LUND UNIVERSITY

Event-Based Control and Estimation with Stochastic Disturbances

Henningsson, Toivo

2008

Link to publication

Total number of authors:
1

Event-Based Control and Estimation with Stochastic Disturbances

Toivo Henningsson

Department of Automatic Control
Lund University
Lund, November 2008

Till Maria och Arthur

Abstract

This thesis deals with event-based control and estimation strategies, motivated by certain bottlenecks in the control loop. Two kinds of implementation constraints are considered: closing one or several control loops over a data network, and sensors that report measurements only as intervals (e.g. with quantization). The proposed strategies depend critically on *events*, when a data packet is sent or when a change in the measurement signal is received. The value of events is that they communicate new information about stochastic process disturbances.

A data network in the control loop imposes constraints on the event timing, modelled as a minimum time between packets. A threshold-based control strategy is suggested and shown to be optimal for first-order systems with impulse control. Different ways to find the optimal threshold are investigated for single and multiple control loops sharing one network. The major gain compared to linear time invariant (LTI) control is with a single loop a greatly reduced communication rate, which with multiple loops can be traded for a similarly reduced regulation error.

With the bottleneck that sensors report only intervals, both the theoretical and practical control problems become more complex. We focus on the estimation problem, where the optimal solution is known but untractable. Two simplifications are explored to find a realistic state estimator: reformulation to a mixed stochastic/worst case scenario and joint maximum a posteriori estimation. The latter approach is simplified and evaluated experimentally on a moving cart with quantized position measurements controlled by a low-end microcontroller.

The examples considered demonstrate that event-based control considerably outperforms LTI control, when the bottleneck addressed is a genuine performance constraint on the latter.

Acknowledgments

First, I would like to thank my supervisor Anton Cervin for many rewarding discussions about the proper way to do event-based control. Anton has also been a great support, available to give good advice and pitching in when it was most needed (i.e. close to deadlines). I would also like to thank my cosupervisor Anders Rantzer, who gave me valuable feedback on some of the parts in the thesis. Great thanks go also to Karl Johan Åström, who supervised my master's thesis and introduced me to the exciting field of event-based control.

I have enjoyed many interesting discussions with my coworkers about event-based control as well as other things; special thanks go to Peter Alriksson who got to listen to all my ideas and tried to understand what I was doing. Thanks also to Erik Johannesson for the work we did together on Paper I. I would like to thank Maben Rabi, for rewarding discussions on event-based control, and for inviting me to spend a very enjoyable day to visit him at KTH. Proofreading and feedback on the material in this thesis was kindly given by Karl Johan Åström, Anders Rantzer, Anton Cervin, Peter Alriksson, Ather Gattami, and Maria Karlsson.

Many thanks go to Anders Blomdell and Rolf Braun, without whom my experimental work would not have gone nearly as smoothly. Leif Andersson deserves credit for computer support and help with the typesetting of this thesis. Thanks also to the department secretaries Eva Schildt, Agneta Tuszynski, Britt-Marie Mårtensson, and Eva Westin for making the department run so smoothly and helping out with all administrative matters.

Acknowledgments

I want to thank Lasse, Kerstin, Rasmus, Harald, and Maja for all their support and for valuable breaks from the world of research. Last, and most importantly, I want to thank Maria for all her love and support and Arthur for making my life brighter.

Contents

Contents

Preface

The field of event-based control is a rich subject area that contains many interesting control and estimation problems. In many practical cases, it turns out that sizable performance benefits can be gained by application of some kind of event-based control when there is a bottleneck in the control loop of event-based type.

The available theory is limited, however, mainly because of the mathematical difficulties involved. It seems that all event-based problems become nonlinear, and often non-convex, causing the closed-form solutions exploited in standard linear control theory to break down. If there exist general classes of event-based control problems that can be solved by systematic means at reasonable complexity, they remain to be discovered. This means that selecting the proper problem formulation is as important as solving the problem.

Thus there remains a wealth of event-based control problems to investigate, and different methods to attempt their solutions. The focus of this thesis is on two such problem formulations. The first is control with constraints on the timing of events when the control loop is closed by sending packets over a data network. The second is control and estimation with sensors that report only an interval for the measured quantity, e.g. with quantization.

To not unduly complicate matters, all problem formulations are kept as close as possible to a standard linear time invariant problem formulation, while incorporating the event-based constraint of interest. An effort has been made to compare the achievable performance with the proposed event-based strategies against what is achievable with linear strategies.

Contributions of the Thesis

The thesis consists of four introductory chapters and five papers, the papers falling into the two categories of event-based control under communication and sensing constraints, respectively. This section describes the contents of the introductory chapters and the contributions of each paper.

Chapter 1 – Event-Based Control

This chapter gives a general overview of the field of event-based control and estimation and tries to distinguish some of the various directions that go under this broad heading. The general approach taken in the thesis is outlined.

Chapter 2 – Control under Communication Constraints

The event-based control problems that arise when packet-based communication forms a constraint on the control loop are described. The approach of Papers I and II, that are formulated in this setting, is outlined.

Chapter 3 – Control under Sensing Constraints

The event-based estimation and control problems that arise with sensing constraints such as quantized measurements in the control loop are described. The approach of Papers III, IV, and V, that deal with this problem, are outlined.

Chapter 4 – Outlook

This chapter outlines some possible directions for future work.

Paper I

Henningsson, T., E. Johannesson, and A. Cervin (2008): "Sporadic event-based control of first-order linear stochastic systems." *Automatica*, **44:11**, pp. 2890–2895.

This paper introduces the concept of sporadic control to model the practical constraint of a minimum time between any two control events. Two variants of sporadic controllers are proposed, using continuous

and discrete time measurements. It is shown for first-order systems with impulse control and white noise disturbances that the optimal sporadic controller will use a threshold strategy, generating a control event whenever it is allowed and the state becomes big enough. Ways of computing the optimal threshold and the associated performance in terms of regulation and event rate are described. The best achievable tradeoff between regulation error and event rate is characterized for the sporadic controllers, periodic and aperiodic controllers, showing that many of the benefits of aperiodic control are retained even with the practical constraint of a minimum inter-event time. It is also shown that sporadic control can not only greatly reduce the required communication rate, but also reduce the regulation error somewhat, compared to periodic control. Different possibilities for the generalization of the control problem at hand to higher dimensional systems are discussed.

T. Henningsson wrote on and did the simulations for the continuous-time case, and rewrote the paper for *Automatica*. E. Johannesson wrote on and did the simulations for the discrete-time case. A. Cervin wrote the introduction and assisted in the structuring and editing of the manuscript.

This paper is an extension of

Johannesson, E., T. Henningsson, and A. Cervin (2007): "Sporadic control of first-order linear stochastic systems." In *Proc. 10th International Conference on Hybrid Systems: Computation and Control*, Lecture Notes in Computer Science 4416. Springer-Verlag, Pisa, Italy.

Paper II

Cervin, A. and T. Henningsson (2008): "Scheduling of event-triggered controllers on a shared network." In *Proc. 47th IEEE Conference on Decision and Control*. Cancún, Mexico. To appear.

This paper treats the control problem when several loops of the same type as in Paper I are closed over a shared network. Models for the medium access protocols TDMA, FDMA, and CSMA, the latter with three different prioritization mechanisms, are stated. A suitable control setup is described for each case, and ways to evaluate the expected

regulation error and choose optimal parameters for the controllers are described, some based on Monte Carlo simulations. The performance when controlling N integrator plants is compared for the protocols, showing that CSMA quickly yields superior performance, asymptotically requiring only a third of the bandwidth or giving a third of the error compared to any of the other protocols. A case study for control of one stable, one integrator, and one unstable plant is also presented, yielding similar conclusions.

A. Cervin wrote most of the paper and performed most of the simulations. T. Henningsson derived the optimal control policy for the case of two integrators and shared information between the controllers, using dynamic programming for controlled Markov processes.

Paper III

Henningsson, T. and K. J. Åström (2006): "Log-concave observers." In *Proceedings of the 17th International Symposium on Mathematical Theory of Networks and Systems*. Kyoto, Japan.

This paper investigates the problem of Bayesian state estimation with log-concave measurement likelihoods and process noise, and the properties of log-concave functions that can be used to say something about the state estimation problem. The optimal Bayesian state estimator for discrete-time linear systems is described, split up into dynamics, process noise, and measurement updates, each acting on the probability distribution of the state conditioned on the measurements. The case when measurement likelihoods and process noise densities are log-concave — a considerable generalization of the Kalman filter setting — is investigated. It is shown that the conditional state density will be log-concave. Using the concept of strongly log-concave functions, theorems are derived that allow to upper bound the estimation error covariance of the Bayesian estimator. The upper bound is compared in simulations to a grid-based approximation of the Bayesian estimator and a Kalman filter designed using insight gained on the estimation problem, for a double integrator with quantized measurements.

T. Henningsson wrote most of the paper. K. J. Åström did extensive reviewing.

Paper IV

Henningsson, T. (2008): "Recursive state estimation for linear systems with mixed stochastic and set-bounded disturbances." In *Proc. 47th IEEE Conference on Decision and Control*. Cancún, Mexico. To appear.

This paper presents a recursive state estimator for linear systems that are subject to both stochastic and uncertain (set-bounded) process and measurement disturbances. The structure for a recursive state estimator is proposed, allowing to model general state estimation problems with combined stochastic and set-bounded measurement and process disturbances. An optimization procedure for selecting the filter gain considering both the resulting stochastic and set-bounded error is described. For the case of ellipsoidal uncertainty sets, an LMI is formulated to optimize the feedback gain and fit the best one-step optimal ellipsoidal overbound on the set-bounded uncertainty, available using the S-procedure. The estimator is compared in simulations to a grid filter and a time-varying Kalman Filter for the case of a double integrator with quantized measurements, where it comes quite close to the almost optimal grid filter.

Paper V

Henningsson, T. and A. Cervin (2009): "Comparison of LTI and event-based control for a moving cart with quantized position measurements." Submitted to European Control Conference 2009, Budapest, Hungary.

This paper investigates practical event-based velocity control of a moving cart with quantized position measurements. Preliminary linear control designs are performed to gain insight into the control problem and the tradeoffs involved between robustness and disturbance rejection. A joint maximum a posteriori (JMAP) state estimator for the process is described. Insight into the behaviour of the JMAP estimator is used to simplify it into an event-based estimator that can run online on a low-end microcontroller. Implementation issues on the microcontroller are discussed, including discretization, fixed-point arithmetic with fast approximate division, real-time tasks and concurrency issues. A suitable step-by-step implementation of the event-based estimator to

allow to verify the correctness of the implementation between each step is described. Different development stages of the event-based controller are compared experimentally to the linear control designs, investigating the tradeoff between control effort and regulation error and also the impact of quantization. It is demonstrated that event-based control can reduce the control effort drastically when quantization starts to become a problem for linear controllers.

T. Henningsson did the experiments and wrote most of the paper. A. Cervin assisted in the structuring and editing of the manuscript.

Other Publications

Henningsson, T. and A. Rantzer (2007): "Scalable distributed Kalman filtering for mass-spring systems." In *Proc. 46th IEEE Conference on Decision and Control*. New Orleans, LA.

This paper has not been included in the thesis because it is not on the topic of event-based control.

1

Event-Based Control

The term *event-based control* can stand for many things, depending on the scenario that motivates it. This chapter tries to distinguish some of the varying flavors of event-based control, outline some of the challenges involved, and describe the general approach of the thesis.

1.1 Event-Based Control Problems

By *event*, we mean an ahead-of-time unspecified time instant when something important happens in the system. What constitutes an event will depend on the problem formulation. By *event-based* we mean a control strategy that employs events to deal with a constraint or bottleneck in the system (see Figure 1.1). The special case of *event-triggered* control denotes the case when events trigger the execution of controller tasks.
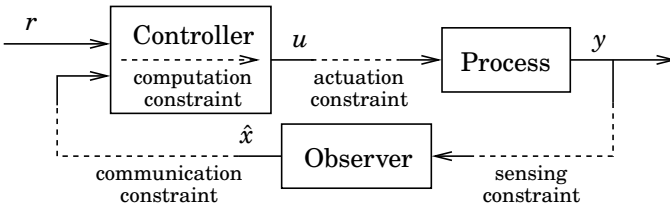
Figure 1.1 Control loop with some constraints motivating event-based control.

Different flavors of event-based control arise depending on the nature of the bottleneck that motivates its use. Examples of bottlenecks are sensor quantization, communication over a data network, cost per actuation, and limitations on CPU processing power. We will now attempt to categorize some of these flavors according to the kind of bottleneck involved.

