$ at every time step.*
- *The decoder is a switched linear filter. If it receives a packet from the encoder, it outputs that as the current estimate. Otherwise it time updates the last estimate it calculated and outputs that.*
- *The controller is the LQ optimal controller. It uses the estimate that the decoder outputs instead of the state value.*

Proof. The details of the proof are in [12]. Essentially the proof follows the following steps.

1. The best performance is achievable if the encoder transmits the entire measurement history at every time step k. That is the maximum information set that the controller can possibly have access to and hence the performance is the best in this case.
2. For such an encoder, a separation principle holds. The decoder calculates the minimum mean squared estimate of the state $x(k)$. The controller uses that estimate in the LQ optimal control.
3. Thus the same performance is achievable as long as the controller has access to the state estimate. The encoder and decoder presented in the result ensure this.

□

Note that the encoder and decoder alone solve the optimal estimation problem. Also note that the solution given above is optimal for any packet-dropping process (not necessarily i.i.d.). Moreover the solution is optimal even though the encoder and the decoder have a recursive structure, and hence require a constant amount of memory and computation. Another nice feature of the solution is that as soon as the decoder receives any packet from the encoder, its estimate is as good as if it had received all previous packets. Hence previous packets do not matter anywhere.

In the version presented above, the encoder needed to know the control input. Since the Kalman filter is linear, the effect of the control input can also be taken care of at the decoder (which is collocated with the controller and hence knows the control input). For other properties of the algorithm, including optimality in the presence of delays, see [12].

Paradoxically, it turns out that the optimal strategy presented above is much easier to analyze then transmitting measurements alone. We can, for instance, calculate the expected error covariance. Stability and performance results for specific packet drop models are given in [12]. As an illustration, the following result holds for channels that drop packets in a Markovian fashion.

Proposition 11. *Consider the control problem defined above. Let the pair (A, B) be stabilizable and (A, C) be detectable. Then the system is stabilizable in the bounded variance sense if and only if*

$$q_{22} \mid \rho(A) \mid^2 < 1,$$

where q_{22} is the probability of dropping two packets consecutively and $\rho(A)$ is the spectral radius of A.

Since the strategy is optimal, this condition is necessary for any other encoding strategy. Further if the condition is satisfied, one scheme that stabilizes the system (or in case of estimation yields the optimal estimate) is presented above.

Assuming a provision of time-stamping, the strategy can be extended to the case of estimation and control over networks composed of packet erasing channels as well [10, 11]. The strategy remains similar. The optimal encoder is a Kalman filter. At every time step, any intermediate node considers the packets on all incoming edges and the packet in its own memory from the last time step. It looks at the time stamp of the packets, chooses the packet with the latest measurement, time updates it and transmits it on all out-going edges. This simple algorithm can be shown to be optimal. All the nice properties carry over from the single channel case. Stability and performance of the system can be analyzed. As an illustration the following result holds for stability.

Proposition 12. *Consider a process being estimated over a network of channels that drop packets independent of each other as well as in an i.i.d. fashion in time. Let the j-th channel have packet drop probability γ_j. Then the expected estimate error covariance is stable if and only if*

$$p_{\textit{max-cut}} \mid \rho(A) \mid^2 < 1,$$

where $p_{\textit{max-cut}}$ is the max-cut capacity of the network. To calculate the max-cut capacity, form all possible cut-sets of the network such that the sensor is in the source *set and the estimator in the* sink *set. For any cut-set, define the cut-set capacity as*

$$p_{\textit{cut-set}} = \prod_{i=1}^{p} \gamma_i,$$

where the product is taken over all the p edges that join a node in the source set to a node in the sink set. Then the max-cut capacity is given by

$$p_{\textit{max-cut}} = \max_{\textit{all possible cut-sets}} p_{\textit{cut-set}}.$$

Thus the cut-set probability is the equivalent probability for stability purposes for any network. As an example, consider a series network of n links each with probability of erasure p. Then the equivalent drop probability using the optimal coding strategy is p. As a comparison, the equivalent probability if only measurements were transmitted and no encoding was done would be the reliability of the network which is $1-(1-p)^n$. For $n = 5$ and $p = 0.2$, this evaluates to a difference between 0.2 and 0.7. Thus there are huge gains to be had by encoding. It may also be noted that the cut-set probability is equivalent only for stability purposes and the performance calculation is harder.