Figure 1.1 shows a prototypical control loop with process, observer (state estimator), and controller. The view is from a single control loop; there may actually be multiple processes and controllers involved, connected in diverse topologies. Some bottlenecks that can motivate event-based control are marked in the figure:

Communication constraints. All modern data communication networks are based on *packet switching*, which means that the smallest unit of data that is transmitted over the network is the *data packet*. An event-based controller can save energy and free up communication bandwidth for others by sending a packet — and thus generating an event — only when something significant has happened. This is especially important in wireless networks, where bandwidth is generally quite scarce, and the medium is a shared resource that can only be used by one node at a time.

A packet consists of a *header*, specifying e.g. the format and destination, and a data *payload*. Because of the overhead involved in handling each packet, in most simple control problems, all relevant information can be sent in every packet at little additional cost. Thus the important constraint on real-time control imposed by data networks is not the available bit rate per se, but the constraints on the timing of data transmission events.

Sensing constraints. It is not uncommon with sensors that give measurements only at ahead of time unknown events. Common examples are rotary motion encoders that give pulses at fixed angular increments and A/D-converters with coarse resolution, where each change in the measured value can be seen as an event. Another example is sensors that transmit data over a network using the *send on delta* protocol: Transmit a new measurement $y(k)$ only if $|y(k) - y_{\text{last transmitted}}| \geq \Delta$, for some tolerance Δ. This protocol is used e.g. in building automation, and standardized in the LONWorks standard, according to [Vasyutynskyy and

Kabitzsch, 2007]. In common for these scenarios is the fact that not only events, but also their *absence*, contain some information about the measured value.

The objective in this case is to make as good use of the available measurements as possible, possibly to close a control loop. If the information contained in the absence of events can be disregarded, a time varying Kalman filter provides a solution of moderate implementation complexity. When there is a need to exploit the absence of events, which is often the case if there may be long, event-free periods, the problem becomes much more challenging.

Actuation constraints. Some actuators can only be turned either on or off, such as satellite thrusters [Dodds, 1981]. Another important case of actuation constraints is when it is desired to avoid to change control signals more often than necessary. One example is in plant-wide control, where it is known that every control action will cause upsets in other control loops.

The objective in this case is to find a control strategy that gives a good tradeoff between regulation performance and few actuation events. The freedom to choose the timing of events can compensate for some of the lack of spatial resolution in the actuators. A simple example that exploits timing, but does not keep down the number of events, is the commonly used technique of pulse-width modulation.

Computation constraints. The majority of all controllers today are implemented using computers. In cost-sensitive applications, it is often desirable to keep down the amount of processing power available to the control task.

By using event-triggered control, where the control task is only triggered at relevant events, CPU time can sometimes be freed up. The event trigger conditions are often very simple, and sometimes the control law can be simplified as well if it needs to be computed only when the trigger conditions are satisfied. Though event-triggered control can actually improve performance somewhat given the same amount of available processing power, the

real gain, however, lies in the lowered CPU burden in stationarity, allowing to serve background tasks or save energy.

In *self-triggered control*, the controller calculates both a control signal and a suitable sleep interval before it should wake up and calculate another one, essentially adapting the sampling period depending on the process state. Self-triggered control is considered for linear systems in [Lemmon *et al.*, 2007] and for nonlinear systems in [Tabuada, 2007].

Another kind of computation constraint arises when the controller has to deal with the timing variations inherent with most real time scheduling protocols, which can cause severe problems for periodic control designs, see [Cervin *et al.*, 2003].

Other motivations for event-based control include cases when the process itself is event-based, such as the queues in a web server, or to avoid tight demands for synchronization in a distributed system imposed by a time triggered framework. A diverse list of applications that motivate event-based control is found in [Åström, 2007].

For some of the categories of control problems outlined above, an element of non-determinism is essential for a meaningful problem formulation. This goes for communication and sensing constrained control, since both communication and measurements are pointless in a deterministic setting. The actuation- and computation-constrained cases, on the other hand, are far from trivial even in a deterministic setting.

Even though the categorization of event-based problems above is based on the dominating constraint involved, it should be noted that most of the problems are also computation constrained in some sense. A theoretical method to find the optimal solution is known, though often prohibitively expensive (such as enumerating all possible controllers and evaluating which one is best). Thus, many of the problems can be formulated as trying to find a practical solution with limited computation requirements. By *computation-constrained* control problems, however, we will in this thesis consider only those where computation resources are deliberately scarce.

1.2 When is Event-Based Control Worthwhile?

Using event-based instead of linear control usually implies a considerable complication either in design, implementation, or both. The necessary theory may very well need to be developed from scratch, unlike the mature field of linear time invariant control. In any practical case, one should strive to do the simplest thing that works.

However, when the bottleneck that motivates its use is a real problem, event-based control can be very effective. A more constraining sensor, communication network, or microprocessor can imply considerable savings. Sometimes the bottleneck is unavoidable, but it is still desired to push performance.

In some cases event-based control results in lower utilization of a certain resource; it may be necessary to have something else that can make use of the freed resource in order to realize the benefits. In the case of controllers sharing a network, this can be the other controllers.

1.3 Challenges

Even when designing for a linear process, all event-based problems seem to become nonlinear, and often non-convex or of very high (sometimes infinite) state dimension. With a nonlinear formulation, the closed form solutions exploited in standard linear control break down. This implies that e.g. the addition of stochastic disturbances makes a real difference to the solution as well as the methods by which it can be found, unlike e.g. the LQ-controller, where the design is independent of whether there is any process noise or what characteristics it has.

The field of event-based control is much less mature than linear control, so if there are any general cases that can be solved in a systematic manner and at reasonable complexity, these remain to be discovered. Thus the challenge lies as much in formulating the problem as in solving it, and different methods must be evaluated trying to find ones that work well for particular problems.

Optimal solutions to event-based control problems often involve elements of other known-to-be-hard control problems. One example is distributed control with non-global knowledge (see [Witsenhausen, 1968]), which appears when closing several control loops over the same net-

work. Another example is *dual control* (see [Feldbaum, 1960 1961]), i.e. the controller can improve its performance by exciting the process in order to extract more information, which appears in the case of quantized measurements. In such cases the most realistic goal is probably to approximate the problem in order to find a suboptimal solution which is hopefully still reasonable.

1.4 Approach of the Thesis

Since the topic is largely unexplored, it makes sense to begin with simple problems before moving on to harder ones. Many event-related phenomena should hopefully occur already in the simpler formulations. Thus, for the papers in this thesis:

- An LTI formulation is always considered as the base case, and minimally modified to make the problem event-based. Knowledge about the LTI solution is exploited, if possible.

- Only one bottleneck is considered at a time. Papers I and II deal with communication constraints, Papers III–V with sensing constraints.

- Reasonable amounts of computation are allowed wherever needed in the control loop. This should improve performance and often gives a problem that is easier to deal with by removing artificial computation constraints.

- Solution methods that are simple and readily available are preferred over more elaborate methods, when they are feasible. This may allow to demonstrate the viability of an idea, even though the general case has not been solved.

- The problem of dual control that arises with sensing constraints is avoided by focusing on the estimation problem, applying certainty equivalence for control.

Since the best methods are not known, different methods are tried for the same problem:

- To handle the stochastic process that describes the system, we consider

    - discretization of the system into a Markov chain,

    - Monte Carlo simulation,

    - log-concave probability densities,

    - convex optimization.

- To optimize controllers, we apply

    - brute force gridding of the parameter space,

    - dynamic programming.

Since one should always strive for the simplest solution that works, a serious attempt has been made to compare what is achievable with the suggested event-based schemes and with standard sampled data control [Åström and Wittenmark, 1997].

2

Control under Communication Constraints

This chapter considers the event-based control problem when the constraining resource is packet-based communication, e.g. when the control loop is closed over a data network (see Figure 2.1). Characteristic for this setup is that, while the resulting control laws are often simple to implement, the offline design computations are usually heavy. When closing a single loop over a network, the substantial gains that can be achieved are in reduced communication rate. With multiple loops, event-based control allows the network to be used by the loop that needs it most, so that the gains can be achieved in either reduced communication rate or increased regulation performance. This setup
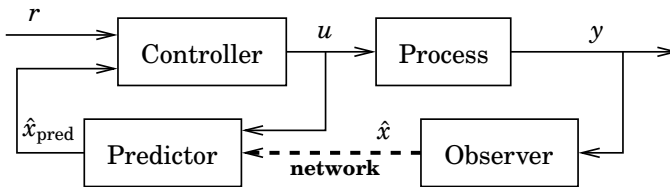
Figure 2.1   Control loop with communication constraints.

should be distinguished from the problem of closing a control loop over a bit-rate constrained channel treated in e.g. [Martins and Dahleh, 2008].

2.1 Making the Most of a Communication-Constrained Control Loop

A key question is how to choose a proper structure for the controller under communication constraints. We want the performance constraints encountered to be genuine and not the artifact of a bad match between controller structure and constraint. Some observations are:

- Since communication is the essential constraint, only *new* information needs to be sent, as opposed to quantities that can be calculated from the data that is already available at a network node. If a node is not capable of the necessary computations, the problem is also *computation constrained*, and probably harder.

- When a packet is sent, there is probably enough room in it to transmit all relevant information, synchronizing sender and receiver. This is clearly better than transmitting only partial information.

- The actual amount of intelligence needed in the receiving end is often rather limited. Following an idea from [Georgiev and Tilbury, 2006], each packet is often big enough that the sender can calculate and transmit a long control signal trajectory, to be used until a new packet is received.

The prototypical control loop in Figure 2.1 was designed based on these considerations: All essential information is transmitted (the state estimate $\hat{x}$), and all necessary computations are carried out on both sides of the network link.

- The Observer monitors the measurements y from the process. Given that it knows the reference r as well as the state estimates $\hat{x}$ received by the Predictor, it can also calculate the control signal u. Thus it knows everything that can be known about the process state, which can be summarized in an optimal state estimate $\hat{x}$, typically using a Kalman filter.

- Since the Observer possesses all information that is available, it is in the best position decide at which events to transmit this knowledge, i.e. the state estimate $\hat{x}$, to the Predictor. The event-triggering rule should be a function of the state of the system, given by $\hat{x}$ and $\hat{x}_{\text{pred}}$.

- At each event, the Predictor is synchronized with the Observer. It then uses the dynamics of the process and the control signal u to form the best prediction $\hat{x}_{\text{pred}}$ of the process state given the information available.

- The Controller uses state feedback from the predicted state estimate $\hat{x}_{\text{pred}}$ to control the process. The state feedback design is a prerequisite to designing the event-triggering rule of the observer.

The Predictor could also make use of the information contained in the absence of an event, but this is not crucial since the Observer will issue an event if disturbances cause too big errors in the control signal or process state.

The problem of when to trigger the sending of a state estimate $\hat{x}$ can be seen as a special case of actuation-constrained control, where the actuation is to update the state estimate in the receiving end. Unlike the general actuation constrained control problem however, in a deterministic setting it will always be optimal to transmit the state estimate right away. Thus it is essential to include an element of non-determinism in the process model.

2.2 Problem Formulations

Many different communication-constrained problem setups are possible, depending on how the system is modeled and which objective that is chosen. One distinction is between estimation problems, where the aim is to minimize the state estimation error at the receiving end, and control problems, where the aim is to keep the process output close to the reference value. The key difference is that in the estimation problem, every event resets the error to zero. If the control is fast enough

to reset the process state to zero before the network becomes available again, there is no great distinction between the control and estimation problems.

Another distinction is between considering single or multiple control loops at the same time. The simplest setup is with a single control loop, trying to achieve the best tradeoff between network utilization and control error. The aim can be to save energy, or to free up communication bandwidth for some unspecified background traffic. A more realistic problem is to consider all the users of the network at the same time, typically multiple control loops. Now it is possible to capture the timing interactions between the loops; as opposed to the single loop case, where the controller typically has the freedom to transmit a packet whenever it is best suited.

Some more elaborate possibilities include using the network for more than one link in the control loop or letting a number of sensors, actuators, and controllers connected by a network cooperate to control one big plant.

Models of Network Constraints

The communication constraint imposed by the data network can be modelled in different ways, yielding different control problems. Since modelling all the intricacies of a real data network leads to a very complex problem formulation, a simplified model is usually desired. Some formulations that have been used in the literature are:

Limited number of packets. Some authors have considered how to maximize performance given a limited number of packets to be used during a finite time, for estimation [Imer and Basar, 2005; Rabi, 2006] and control [Rabi *et al.*, 2008].

Minimize the number of packets. Generalizing to an infinite time setting, a simple criterion is to seek the best tradeoff between state cost and average frequency of events. This is done for first order systems in [Åström and Bernhardsson, 1999; Rabi, 2006], and for systems of arbitrary order, within a factor 6 of the optimal cost, in [Cogill *et al.*, 2006].

Sporadic control. A constraint that applies for most networks in practice is that each time a packet is sent, the network, or at least

the part of it involved, is busy for a certain time. This imposes further restrictions on the control policy, and leads to a simple model of the interaction between several control loops sharing the same network.

The tradeoff between state cost and average frequency of events under the sporadic constraint is treated in Paper I for the single loop case. [Cervin and Johannesson, 2008] extends this setting to deal with delay, jitter, and measurement noise. Paper II treats the case of multiple loops sharing the same network.

Close limited number of control loops at one time. In [Hristu-Varsakelis and Kumar, 2002], the communication constraint is modelled such that only a limited number of control loops can be closed at a given time. This is really a computation constraint, since with a predictor-based controller at the receiving end of each network connection, the communication could be cycled rapidly between the loops to virtually close all loops at the same time.

Heuristic. One heuristic idea to keep network traffic down is to use dead bands on the transmitted sensor signal [Vasyutynskyy and Kabitzsch, 2007; Årzén, 1999], control signal [Sandee, 2006, ch. 5], or state estimate [Wang and Lemmon, 2008; Yook *et al.*, 2002; Otanez *et al.*, 2002].

2.3 Approach of Papers I–II

The starting point of Papers I and II is [Åström and Bernhardsson, 1999]. The paper considers the policy of triggering a control event whenever the state exceeds a certain threshold, which is shown to give a factor of 3 improvement in the ratio of mean time between events and state variance compared to periodic control, in the case of a continuous time integrator with white noise disturbance. This is an *aperiodic* control policy in our terminology, meaning that events may be triggered at arbitrary times.

From a real time systems point of view, however, a control task can not be guaranteed service unless there is a minimum inter-arrival time between events, which we denote as *sporadic* control. Papers I and II

aim to answer the question: If we respect the practical constraint of sporadic control, will there still be sizable benefits compared to periodic control?

The sporadic constraint of a minimum inter-event time is crucial in many practical implementation situations, such as controller tasks implemented in a microprocessor or control signals communicated over a network. An important aspect is that it is perhaps the simplest model that accounts for the interaction of several controller sharing a common resource, be it a microprocessor or a network.

In order to get as good understanding of the usefulness of sporadic control as possible, the problem formulation has been streamlined to the bare essentials that would still capture the sporadic control problem. By minimizing the number of parameters in the system, a good overview can be gained by gridding over the few parameters left.

- A linear process with a Gaussian white noise disturbance allows to use standard theory for linear stochastic processes in parts of the calculations.

- The notion of impulse control streamlines the definition of an event. Other pulse shapes can be accommodated by inserting a linear prefilter at the control input, such as a prediction based state feedback controller.

- The restriction to first order systems minimizes the number of parameters, and allows to discretize the system into a Markov chain without concerns for the curse of dimensionality. A further parameter can be reduced by eliminating the time constant and focusing on integrator processes.

- In the case of multiple control loops, the focus on N instances of the same control loop helps reduce the number of parameters drastically, since it should then be optimal to use the same parameters for all loops.

- The simple strategy of triggering an event whenever the magnitude of the state exceeds a certain threshold is used throughout. This is shown to be optimal in the single loop case. In the multiple loop case, it is not optimal but seems to come close to the optimal cost without introducing unnecessary complications.

The main conclusions can be summarized:

- In the single loop case, the performance with sporadic control is almost as good as with aperiodic if the aim is to reduce the rate of events. If the aim is instead to reduce the state variance, a limited gain is possible, while still reducing the rate of events considerably.

- In the multiple loop case, the network time freed by one control loop can be used by another. The result is that the performance of sporadic control quickly approaches that of aperiodic control, as the number of loops sharing the network increases.

3

Control under Sensing Constraints

This chapter discusses event-based control and estimation when the constraint comes from sensors that only deliver measurements at certain events, such as with coarse quantization (see Figure 3.1). Unlike the communication-constrained case, the optimal estimator or controller can require very heavy online computations. The hardest part of the estimation problem is to handle the information contained in the absence of an event; if this is not needed, a time varying Kalman filter gives a solution.

In the control case, the optimal control strategy requires dual control, where the controller may choose to excite the process so as to extract more information. In practice, approximations are necessary to arrive at any realistic estimator or controller.
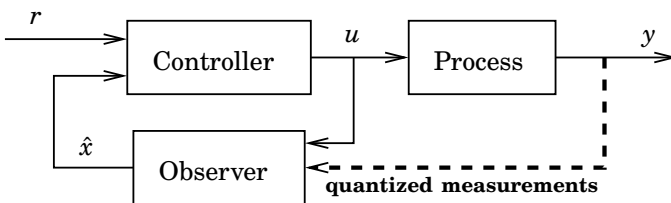
Figure 3.1 Control loop with quantized measurements.

3.1 Problem Setup

The focus of this chapter is a particular problem formulation, depicted in the control loop in Figure 3.1. Let the process have linear dynamics, with a white noise disturbance,

$$dx = Ax\,dt + Bu\,dt + dw,$$

where x is the state, u the control signal, w is a Wiener process with $\mathrm{E}(dw) = 0$, $\mathrm{E}(dw\,dw^T) = R\,dt$, and A and B are dynamics and control input matrices.

The non-classical assumption lies in the measurement model,

$$y = g(x, y_{\text{last}}),$$

which can model e.g. quantization,

$$y = \text{round}(Cx),$$

or send-on-delta transmissions,

$$y = \begin{cases} y_{\text{last}} & |Cx - y_{\text{last}}| < \Delta \\ Cx & \text{otherwise,} \end{cases}$$

where y_{last} is the last measurement transmitted and C is a sensor output matrix. The measurements may be taken continuously, or more realistically, at periodic sampling instants.

Important for the estimation problem is the set $X(y, y_{\text{last}})$ of possible process states x conditioned on that the measurement y was received,

$$X(y, y_{\text{last}}) = \left\{ x; g(x, y_{\text{last}}) = y \right\}.$$

In many interesting cases, such as the examples above, the set X is convex. (In both examples, it is a slab.) This convexity lends some structure to the estimation problem, as is shown in Paper III.

3.2 Solving the Estimation Problem

As mentioned earlier, searching for an optimal controller for the system above generally includes the problem of dual control, where the controller should excite the process for the sole purpose of gaining better information about its state. This is known to be a very hard problem in general [Feldbaum, 1960 1961]. To avoid this issue, we concentrate on the estimation problem, applying certainty equivalence for control even though separation does not hold.

The optimal solution to the state estimation problem is given by the Bayesian estimator, described for the case of quantized measurements in [Curry, 1970] and in Paper III. This involves keeping track of the conditional probability density $f_{X|Y}$ of the state conditioned on the available measurements. The conditional density lives in an infinite dimensional space, and its updates are described by partial differential equations in continuous time, or convolutions, affine transformations and pointwise multiplications in discrete time (see Paper III).

Some observations are:

- In general, the measurement y only gives the information that the state x belongs to some set X, which in general extends along all state dimensions. At *events*, however, when y changes value, by a continuity assumption on the state x, it will lie on a specific part of the boundary of X, a lower dimensional set.

  Thus, events specify the state exactly along some state dimension, and can be used in classical state estimators like the Kalman filter. In discrete time, the assumption of known Cx at events is reasonable with fast sampling.

- Although a measurement usually does not specify the state exactly along any dimension, it is often specified within some absolute error, such as in the examples above. Deriving an estimator trying to minimize the gain from e.g. quantization error to state estimates will give a lower bound on the achievable performance.

3.3 Approximations for Estimation

To obtain a realistic estimator, some kind of approximation is necessary. Some possibilities are:

Use the information in events only. A simple approximation is to disregard the information contained in the absence of an event. With Gaussian process noise, the conditional process state distribution now becomes Gaussian, allowing to use a time varying Kalman filter, see [Kalman, 1960; Kalman and Bucy, 1961]. This is also the approach taken in [Sandee *et al.*, 2007], where the control law is sampled as a function of distance instead of time.

Grid filtering. The state x can be discretized onto a finite grid, turning the stochastic state process into a Markov chain. The estimation problem becomes solvable, but the necessary number of grid points, and thus estimator states, increases exponentially with the state dimension. Papers III and IV use this approach to try to come close to the Bayesian estimator. The computations are heavy but not impossible for the second-order system considered.

Particle filtering. Another approach to state estimation that relaxes the assumptions of the Kalman filter is Particle filtering, where the conditional state density is approximated by a cloud of point densities, see [Arulampalam *et al.*, 2002].

JMAP estimation. Another approximation is to use only the most probable state trajectory given the measurements, resulting in a *joint maximum a posteriori* (JMAP) formulation, see [Cox, 1964]. The advantage is that with log-concave noise densities, the estimation problem becomes a convex optimization problem, see [Schön *et al.*, 2003]. In the case of Gaussian disturbances and quantized measurements, the estimation problem becomes a quadratic program, suitable for e.g. moving horizon estimation [Rawlings and Bakshi, 2006]. Paper V starts from a JMAP formulation, which is simplified to the point that it can be used for real time control of a moving cart with quantized position measurements, using a low-end microcontroller.

Worst-case measurement disturbances. Instead of modelling the measurement function g, a possible relaxation is to specify an

upper bound on the error $|Cx-y|$. This leads to a worst-case analysis, or a mixed worst-case/stochastic analysis if stochastic disturbances are still kept in the model. Estimation with worst-case process and measurement disturbances is treated in [Bertsekas and Rhodes, Apr 1971; Durieu *et al.*, 2001]. Different approaches to the mixed estimation problem are explored in [Hanebeck and Horn, 2001] and in Paper IV. [Morrell and Stirling, 1988] treats the mixed case when only the initial conditions have a worst case component.

3.4 Approach of Papers III–V

Although the measurement situations discussed in this chapter can be treated with some success using linear methods as long as the measurement can be modeled with a bounded disturbance, the results can be very conservative. With coarse quantization, the disturbances fed into the feedback loop will either cause excessive noise in the control signal or force the control loop bandwidth to decrease drastically. The measurement disturbance may also drive the system into limit cycles. The aim of Papers III–V is to find practical control strategies that can deal with this type of sensing constraints.

In order to focus on the essentials, the problem formulations are streamlined where possible:

- The recurring example process is a double integrator with noise entering along with the control signal. This is a common process to encounter in practice, exemplified by the moving cart studied in Paper V. Also, it is the simplest process where a naive LTI estimator can give arbitrarily large gain from measurement disturbance to state estimate. Finally, the low order allows to approximate the optimal Bayesian estimator with a grid filter, and allows simple representation and manipulation of uncertainty sets in the state space.

- At events, the measured value is considered to be known, or known up to a Gaussian disturbance.

- The problems are formulated in discrete time to avoid technical issues related to partial differential equations.

The progression goes from Paper III, which mainly considers the theoretical background, to Paper V, which is focused on a practical control implementation that is verified experimentally.

Paper III studies the Bayesian estimator, showing that it is in some sense well-behaved for e.g. the estimation problems considered in this chapter. For instance, the conditional probability density of the state is log-concave, which implies, among other things, that it essentially has a global maximum, an indication that a simple estimator may be useful in practice. A heuristically designed Kalman filter is shown to work well in the example.

Paper IV proposes an estimator that is implementable for reasonably high order processes, reducing the dimension of the estimation problem from infinite to a modest finite number. This is achieved by incorporating elements of worst-case analysis and ellipsoidal over-approximation of uncertainty sets. The estimator is shown to come quite close to the performance of the grid filter approximated Bayesian estimator in a double integrator simulation example.

Paper V finally derives an event-based state estimator so simple that it can be used in real-time on a low-end microcontroller. Insights from LTI control design are used as a guide to maintain robustness in the control loop. The controller is applied to control a moving cart with quantized position measurements and is shown in experiments to have a dramatically higher tolerance for quantization than a similar linear controller.

4

Outlook

There is a wealth of possible directions to continue the work presented in this thesis. Some interesting options are:

- Find new problem formulations:

  - Which are the control problems where event-based control has the greatest impact?

  - Which relevant problem formulations are easiest to solve?

- Include the effect of packet losses in the network model. This applies both to communication constrained control and send-on-delta measurements.

- Investigate the robustness to modelling error between event-based and linear strategies.

- Search for more efficient computational methods and better approximations; this would allow to treat higher order systems in a better way.

For the communication constrained case:

- Extend the analysis to higher order systems and systems without impulse control.

- Investigate if the controller can adapt to changing process noise intensity.

- Investigate the coupling between the state feedback design and event-triggering threshold design.

- Search for a systematic means to optimize the controller parameters of multiple loops sharing the same network.

For the sensing constrained case:

- Analyze stability/robustness of the closed loop using state feedback from e.g. simplified JMAP estimators.

- Investigate if particle filtering can be fruitful for these problems.

References

An, M. (1996): "Log-concave probability distributions : Theory and statistical testing." Papers 96-01. Centre for Labour Market and Social Research, Denmark.

Arulampalam, S., S. Maskell, and N. Gordon (2002): "A tutorial on particle filters for online nonlinear/non-Gaussian Bayesian tracking." *IEEE Transactions on Signal Processing*, **50**, pp. 174–188.