The performance results, e.g., can be used for the purpose of synthesizing a network. To consider a simple example, consider a scalar system observed by sensor s. Assume that the destination is located at distance d_0 from the sensor. The probability of dropping a packet on a link depends on its physical length. A reasonable model for probability of dropping packets is given by[4]

$$p(d) = 1 - \exp(-\beta d^{\alpha}),$$

where β, α are positive constants. α denotes the exponent of power decay in the wireless environment. We can easily calculate the optimal number n of relay nodes that we should place between sensor and the destination so as to minimize the expected steady-state error covariance.

For the case of multiple sensors, the optimal encoding and decoding has been identified for the case when only one sensor has a channel that drops packets [13, 14]. The problem is largely open for multiple packet dropping channels.

References

[1] D.P. Spanos, R. Olfati-Saber, and R.M. Murray, "Distributed Sensor Fusion Using Dynamic Consensus", IFAC 2005

[2] S. Tatikonda, "Control under Communication Constraints", PhD Thesis, MIT, Cambridge, MA 2000.

[3] R. W. Brockett and D. Liberzon, "Quantized Feedback Stabilization of Linear Systems", IEEE Transactions on Automatic Control, 45(7), 2000, pp. 1279-89.

[4] G. N. Nair and R. J. Evans, "Stabilizability of Stochastic Linear Systems with Finite Feedback Data Rates", SIAM Journal on Control and Optimization, 43(2), July 2004, pp. 413-436.

[5] S. Tatikonda, "Some Scaling Properties of Large Distributed Control Systems," 42nd IEEE Conference on Decision and Control, December 2003 (Maui, Hawaii).

[6] A. Sahai, "Anytime Information Theory," PhD thesis, Massachusetts Institute of Technology, February 2001.

[4] This expression can be derived by considering the probability of outage in a Rayleigh fading environment.

[7] B. Sinopoli, L. Schenato, M. Franceschetti, K. Poolla, M. Jordan and S. Sastry, “Kalman Filtering with Intermittent Observations”, IEEE Transactions on Automatic Control, Special Issue on Networked Control Systems, September 2004.

[8] X. Liu and A. J. Goldsmith, “Kalman Filtering with Partial Observation Losses,” IEEE Conference on Decision and Control, 2004.

[9] V. Gupta, T. Chung, B. Hassibi and R. M. Murray, ”On a Stochastic Sensor Selection Algorithm with Applications in Sensor Scheduling and Dynamic Sensor Coverage,” Automatica, Vol. 42, No. 2, February 2006, Pages: 251-260.

[10] V. Gupta, A. F. Dana, J. P. Hespanha and R. M. Murray, ”Data Transmission over Networks for Estimation”, 17th International Symposium on Mathematical Theory of Networks and Systems, (MTNS 06). Accepted.

[11] A. F. Dana, V. Gupta, J. P. Hespanha, B. Hassibi, R. M. Murray, “Estimation over Communication Networks: Performance Bounds and Achievability Results,” Submitted, 2006 Conference on Decision and Control.

[12] V. Gupta, D. Spanos, B. Hassibi and R. M. Murray, ”Optimal LQG Control Across Packet-Dropping Links”, Proceedings of the American Control Conference, 2005 (ACC 05). Page(s):360 - 365.

[13] V. Gupta, D. Spanos, B. Hassibi and R. M. Murray, ”Optimal LQG Control Across Packet-Dropping Links”, System and Control Letters, Submitted July 2005. Available at http://www.cds.caltech.edu/∼vijay

[14] V. Gupta, B. Hassibi and R. M. Murray, ”On Sensor Fusion in the Presence of Packet-dropping Communication Channels”, IEEE Conference on Decision and Control 2005 (CDC 05). Page(s): 3547-3552.

[15] D.P. Spanos, R. Olfati-Saber, and R.M. Murray, ”Dynamic Consensus on Mobile Networks”, IFAC 2005.

[16] J. Marschak and R. Radner, “Economic Theory of Teams”, Cowles Foundation Monograph 22, Yale University Press, 1972.

[17] R. Smith and F. Hadaegh, “Closed-loop Dynamics of Cooperative Vehicle Formations with Parallel Estimators and Communication”, Submitted to IEEE Transactions on Automatic Control, 2005.

[18] C. Langbort and V. Gupta, “Minimal Interconnection Topology in Distributed Control Design,” Proceedings of the American Control Conference, ACC 2006.