Årzén, K.-E. (1999): "A simple event-based PID controller." In *Preprints 14th World Congress of IFAC*. Beijing, P.R. China.

Åström, K. J. (2007): "Event based control." In *Analysis and Design of Nonlinear Control Systems: In Honor of Alberto Isidori*. Springer Verlag.

Åström, K. J. and B. Bernhardsson (1999): "Comparison of periodic and event based sampling for first-order stochastic systems." In *Preprints 14th World Congress of IFAC*, vol. J, pp. 301–306. Beijing, P.R. China.

Åström, K. J. and B. Wittenmark (1997): *Computer-Controlled Systems*. Prentice Hall.

Bagnoli, M. and T. Bergstrom (1989): "Log-concave probability and its applications." Papers 89-23. Michigan - Center for Research on Economic & Social Theory.

Bertsekas, D. and I. Rhodes (Apr 1971): "Recursive state estimation for a set-membership description of uncertainty." *Automatic Control, IEEE Transactions on*, **16:2**, pp. 117–128.

References

Boyd, S., L. El Ghaoui, E. Feron, and V. Balakrishnan (1994): *Linear Matrix Inequalities in System and Control Theory*, vol. 15 of *Studies in Applied Mathematics*. SIAM, Philadelphia, PA. Also available at `http://www.stanford.edu/~boyd/lmibook/`.

Boyd, S. and L. Vandenberghe (2004): *Convex Optimization*. Cambridge University Press, New York, NY, USA.

Buttazzo, G. C. (1997): *Hard Real-Time Computing Systems: Predictable Scheduling Algorithms and Applications*. Kluwer Academic Publishers.

Cervin, A., D. Henriksson, B. Lincoln, J. Eker, and K.-E. Årzén (2003): "How does control timing affect performance?" *IEEE Control Systems Magazine*, **23:3**, pp. 16–30.

Cervin, A. and E. Johannesson (2008): "Sporadic control of scalar systems with delay, jitter and measurement noise." In *Proc. 17th IFAC World Congress*. Seoul, Korea.

Cogill, R., S. Lall, and J. P. Hespanha (2006): "A constant factor approximation algorithm for event-based sampling." In *Proc. Allerton Conference on Communication, Control, And Computing*.

Cox, H. (1964): "On the estimation of state variables and parameters for noisy dynamic systems." *IEEE Transactions on Automatic Control*, **9:1**, pp. 5–12.

Curry, R. E. (1970): *Estimation and Control with Quantized Measurements*. Research Monograph No. 60. M.I.T. Press.

Dodds, S. J. (1981): "Adaptive, high precision, satellite attitude control for microprocessor implementation." *Automatica*, **17:4**, pp. 563–573.

Durieu, C., E. Walter, and B. Polyak (2001): "Multi-input multi-output ellipsoidal state bounding." *Journal of Optimization Theory and Applications*, **111**, November, pp. 273–303(31).

Feldbaum, A. A. (1960–1961): "Dual control theory. I–IV." *Automation Remote Control*, **21, 22**, pp. 874–880, 1033–1039, 1–12, 109–121.

Georgiev, D. and D. M. Tilbury (2006): "Packet-based control: The H2-optimal solution." *Automatica*, **42:1**, pp. 137–144.

Hanebeck, U. and J. Horn (2001): "New estimators for mixed stochastic and set theoretic uncertainty models: the general case." *American Control Conference, 2001. Proceedings of the 2001*, **6**, pp. 5040–5045 vol.6.

Hristu-Varsakelis, D. and P. R. Kumar (2002): "Interrupt-based control over a shared communication medium." In *Proc. IEEE Conference on Decision and Control*.

Hu, J., J. Lygeros, and S. Sastry (2000): "Towards a theory of stochastic hybrid systems." In *Proc. Hybrid Systems: Computation and Control*.

Imer, O. C. and T. Basar (2005): "Optimal estimation with limited measurements." In *Proc. IEEE Conference on Decision and Control*.

Johannesson, E., T. Henningsson, and A. Cervin (2007): "Sporadic control of first-order linear stochastic systems." In *Proc. 10th International Conference on Hybrid Systems: Computation and Control*, Lecture Notes in Computer Science 4416. Springer-Verlag, Pisa, Italy.

Kalman, R. E. (1960): "A new approach to linear filtering and prediction problems." *Transactions of the ASME–Journal of Basic Engineering*, **82:Series D**, pp. 35–45.

Kalman, R. E. and R. S. Bucy (1961): "New results in linear filtering and prediction theory." *Transactions of the ASME - Journal of Basic Engineering*, **83**, pp. 95–107.

Lemmon, M. D., T. Chantem, X. S. Hu, and M. Zyskowski (2007): "On self-triggered full-information H-infinity controllers." In Bemporad *et al.*, Eds., *HSCC*, vol. 4416 of *Lecture Notes in Computer Science*, pp. 371–384. Springer.

Martins, N. and M. Dahleh (2008): "Feedback control in the presence of noisy channels: bode-like fundamental limitations of performance." *Automatic Control, IEEE Transactions on*, **53:7**, pp. 1604–1615.

Morrell, D. and W. Stirling (1988): "Set-valued filtering and smoothing." *Twenty-Second Asilomar Conference on Signals, Systems and Computers, 1988.*, pp. 236–240.

References

Otanez, P. G., J. R. Moyne, and D. M. Tilbury (2002): "Using deadbands to reduce communication in networked control systems." In *Proc. American Control Conference*.

Prékopa, A. (1971): "Logarithmic concave measures with application to stochastic programming." *Acta Sci. Math. (Szeged)*, **32**, pp. 301–316.

Prékopa, A. (1973): "On logarithmic concave measures and functions." *Acta Sci. math. 34, 335-343.*

Rabi, M. (2006): *Packet based Inference and Control*. PhD thesis, Institute for Systems Research, University of Maryland. URL: `http://hdl.handle.net/1903/3970`.

Rabi, M., K. H. Johansson, and M. Johansson (2008): "Optimal stopping for event-triggered sensing and actuation." In *Proc. 47th IEEE Conference on Decision and Control*. Cancún, Mexico. To appear.

Rawlings, J. and B. Bakshi (2006): "Particle filtering and moving horizon estimation." *Computers and Chemical Engineering*, **30:10-12**, pp. 1529–1541.

Rehbinder, H. and M. Sanfridson (2004): "Scheduling of a limited communication channel for optimal control." *Automatica*, **30:3**, pp. 491–500.

Sandee, J. H. (2006): *Event-driven control in theory and practice — trade-offs in software and control performance.* PhD thesis, Eindhoven Univ. of Techn., Netherlands.

Sandee, J. H., W. P. M. H. Heemels, and P. P. J. van den Bosch (2007): "Case studies in event-driven control." In *Proc. Hybrid Systems: Computation and Control*, pp. 762–765.

Schön, T., F. Gustafsson, and A. Hansson (2003): "A note on state estimation as a convex optimization problem." *Acoustics, Speech, and Signal Processing, 2003. Proceedings. (ICASSP '03). 2003 IEEE International Conference on*, **6**, April, pp. VI–61–4 vol.6.

Tabuada, P. (2007): "Event-triggered real-time scheduling of stabilizing control tasks." *IEEE Transactions on Automatic Control*, **52:9**, pp. 1680–1685.

Vasyutynskyy, V. and K. Kabitzsch (2007): "Simple PID control algorithm adapted to deadband sampling." In *Proc. 12th IEEE Conference on Emerging Technologies and Factory Automation (ETFA'07)*, pp. 932–940.

Wang, X. and M. Lemmon (2008): "Event-triggered broadcasting across distributed networked control systems." In *Proc. American Control Conference*, pp. 3139–3144.

Witsenhausen, H. S. (1968): "A counterexample in stochastic optimum control." *SIAM Journal on Control*, **6:1**, pp. 131–147.

Yook, J., D. Tilbury, and N. Soparkar (2002): "Trading computation for bandwidth: reducing communication in distributed control systems using state estimators." *Control Systems Technology, IEEE Transactions on*, **10:4**, pp. 503–518.

Paper I

Sporadic Event-Based Control of First-Order Linear Stochastic Systems

**Toivo Henningsson   Erik Johannesson
Anton Cervin**

Abstract

The standard approach in computer-controlled systems is to sample and control periodically. In certain applications, such as networked control systems or energy-constrained systems, it could be advantageous to instead use event-based control schemes. Aperiodic event-based control of first-order stochastic systems has been investigated in previous work. In any real implementation, however, it is necessary to have a well-defined minimum inter-event time. In this paper, we explore two such sporadic control schemes for first-order linear stochastic systems and compare the achievable performance to both periodic and aperiodic control. The results show that sporadic control can give better performance than periodic control in terms of both reduced process state variance and reduced control action frequency.

1. Introduction

Digital feedback controllers are most often implemented using periodic sampling, computation, and actuation. This approach enables the control designer to utilize standard sampled-data system theory or to discretize a continuous-time controller assuming a fixed sampling rate and constant hold intervals [Åström and Wittenmark, 1997].

For some applications, however, event-based control schemes may have an advantage over periodic schemes. In networked control applications, it could make sense to only transmit information when something significant has occurred in the system, in order to save bandwidth. In embedded applications, it may be essential to minimize the number of control actions in order to save energy. In the application of inventory control it seems rational to replenish stock only when it is low rather than on a periodic basis, if there is a fixed transportation cost. Some sensors such as rotary motion encoders only give new measurements at ahead-of-time unknown events.

Event-based control as a technology is of course not new. Mostly, however, it has been applied in an ad-hoc way. This can be attributed to the lack of a comprehensive theory, which in turn can be explained by the mathematical difficulties involved. A discrete-time formulation can sometimes make it slightly easier to obtain a solution. Some recent papers have thus solved optimal discrete-time estimation problems, with limited [Imer and Basar, 2005] or event-triggered [Cogill *et al.*, 2006] measurements.

From a control-theoretic point of view, event-based control systems can be viewed as hybrid systems. In this paper, we consider first-order linear stochastic systems, where an exogenous random disturbance (modelled as white noise) causes the process state to drift. The control law generates discrete events when the state crosses certain boundaries. Hence, our system falls into the category of stochastic hybrid systems as defined in [Hu *et al.*, 2000].

Event-based control of first-order linear stochastic systems was studied in [Åström and Bernhardsson, 1999]. It was shown that, compared to periodic control, the output variance could be significantly reduced assuming the same mean time between events. The control was realized by applying an impulse action whenever the magnitude of the system state exceeded a certain threshold. This work was elaborated in

[Rabi, 2006], which explores, among other things, event-based control with piecewise constant control signals and level-triggered sampling.

From a real-time systems point of view, however, tasks triggered by asynchronously generated events cannot be guaranteed service unless there is a well-defined minimum inter-arrival time. For the controller presented in [Åström and Bernhardsson, 1999] there was no such minimum inter-arrival time. In accordance with real-time systems terminology [Buttazzo, 1997], we will refer to such a control policy as *aperiodic*.

In this paper, we explore the class of *sporadic* event-based controllers for first-order linear stochastic systems. With a minimum inter-arrival time T between events, such a controller can be guaranteed not to consume more than a certain network bandwidth or CPU utilization. Two sporadic controllers will be studied. The first controller measures the process state continuously and can take control actions at any time, but no more often than every T seconds. The second controller measures the process state every T_s seconds until a control action is applied, and resumes measurements T seconds after the last control action.

2. Problem Formulation

The process to be controlled is given by the linear stochastic differential equation

$$dx = axdt + udt + \sigma dw, \qquad x(0) = 0, \tag{1}$$

where x is the state, u the control signal, w is a Wiener process with unit incremental variance, a is the pole of the process, and $\sigma > 0$ is the intensity of the process noise. The control signal is zero except at *events t_k*, when it is allowed to be a Dirac pulse of magnitude u_k:

$$u(t) = \sum_{k=0}^{\infty} \delta(t - t_k)u_k. \tag{2}$$

The controller chooses when to generate an event based on the state of the system. After each event there is a period of *inactive state* of duration T, when no new events can be generated, see Fig. 1.
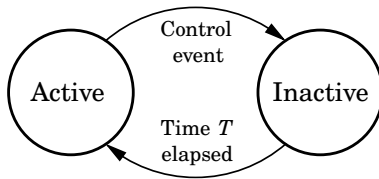
Figure 1.   Controller state transitions. Control events may only be generated in the active state.

The performance is measured by the stationary state cost,

$$J_x = \limsup_{t \to \infty} \mathrm{E}\, \frac{1}{t} \int_0^t x^2 ds,$$

and by the average control rate (or control cost),

$$J_u = \limsup_{t \to \infty} \mathrm{E}\, \frac{1}{t} N_u(0, t),$$

where $N_u(t_1, t_2)$ is the number of control actions in the interval (t_1, t_2). The total cost to be minimized is

$$J = J_x + \rho J_u, \tag{3}$$

where $\rho \geq 0$ is the relative cost of control actions.

Normalized Formulation

To reduce the number of free parameters we can use coordinate scaling to fix $\sigma = T = 1$. The parameters that remain, a and ρ, suffice to specify the problem up to coordinate scaling, and the original variables can be retrieved from inverse scaling. The parameters σ and T will be kept in the presentation when they add insight.

Let the transformed variables be described by

$$dt = T d\tau, \qquad dw = \sqrt{T} dv, \qquad x = \sigma \sqrt{T} x'$$

The dynamics become

$$dx' = a' x' d\tau + u' d\tau + dv,$$

where $u' = \sqrt{T}\sigma^{-1}u$, and $a' = aT$ is the relevant measure of process speed. The original costs are retrieved as

$$J_x = \sigma^2 T J'_x, \qquad J_u = T^{-1}J'_u,$$

so $\rho' = \frac{\rho}{\sigma^2 T^2}$ is the proper weighting after normalization. The normalized problem is described by the parameters

$$a' = aT, \qquad \rho' = \frac{\rho}{\sigma^2 T^2}.$$

3. Sporadic Control

3.1 General Observations

A sporadic controller is defined by two properties: when it generates an event and what control signal is used at the event. It is easy to see that an optimal controller for the problem above must satisfy the following:

- At any event t_k, the control signal u_k is chosen to bring x to the origin, i.e. $u_k = -x(t_k - 0)$.

- When an event is permitted, the decision of whether to generate one is a function only of the state x, and due to symmetry only of the absolute value $|x|$.

- If an event should be generated when $|x| = r$, one should also be generated whenever $|x| \geq r$.

Thus, the optimal control policy is a threshold policy where an event is triggered to bring x to zero whenever permitted and $|x| \geq r$.

To find the optimal threshold r, we evaluate J as a function of r in the closed loop system. To facilitate this, we first consider what happens between events.

3.2 Evolution Between Events

Between events the control signal is known, and the system evolves as a linear stochastic process. Assume that an event occurs at time

$t_k = 0$, and that we want to predict the evolution from that time, from the state prior to the event $x_0 = x(0-)$. Let

$$m(t) = \mathrm{E}\big(x(t)\big), \quad P(t) = \mathrm{E}\big(x(t)^2\big) - m(t)^2$$

be the expected state trajectory and the expected state variance due to process noise entering after the event respectively, with initial conditions

$$m(0) = x_0 + u_k, \qquad P(0) = 0.$$

The distribution of $x(t)$ will be Gaussian with mean $m(t)$ and variance $P(t)$.

The expected state cost during the interval $(0, t)$ can be expressed as the sum of one contribution $V_P(t)$ from P and one $V_m(t)$ from m according to

$$\int_0^t \mathrm{E}\big(x(s)^2\big)ds = \int_0^t \big(P(\tau) + m(\tau)^2\big)d\tau = V_P(t) + V_m(t).$$

Since there is no feedback between events, u will enter the evolution only through $m(t)$. We find that

$$\begin{aligned}
\mathrm{E}(dP) &= \mathrm{E}(2xdx + dx^2) - 2m\,\mathrm{E}(dm) \\
&= \mathrm{E}(2x(axdt + udt) + \sigma^2 dw^2) - 2m(am + u) \\
&= (2aP + \sigma^2)dt.
\end{aligned}$$

Starting from $P(0) = 0$, the solution is

$$P(t) = \begin{cases} (1 - e^{2at})\dfrac{\sigma^2}{-2a}, & a \neq 0, \\[2mm] \sigma^2 t, & a = 0. \end{cases} \tag{4}$$

Integrating, the process noise contribution to the state cost during the interval $(0, t)$ is

$$V_P(t) = \begin{cases} \dfrac{\sigma^2}{-2a}\left(t - \dfrac{e^{2at} - 1}{2a}\right), & a \neq 0, \\[3mm] \dfrac{1}{2}\sigma^2 t^2, & a = 0. \end{cases} \tag{5}$$

The expected trajectory evolves according to $\mathrm{E}(dm) = \mathrm{E}(dx) = (am + u)dt$, giving the prediction

$$m(t) = e^{at}m(0) + \int_{0+}^{t} e^{a(t-\tau)}u(\tau)d\tau.$$

With no control during the interval $(0, t)$, the cost is

$$V_m(t) = \int_0^t m(s)^2 ds = Q(t)m(0)^2,$$

$$Q(t) = \begin{cases} \dfrac{e^{2at} - 1}{2a}, & a \neq 0, \\[2mm] t, & a = 0. \end{cases} \tag{6}$$

3.3 Sporadic Control with Continuous Measurements

We assume that the process state is measured continuously in the active state. As soon as the state leaves the region $|x| < r$, an event is generated and the controller is put in the inactive state for an interval of length T.

Since the system is reset to the same state at each event, the expected cost and time from one event to the next are enough to find the stationary costs, as

$$J_x = \frac{V_{\text{active}} + V_{\text{inactive}}}{T_{\text{active}} + T}, \qquad J_u = \frac{1}{T_{\text{active}} + T},$$

where V_{active} and T_{active} are the expected state costs and dwell times during one period of active state, and $V_{\text{inactive}} = V_P(T)$. We will characterize the behavior between two events by modifying the system so that it starts at one event and is stopped at the next.

The expected cost and dwell time during one period of active state can be found as

$$V_{\text{active}} = \int x^2 F(x)dx, \qquad T_{\text{active}} = \int F(x)dx, \tag{7}$$

where $F(x) = \int_0^\infty f(x,t)dt$ is the accumulated state density of the density $f(x,t)$ in the active state.

The system enters the active state as

$$f(x, t = 0) = \begin{cases} \varphi(x), & |x| < r \\ 0, & |x| \geq r \end{cases}$$

where $\varphi(x)$ is Gaussian with zero mean and variance $P(T)$. The time evolution is given by the Fokker-Planck equation (see e.g. [Åström, 1970], [Feller, 1971]) (with $\sigma = 1$):

$$\frac{\partial f}{\partial t} = \frac{1}{2}\frac{\partial^2}{\partial x^2}(f\sigma^2) - \frac{\partial}{\partial x}(fax) = \frac{1}{2}\frac{\partial^2 f}{\partial x^2} - ax\frac{\partial f}{\partial x} - af,$$

with absorbing boundary conditions $f(\pm r, t) = 0$. Since $f(x, t) \to 0$ as $t \to \infty$ we can integrate over $t \in [0, \infty)$ to find a differential equation for $F(x)$:

$$-\varphi(x) = \int_0^\infty \frac{\partial f}{\partial t}dt = \frac{1}{2}F''(x) - axF'(x) - aF(x),$$

with boundary conditions $F(\pm r) = 0$. The solution exists as long as $\varphi(x)$ does, and can be found numerically with a linear ODE Boundary Value Problem (BVP) solver or analytically as

$$F(x) = 2\int_{y=-r}^x e^{a(x^2 - y^2)}\int_{z=y}^0 \varphi(z)dzdy, \qquad |x| \leq r. \tag{8}$$

Fig. 2 shows the costs as a function of r for the case $T = \sigma = 1$ and $a \in \{-0.5, 0, 0.5\}$. Other cases can be reconstructed by scaling as explained in Section 2. We see an initial decrease in the state cost as the threshold is increased, so the optimal threshold is non-zero even when $\rho = 0$. We also see that both costs decrease as a decreases, since the system becomes easier to control.

The cost functions can alternatively be found from

$$V_{\text{active}} = \int \varphi(x)V_x(x)dx, \quad T_{\text{active}} = \int \varphi(x)\theta(x)dx, \tag{9}$$

where $V_x(x)$ is the expected state cost until the next event starting in the active state at x, and $\theta(x)$ is the corresponding expected dwell

time (or first passage time, (see [Feller, 1971]). The value function $V(x) = V_x(x) - J\theta(x)$ can be used for dynamic programming.

When $x = \pm r$, $V_x(x) = \theta(x) = 0$, and when $|x| < r$,

$$\mathrm{E}\big(dV_x(x)\big) = -x^2 dt, \quad \mathrm{E}\big(d\theta(x)\big) = -1 dt$$

which together with the dynamics (1) gives

$$-x^2 dt = \mathrm{E}\left(V_x' dx + \frac{1}{2} V_x'' dx^2\right) = \left(ax V_x' + \frac{1}{2} V''\right) dt,$$

for $V_x(x)$, and similarly for $\theta(x)$. The solutions can be found numerically with an ODE BVP solver, or as

$$\begin{pmatrix} V_x(x) \\ \theta(x) \end{pmatrix} = 2 \int_{y=x}^{r} \int_{z=0}^{y} e^{a(z^2 - y^2)} \begin{pmatrix} z^2 \\ 1 \end{pmatrix} dz\,dy. \tag{10}$$

We note that problem can be extended in a few ways that fit well with our solution methods. Behavior in the inactive state only affects the solution through T_{inactive}, V_{inactive}, and the state density when entering the active state $\varphi(x)$. Possible extensions include a delay $\tau \leq T$ from the issue of an event to the actuation of the control impulse, and a stochastically varying inactive time T.

3.4 Sporadic Control with Discrete Measurements

We now assume that the process is sampled with the interval $T_s \leq T$ in the active state. Any deviations of the state outside the threshold between samples will go unnoticed. As before, when a deviation is detected at time t_k, the controller issues a control event and enters the inactive state, where it stays for T seconds. We now let $\{t_k\}$ denote all sampling instants, which progress as

$$t_{k+1} = \begin{cases} t_k + T_s, & |x_k| < r, \\ t_k + T, & |x_k| \geq r. \end{cases}$$

To find the optimal threshold r, the cost is characterized as a function of r. To this end, we compute the stationary state distribution (see
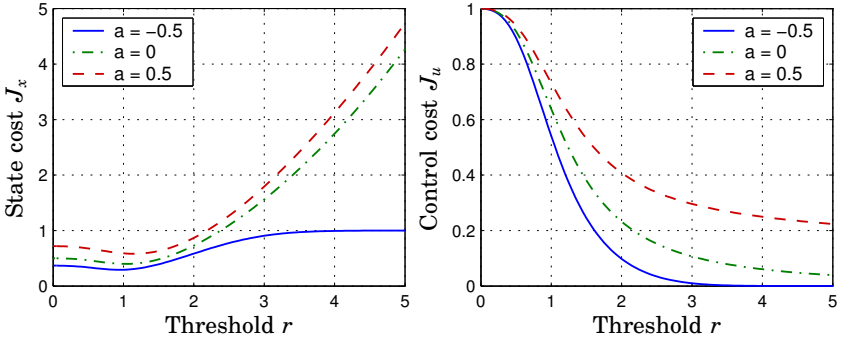
Figure 2. Cost functions for sporadic control with continuous-time measurements assuming $\sigma = T = 1$. Top: State cost J_x as a function of threshold r. Bottom: Control cost J_u as a function of threshold r. Both functions are plotted for systems with $a = -0.5$, $a = 0$ and $a = 0.5$ respectively.
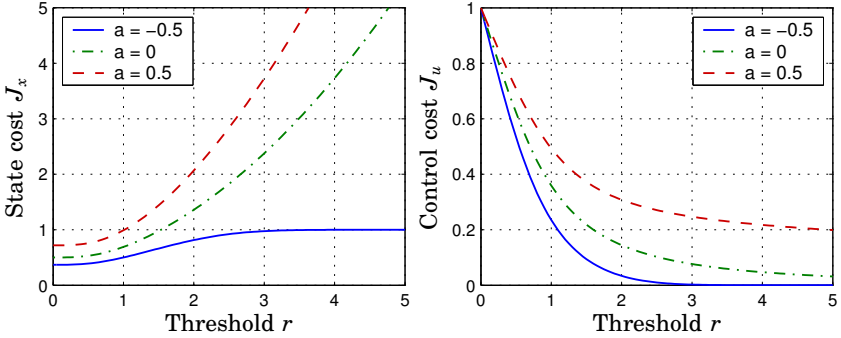
Figure 3. Cost functions for sporadic control with discrete-time measurements assuming $\sigma = T = T_s = 1$. Top: State cost J_x as a function of threshold r. Bottom: Control cost J_u as a function of threshold r. Both functions are plotted for systems with $a = -0.5$, $a = 0$ and $a = 0.5$ respectively.

[Feller, 1971]) at the sampling instants. Between sampling instants, the state evolves as

$$x_{k+1} = \begin{cases} e^{aT_s}x_k + w_k(T_s), & |x_k| < r, \\ w_k(T), & |x_k| \geq r, \end{cases} \tag{11}$$

where $w_k(t)$ is a Gaussian random variable with zero mean and variance $P(t)$. The stationary density always exists since there is a positive probability to go from any state x to any state interval (x_1, x_2) in one step, and for $|x| \geq r$ the density after any time step falls off as a Gaussian with variance $P(T)$. The accumulated state cost from time t_k to time t_{k+1} is given by

$$V_{\text{stay}} = Q(T_s) \, \text{E} \left\{ x_k^2 \mid |x_k| < r \right\} + V_P(T_s) \tag{12}$$

if the controller stays in the active state and by

$$V_{\text{exit}} = V_P(T) \tag{13}$$

if the controller enters the inactive state.

Finally, assuming stationarity, the costs become

$$J_x = \frac{p_{\text{stay}} V_{\text{stay}} + p_{\text{exit}} V_{\text{exit}}}{p_{\text{stay}} T_s + p_{\text{exit}} T}, \quad J_u = \frac{p_{\text{exit}}}{p_{\text{stay}} T_s + p_{\text{exit}} T}$$

where

$$p_{\text{stay}} = \text{Prob} \left\{ |x_k| < r \right\} = 1 - p_{\text{exit}}.$$

The stationary distribution of x_k can be found numerically by discretizing the state space and then iterating the distribution according to (11) until convergence.

Fig. 3 shows the costs as a function of r for the case $T = T_s = \sigma = 1$ and $a \in \{-0.5, 0, 0.5\}$. Here, the state cost increases monotonically with r. With $T_s < T$ we would have an initial decrease, approaching the behavior for continous measurements as $T_s/T \to 0$. The control action frequency J_u falls off faster with increasing threshold than for the continous measurement case, since x is checked against the threshold less often with discrete measurements. As expected, both costs decrease with a.

Alternatively, the expected state cost $V_x(x)$ and dwell time $\theta(x)$ until the next event starting from state x can be iterated until convergence. As in the continuous case, we could extend the problem formulation with actuation delay and stochastically varying inactive time.

4. Comparison of Control Schemes

Sporadic control with continuous and discrete measurements (with $T_s = T$) will now be compared to periodic and aperiodic control. We first discuss how to make the comparison.

4.1 Periodic and Aperiodic Control

An aperiodic controller sets the process state x to zero whenever $|x| \geq r$ using an impulse control action [Åström and Bernhardsson, 1999]. The cost functions can be found by letting $\varphi(x)$ approach a unity Dirac pulse in (8) or (9), yielding

$$J_x = V_{\text{active}}/T_{\text{active}}, \qquad J_u = T_{\text{active}}^{-1}.$$

We assume that periodic control is also implemented with impulse control action, such that x is periodically reset to zero. The sampling interval is restricted to be no shorter than for the sporadic schemes. The costs become

$$J_x = V_P(T)/T, \qquad J_u = T^{-1}. \tag{14}$$

4.2 Preliminaries

For the sporadic controllers, minimization of the loss function J for a given ρ determines an optimal threshold r, which maps to an optimal average event rate J_u. The same holds for aperiodic control. In periodic control, however, there is no threshold. Instead, ρ determines the optimal sampling interval. Hence, we can parametrize controllers from all four classes by average event rate.

   The four controllers differ by the constraints on when they can generate control events. A scheme with fewer restrictions will be harder to implement but give a lower cost J. As $p_{\text{inactive}} = J_u T \to 0$ and events become rare, sporadic control should approach aperiodic since T becomes negligible. When $\rho \to 0$, sporadic control with discrete measurements and $T_s = T$ will approach periodic since there remains no incentive to omit an event.

   When $a < 0$, J_x and therefore J is bounded by the variance achieved in open loop. As ρ increases, all controllers will generate fewer events so that $J_u \to 0$, and ultimately J_x will approach a maximum. The limit can be found from (14), where $J_x \to -1/2a$ as $T \to \infty$.
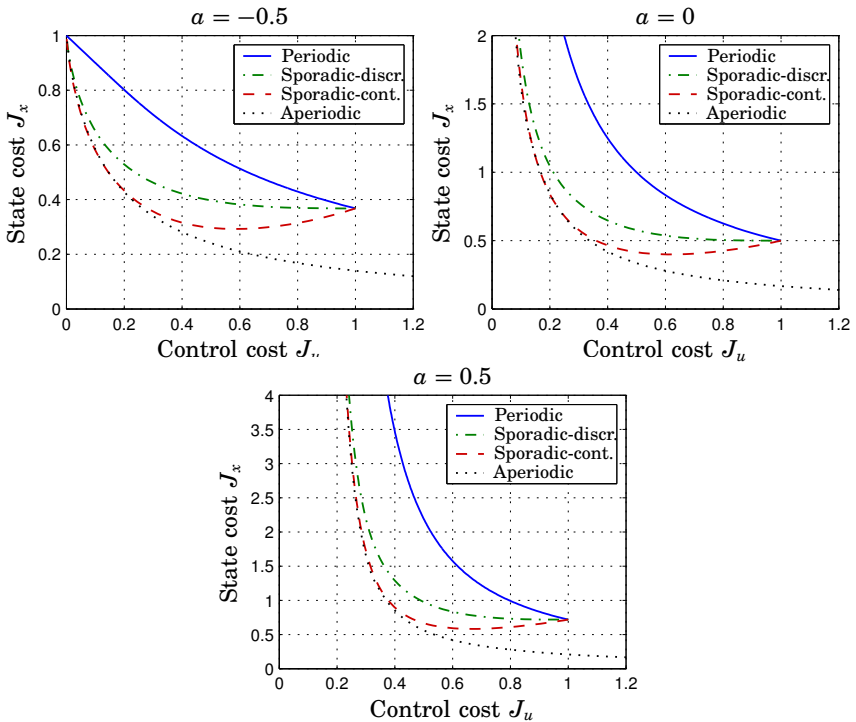
Figure 4. Trade-off between state cost J_x and control cost J_u for the four classes of controllers. Note the different vertical scales.

4.3 Comparison

The trade-off between state variance and average event frequency is made explicit in Fig. 4, where J_x is plotted against J_u for the four controllers. The results for $\sigma \neq 1$ are found by scaling J_x by σ^2. It is seen that the controllers are strictly ranked in performance by how much freedom they have to generate events, and that the sporadic controller with discrete measurements always outperforms the periodic one.

Fig. 4 also shows what we consider the main advantage of event-based control: fewer events are needed for the same state cost. With periodic control, the variance increases quite rapidly with lower sam-

pling rate. However, with sporadic control the average control rate can be reduced much further without the same penalty. For example, when $a = 0.5$ the average control rate may be decreased by about 40 % for only slightly more variance, using sporadic control with discrete measurements.

A notable result is that for sporadic control with continuous measurements, J_x can be made somewhat smaller *with fewer events*. This is also seen in the upper plot of Fig. 2, where J_x attains a minimum for $r > 0$. Apparently, there is a hidden cost in issuing a control event, due to the risk that large state errors will arise while in the inactive state. This phenomenon is absent for discrete measurements and $T_s = T$ since in this case events are generated independent of past actions.

Fig. 5 shows the optimal achievable cost J^* for the four controllers. It is notable that for the stable system $a = -0.5$ the optimal periodic controller chooses to never sample when $\rho \geq 1$, while the sporadic controllers just raise their thresholds and remain ready to deal with large disturbances.

5. Higher Order Systems

So far, we have only considered first order systems. When raising the state dimension, there are many different generalizations worthy of study, depending on which is the constraining resource that motivates using event based control. We will briefly discuss some possibilities.

5.1 Formulations

The dynamics and cost J_x are naturally extended to

$$dx = Ax dt + Bu dt + dw,$$

$$J_x = \limsup_{t \to \infty} \mathrm{E}\, \frac{1}{t} \int_0^t x^T Q x\, ds.$$

where now x, w and possibly u are vectors. One natural generalization of the measurement equation is

$$dy = Cx dt + dw_m,$$
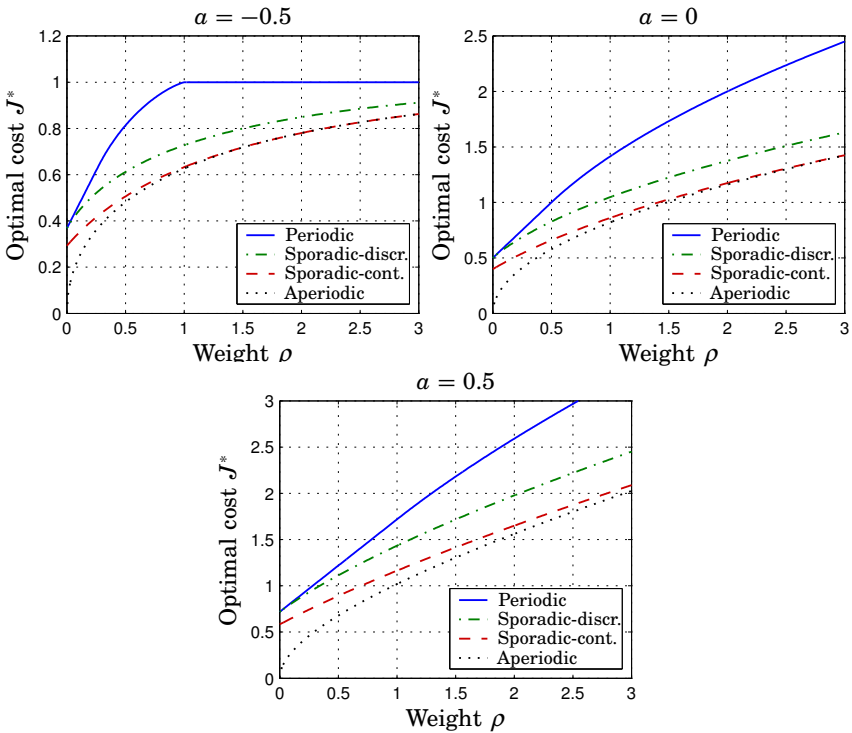
Figure 5. Optimal achievable cost J^* for the four classes of controllers. Note the different vertical scales.

where dw_m is measurement noise.

The possible forms of the controller, actuators, and sensors are more varied. Some scenarios are:

Communication constraints. Events are packets sent over a communication channel, from an observer at the sensor to a controller at the actuator. The observer decides when to send a state estimate to the controller, which predicts the plant state in open loop in between. Each event resets the prediction error.

Actuator constraints. The actuator only generates pulses of certain shapes, with some cost per pulse. The controller plans for an

optimal and possibly long sequence of pulses, which is sensitive to timing.

Sensor constraints. The sensor only gives measurements under some conditions, e.g. at or beyond some thresholds. The control problem becomes a state estimation problem with nonstandard measurement information, for which the Kalman Filter is not optimal.

Processing power constraints. A simple control law is needed. The best bet is probably to postulate one and optimize over a few parameters.

We can consider a single control loop, or multiple loops sharing the same limited resource. The loops can be independent, or cooperate to control a single plant. It seems unreasonable that the controllers should know each others' state, especially with communication constraints.

5.2 Methods

The discretizations applied in this paper can be generalized to higher state dimensions, but become impractical beyond a few states due to the curse of dimensionality. Sometimes the dimension can be reduced somewhat; e.g. if the state is estimated with a stationary Kalman Filter, the distribution of the actual state is known conditioned on the estimate. Otherwise, nonlinear process dynamics come at a modest additional cost. Optimal stochastic control is in principle applicable to both the communication and actuation constrained scenarios.

Beyond a few states, simpler formulations are necessary for a solvable problem. This may include reducing the amount of uncertainty in the problem. A communication constrained problem easily becomes pointless with too little uncertainty, while an actuator constrained problem may still preserve its major features.

6. Conclusions

In some applications there is a cost related to the execution of a control signal, regardless of the magnitude of that signal. If that cost is

included in the performance objective of the controller, it will be meaningful to reduce the frequency of control actions. This may be accomplished with a periodic controller by lengthening the sampling interval. However, the penalty in terms of increased process state variance is significant. Trying to improve the tradeoff by not acting on small state errors naturally leads to the notion of event-based control.

In this paper, we have shown that sporadic control can provide a better tradeoff between control objectives as well as better overall control performance than periodic control, when there is a fixed cost of control actions. It is noted that the average frequency of control events can be reduced with only a small increase in state variance. Moreover, we show that sporadic control can actually reduce both the average frequency of control events and the state variance simultaneously. When the objective is to reduce the frequency of events as well as the state variance, the sporadic control schemes presented here even perform almost as well as aperiodic control, while respecting a prespecified shortest inter-event time.

Event-based control has an additional threshold parameter that should scale with the size of disturbances. If they are bigger than expected, the control approaches periodic control. If they are smaller, the theshold will act as a tolerable margin of error. Both responses are reasonable in the face of a mismatched disturbance intensity.

Obviously, to implement sporadic control where periodic control is currently used requires some changes. Unless the hardware supports continuous measurements, discrete measurements are an easier option and approach the continuous performance quite fast if one can measure more often than control. The change from periodic to sporadic control with the same measurement and control interval should require minimal modifications.

References

Åström, K. J. (1970): *Introduction to Stochastic Control Theory*. Academic Press, New York.

Åström, K. J. and B. Bernhardsson (1999): "Comparison of periodic and event based sampling for first-order stochastic systems." In

Preprints 14th World Congress of IFAC, vol. J, pp. 301–306. Beijing, P.R. China.

Åström, K. J. and B. Wittenmark (1997): *Computer-Controlled Systems*. Prentice Hall.

Buttazzo, G. C. (1997): *Hard Real-Time Computing Systems: Predictable Scheduling Algorithms and Applications*. Kluwer Academic Publishers.

Cogill, R., S. Lall, and J. P. Hespanha (2006): "A constant factor approximation algorithm for event-based sampling." In *Proc. Allerton Conference on Communication, Control, And Computing*.

Feller, W. (1971): *An Introduction to Probability Theory and Its Applications*, vol. 2. John Wiley & Sons.

Hu, J., J. Lygeros, and S. Sastry (2000): "Towards a theory of stochastic hybrid systems." In *Proc. Hybrid Systems: Computation and Control*.

Imer, O. C. and T. Basar (2005): "Optimal estimation with limited measurements." In *Proc. IEEE Conference on Decision and Control*.

Rabi, M. (2006): *Packet based Inference and Control*. PhD thesis, Institute for Systems Research, University of Maryland. URL: `http://hdl.handle.net/1903/3970`.

Paper II

Scheduling of Event-Triggered Controllers on a Shared Network

Anton Cervin   Toivo Henningsson

Abstract

We consider a system where a number of independent, time-triggered or event-triggered control loops are closed over a shared communication network. Each plant is described by a first-order linear stochastic system. In the event-triggered case, a sensor at each plant frequently samples the output but attempts to communicate only when the magnitude of the output is above a threshold. Once access to the network has been gained, the network is busy for T seconds (corresponding to the communication delay from sensor to actuator), after which the control action is applied to the plant. Using numerical methods, we compute the minimum-variance control performance under various common MAC-protocols, including TDMA, FDMA, and CSMA (with random, dynamic-priority, or static-priority access). The results show that event-triggered control under CSMA gives the best performance throughout.

1. Introduction

Networked feedback control systems are normally implemented using periodic sampling at the sensor nodes, combined with either time-triggered or event-triggered communication between the sensor, controller, and actuator nodes. Periodic sampling allows for standard sampled-data control theory (e.g. [Åström and Wittenmark, 1997]) to be used, although network-induced delay and jitter may limit the performance [Cervin *et al.*, 2003].

In recent work [Åström and Bernhardsson, 1999; Hristu-Varsakelis and Kumar, 2002; Rabi, 2006; Johannesson *et al.*, 2007], event-triggered sampling has been proposed as a means for more efficient resource usage in networked control. The basic idea is to sample, communicate, and control only when something significant has occurred in the system. For first-order stochastic systems, it has been shown that event-based sampling can significantly reduce the output variance and/or the average control rate compared to periodic sampling [Åström and Bernhardsson, 1999]. A similar idea is to introduce a deadband in the sensor. The trade-off between network traffic and control performance for higher-order control loops with deadband sampling was studied in [Otanez *et al.*, 2002].

When multiple control loops are closed over a shared medium (like a communication bus or a wireless local-area network), a multiple access method such as TDMA (time division multiple access), FDMA (frequency division multiple access), or CSMA (carrier sense multiple access) is needed to multiplex the data streams. It is clear that the choice of access method can have a great impact on the control performance. Intuitively, TDMA should be suitable for time-triggered control loops, while CSMA, being a random-access method, would seem to be well suited for event-based control. FDMA provides a way to share the bandwidth without regard to synchronization among the loops, which could potentially be beneficial for both time-triggered and event-triggered control. At the same time, less bandwidth per control loop means longer transmission times and hence longer feedback delays.

Multi-loop networked control systems—taking into account issues such as clock synchronization, medium access, communication protocols, imperfect transmissions, delay and jitter, and event-triggered sampling, as well as the control algorithms themselves—are very com-
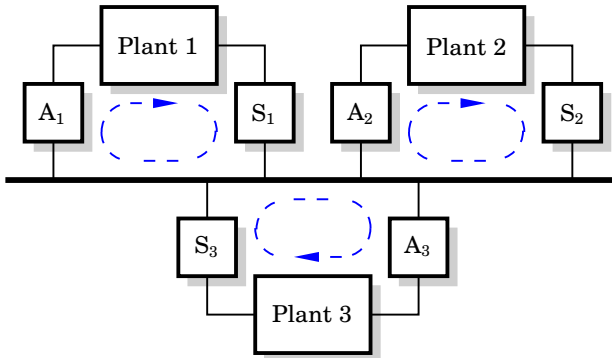
Figure 1. Multiple control loops are closed over a shared communication medium. The controller in each loop may be co-located with either the sensor (S) or the actuator (A).

plex systems. To facilitate analysis, great simplifications are needed. In this paper, we study a scenario where a number of independent control loops are closed over a shared network (see Fig. 1). Using very simple models for the plants, controllers, and network arbitration, we are able to numerically compute and compare the minimum-variance control performance under the various medium access protocols. In particular, we apply recent results in sporadic event-based control of first-order systems [Johannesson *et al.*, 2007; Cervin and Johannesson, 2008] to model and analyze the interaction between control loops and medium-access schemes. Although far from an exhaustive study, the results offer some interesting insight into the suitability of the studied MAC-protocols for networked control.

The remainder of this paper is outlined as follows. In Section II, the system description is given. Section III reviews how to calculate the stationary variance under time-triggered and event-triggered sampling. In Section IV, we model the medium access schemes and describe the co-design problem associated with each scheme. Section V reports numerical results for symmetrical integrator plants. In Section VI, we digress and compare the achievable performance under global vs local scheduling decisions. Section VII contains a case study with three asymmetric plants. Finally, the conclusions are given in Section VIII.

2. System Description

We consider a system where N control loops are closed over a shared network. Each plant $i \in 1 \ldots N$ is described by a first-order stochastic differential equation

$$dx_i(t) = a_i x_i(t)dt + u_i(t)dt + \sigma_i dw_i(t), \quad x_i(0) = 0, \qquad (1)$$

where x_i is the state, a_i is the process pole, u_i is the control signal, w_i is a Wiener process with unit incremental variance, and $\sigma_i > 0$ is the intensity of the noise. All noise processes are assumed independent.

A sensor located at each plant i takes samples of the plant state at certain discrete time instants $\{t_{i,k}\}_{k=0}^{\infty}$:

$$x_{i,k} = x_i(t_{i,k}). \qquad (2)$$

The sampling can be either time-triggered or event-triggered, depending on the medium access scheme. After obtaining a sample, the sensor tries to initiate a control event by transmitting the value to the actuator. The network is however a shared resource that only one control loop may access at a time[1]. If two or more sensors attempt to transmit at the exact same time, a resolution mechanism determines who will gain access to the network. (The other nodes will simply discard their samples.) Once access has been gained, the network stays occupied for T seconds, corresponding to the transmission delay from sensor to actuator. During this interval, no new control events may be generated (see Fig. 2).

The controller in each loop may be co-located with either the sensor or the actuator; the network delay is assumed constant and known, so it does not matter which. The overall goal is to minimize the total cost

$$J = \sum_{i=1}^{N} J_i, \qquad (3)$$

where the performance of loop i is measured by the stationary state variance

$$J_i = \lim_{t \to \infty} \frac{1}{t} \, \mathrm{E} \int_0^t (x_i(s))^2 ds. \qquad (4)$$

---

[1]This is not true under FDMA. Under FDMA, we rather assume that each control loop has access to its own private network with lower bandwidth.
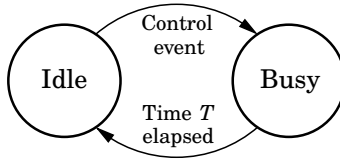
Figure 2. Network state transitions. Control events may only be generated in the idle state.

In response to a sample taken at time $t_{i,k}$, the actuator is allowed to emit a Dirac pulse of size $u_{i,k}$. It is clear (see [Cervin and Johannesson, 2008]) that minimum variance is achieved by driving the expected value of the state at time $t_{i,k} + T$ to zero, implying the deadbeat control law

$$u_{i,k} = -e^{a_i T} x_{i,k}. \tag{5}$$

The control signal generated by actuator i is hence given by the pulse train

$$u_i(t) = \sum_{k=0}^{\infty} \delta(t - t_{i,k} - T) u_{i,k}. \tag{6}$$

While it may seem unrealistic to allow Dirac controls, it allows for a fair and straightforward comparison between time-triggered and event-triggered control. The Dirac pulse may be replaced by an arbitrary pulse shape of length no longer than T at the expense of slightly more complicated cost calculations.

3. Evaluation of Cost

We here briefly review how to compute the cost (4) under time-triggered and event-triggered sampling with a delay and minimum inter-event interval T. For more details, see [Åström, 1970; Johannesson *et al.*, 2007; Cervin and Johannesson, 2008]. For clarity, we here drop the plant index i.

3.1 Time-Triggered Sampling

Under time-triggered sampling, the stationary variance (4) can be calculated analytically. The sampling instants t_k are known a-priori and

do not depend on the plant state, which will be normal distributed at all times. The (possibly irregularly) sampled closed-loop system becomes

$$x_{k+1} = w_k, \tag{7}$$

where $\{w_k\}_{k=0}^{\infty}$ are independent, zero-mean Gaussian variables with variance $P(t_{k+1} - t_k)$, where

$$P(t) = \begin{cases} \sigma^2 \dfrac{e^{2at} - 1}{2a}, & a \neq 0, \\ \sigma^2 t, & a = 0. \end{cases} \tag{8}$$

(Note that the delay does not affect the state distribution at the sampling instants.) Sampling the cost function gives

$$\mathrm{E} \int_{t_k}^{t_{k+1}} x^2 ds = Q(T)\,\mathrm{E}(x_k)^2 + J_v(t_{k+1} - t_k), \tag{9}$$

where

$$Q(T) = \begin{cases} \frac{e^{2aT}-1}{2a}, & a \neq 0, \\ T, & a = 0 \end{cases} \tag{10}$$

is the state weight due to delay, while

$$J_v(t) = \begin{cases} \frac{e^{2at}-2at-1}{4a^2}, & a \neq 0, \\ \frac{t^2}{2}, & a = 0 \end{cases} \tag{11}$$

accounts for the inter-sample noise (see e.g. [Åström, 1970]). Finally, we know that $\mathrm{E}\, x^2(t_k) = P(t_k - t_{k-1})$. Using the expressions above, it is straightforward to evaluate the cost under any static cyclic schedule.

3.2 Event-Triggered Sampling

Under event-triggered sampling, control events may only be generated when the network is idle and $|x(t)| \geq r$, where r is the event detection threshold. The state will no longer be Gaussian, which complicates the calculation of $\mathrm{E}\, x^2(t_k)$. A useful and realistic approximation is to assume that the sensor does not measure x continuously, but rather

uses fast sampling with the interval $T_s \ll T$. The (irregularly) sampled closed-loop system then becomes

$$
x_{k+1} = \begin{cases} e^{aT_s}x_k + w_k(T_s), & |x_k| < r \\ w_k(T), & |x_k| \geq r \ \& \ \text{won} \\ e^{aT}x_k + w_k(T), & |x_k| \geq r \ \& \ \text{lost} \end{cases} \tag{12}
$$

where $\{w_k(t)\}_{k=0}^{\infty}$ is a sequence of independent, zero-mean Gaussian variables with variance $P(t)$; "won" means that the sensor node won the network arbitration, while "lost" means the opposite. Letting the system run in open loop between the fast samples, the expressions (8)–(11) for the sampled cost are still valid.

The update equation (12) is useful both for calculation of the state distribution and for Monte Carlo simulations. Because of the shared medium, the stationary probability distributions of $x_1, \ldots, x_N$ are not independent. To evaluate the cost using the first approach, it is hence necessary to find the multi-dimensional probability distribution $f(x_1, \ldots, x_N)$. This can in theory be done by gridding the state space and then iterating the distribution according to (12) until convergence. In practice, this can be done for a few dimensions, forcing us to rely on Monte Carlo simulations for $N \geq 3$ in this paper.

4. Medium Access Schemes and Control Policies

In this section, we present simple scheduling and control models for three medium access schemes and discuss how to derive optimal schedules and control policies.

4.1 TDMA (Time Division Multiple Access)

In TDMA (see Fig. 3), a cyclic access schedule is determined off-line. In each slot in the schedule, one control loop has access to the network for T seconds. Since there is no cost associated with using the network in our problem formulation, it is obvious that no slot should be left empty, and that the sensor should always sample and transmit in its slot. Hence, the optimal control scheme associated with TDMA will be a pure time-triggered scheme.
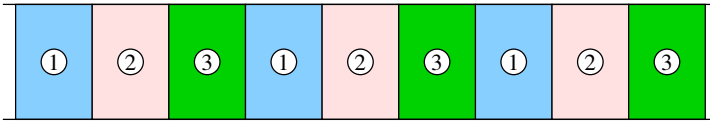
Figure 3. Time division multiple access (TDMA). A static cyclic schedule determines which sensor node samples and transmits in which time slot.
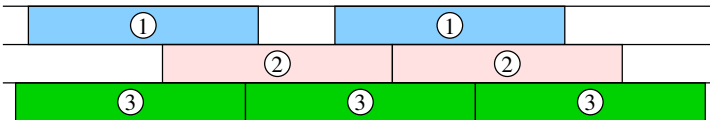

Figure 4. Frequency division multiple access (FDMA). The bandwidth is divided into fixed shares, giving each loop a dedicated channel. Within each share, an event-triggered control loop is implemented.

For symmetric plants (with $a_i = a$, $\sigma_i = \sigma$, $\forall i$), a simple round-robin schedule is optimal. For asymmetric plants, an optimal schedule of length n can be found by evaluating the resulting cost for each possible schedule. (The search for an optimal schedule can be done more efficiently. The LQ-optimal cyclic scheduling and control problem for multiple higher-order plants is treated in [Rehbinder and Sanfridson, 2004].)

4.2 FDMA (Frequency Division Multiple Access)

In FDMA (see Fig. 4), the communication bandwidth is divided between the nodes, such that each loop receives a fixed fraction U_i of the total capacity $\sum_{i=1}^{N} U_i = 1$. Accounting for the lower transmission rate, the delay from sensor i to actuator i is now T/U_i.

It is previously known [Johannesson *et al.*, 2007] that event-triggered sampling with a minimum inter-event interval T is superior to time-triggered sampling with the interval T, also when there is delay in the system. Hence, event-triggered control is the better choice for FDMA. The optimal event detection threshold and the associated optimal cost can be found numerically by sweeping r and computing the cost for each value.
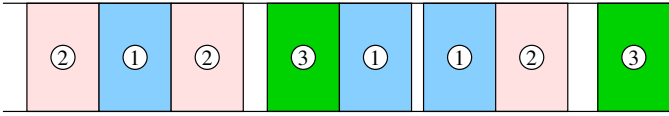
Figure 5. Carrier sense multiple access (CSMA). Each loop is event-triggered. A static, dynamic, or random priority function determines who will transmit if many nodes try to access the network at the same time.

For symmetric plants, an even division of the bandwidth is optimal. For asymmetric plants, the shares U_i can be found using optimization. Since the cost functions $J_i(U_i)$ are smooth and strictly decreasing, it is feasible to use standard nonlinear optimization tools to find the shares.

4.3 CSMA (Carrier Sense Multiple Access)

In CSMA (see Fig. 5), any node may try to access the network as soon as it becomes idle, making it suitable for event-triggered control loops. If many nodes want to transmit at the same time, some resolution mechanism must be used. In shared-medium Ethernet for instance, the collision detection and random back-off strategy will grant a random node access to the network (after some delay). In the Controller Area Network (CAN) on the other hand, access can be resolved based on either fixed (node) priorities or dynamic (message) priorities.

We will consider three different resolution mechanisms:

Random (CSMA-rand). As in Ethernet or WLAN, a random node will eventually win the contention. For simplicity, it is assumed that the resolution time is very small compared to the transmission time so that it can be neglected. The overall performance is optimized by selecting suitable event detection thresholds for the control loops. This is done by sweeping r_i and computing the cost for each value.

Static priority (CSMA-statprio). Each sensor node is assigned a static priority, which determines who will win the arbitration. Such a scheme can be useful for asymmetric plants where it is known that some plants are more sensitive to long access delays than others.

Dynamic Priority (CSMA-dynprio). For symmetric first-order plants, it can make sense to use the control error as a dynamic priority.

(This idea was put forth in [Walsh *et al.*, 1999], where it was called the Maximum-Error-First (MEF) scheduling technique.) It is assumed that the network interface provides a mechanism (such as message priorities in CAN) so that priority access can be given to the node with the largest control error. It is obvious that this scheme will be better than random priorities. Again, the overall performance is optimized by selecting event thresholds for the loops.

5. Results for Symmetric Integrator Plants

We here present numerical results for N symmetric integrator plants with $a_i = 0$ and $\sigma_i = 1$. We assume that the network bandwidth scales in proportion to the number of plants, such that the transmission delay from sensor to actuator is $T = 1/N$ when the full bandwidth is utilized. For the numerical computations, we assume fast sampling with $T_s = T/100$.

Under TDMA, the optimal cyclic transmission schedule is $\{1, 2, \ldots, N\}$. The sampling period of each loop is 1 and the delay is $T = 1/N$, giving the following exact value for the cost per loop:

$$J_i = \left(J_v(T) + Q(T) \, \mathrm{E} \, x^2(t_k)\right) / T = \frac{1}{2} + \frac{1}{N}. \tag{13}$$

Under FDMA, each loop receives a share $U_i = 1/N$ of the bandwidth, implying the same performance regardless of the number of nodes. Computing the stationary state distribution under event-triggered sampling for different values of r, we find the optimal threshold $r = 1.06$, yielding the cost

$$J_i = 1.40. \tag{14}$$

For the CSMA case, we use Monte Carlo simulations to find the stationary variance of the plants under random or dynamic priority access. For each N, we sweep r to find the optimal threshold and the corresponding optimal cost. Each configuration was simulated for 10^8 time steps, corresponding to in the order of 10^6 simulated seconds. (The simulation time was around $15n$ seconds for each configuration on an Intel Core 2 CPU @1.83 GHz.)
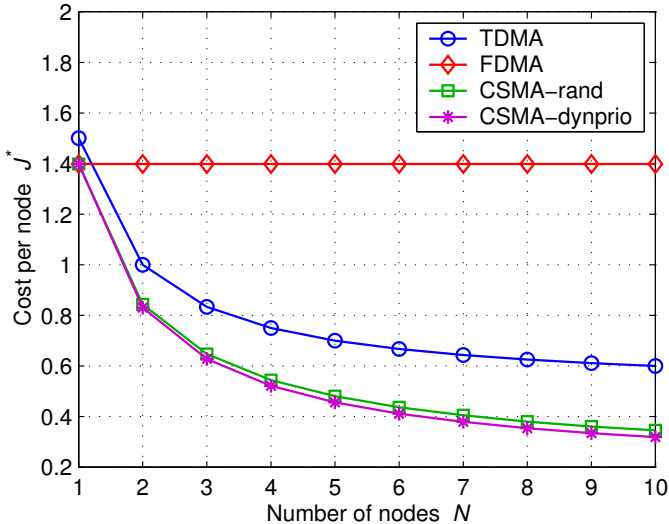
Figure 6.   Optimal cost per node vs number of nodes when controlling symmetric integrator plants.

The optimal costs under the various policies described above for $N = 1 \ldots 10$ nodes are reported in Fig. 6, and the optimal thresholds under CSMA are shown in Figs. 7. It is seen that TDMA outperforms FDMA, except for $N = 1$ where sporadic event-based control has the edge over periodic control. In turn, both variants of CSMA outperform TDMA, CSMA with dynamic priorities performing slightly better than CSMA with random access. The results are not surprising, since CSMA with event-triggered sampling dynamically allocates the bandwidth to the loop(s) most in need. A higher event threshold is needed for the random priority scheme in order to be more selective about which plant to control.

It is possible to reason about what happens when $N \to \infty$ under the various access schemes. Under TDMA, the performance approaches $J_i = 1/2$, while under FDMA, the performance is unaffected by N and is constant $J_i = 1.40$. CSMA approaches aperiodic event-based control [Åström and Bernhardsson, 1999] when $N \to \infty$, regardless of the
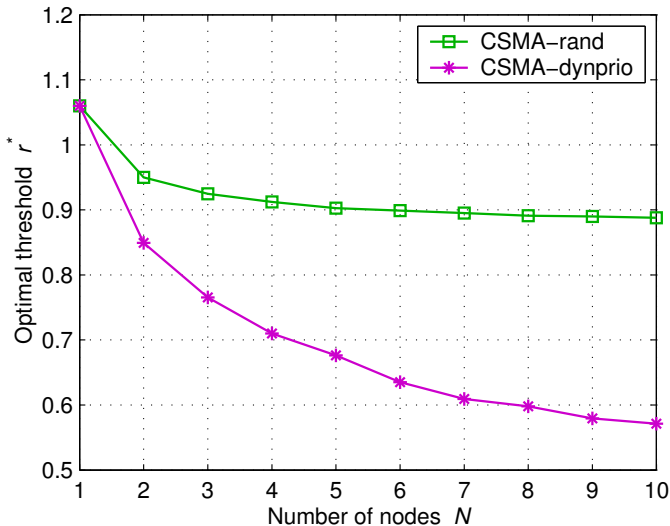
Figure 7. Optimal threshold vs number of nodes for CSMA with random or dynamic priority access when controlling symmetric integrator plants.

priority scheme used. For integrator plants, the optimal cost per plant approaches $J_i = 1/6$. Hence, CSMA asymptotically gives 67% lower cost than TDMA and 88% lower cost than FDMA when the number of control loops increases. Equivalently, one can reason about the network capacity needed to maintain the same performance as the number of integrator plants grows. Here, again, CSMA will asymptotically require 67% less bandwidth than TDMA and 88% less bandwidth than FDMA to achieve the same cost per loop.

6. Local vs Global Knowledge

One important assumption in our model is that the decisions as to whether to transmit or not are taken locally at each sensor node. It was seen above that event-triggered control under CSMA with dynamic priority access gave the lowest cost among all the considered schemes.
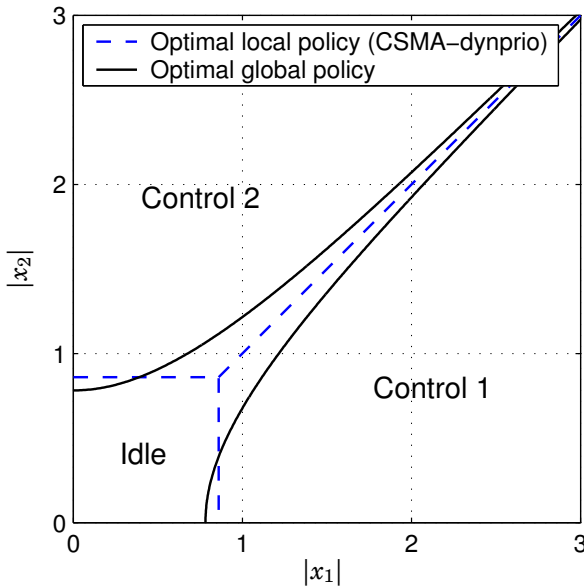
Figure 8.   Event-triggered control of two integrators: optimal global and local policies.

It is interesting to compare the performance to a controller with global knowledge of the plant states. Such a controller would of course not be implementable in a networked setting but can provide a lower bound on the achievable cost.

We consider the special case of $N = 2$ symmetric integrator plants with the minimum inter-control interval and delay $T = 1/2$. The optimal local scheme under CSMA with dynamic priorities was computed above, giving the optimal cost $J_i = 0.834$ for the threshold $r = 0.85$. For the global scheme, we gridded the plant state space in the two dimensions and applied dynamic programming to derive the optimal control policy. For each state (x_1, x_2), the controller has the choice to control to the first plant, the second plant, or to idle. The resulting optimal global control policy is shown in Fig. 8, together with the local CSMA policy with dynamic priorities. It is seen that the control policies are quite similar. One difference is that the global controller

will idle if both plants have about the same error magnitude, waiting to see where the processes will go next. The resulting cost under the global policy is found to be $J_i = 0.828$, which is only one percent lower than the cost for the optimal local scheme.

7. Results for Three Asymmetric Plants

As a final numerical example, we consider a case where three asymmetric first-order systems should be controlled: one asymptotically stable plant, one integrator, and one unstable plant. The plant parameters are $\sigma_i = 1$ and

$$a_1 = -0.5, \qquad a_2 = 0, \qquad a_3 = 0.5.$$

Further, we let $T = 1/3$. Here, intuition tells us that more resources should be allocated to the unstable plant (Plant 3) while the stable plant (Plant 1) can manage with less resources.

For TDMA, the total cost is computed for all possible cyclic schedules of length $n = 2, \ldots, 12$. Since the unstable plant must be controlled at least once per cycle, we fix the first entry in the schedule to 3, leaving about 3^{n-1} schedules to test per value of n (including "necklace duplicates"). The optimal schedule for each value of n is reported in Table 1. It is seen that the best schedule is of length 6: $\{3, 2, 3, 2, 3, 1\}$, giving a total cost of $J = 2.56$. In the optimal schedule, the stable plant is controlled once per cycle, the integrator twice, and the unstable plant three times per cycle.

For FDMA, we optimize over the bandwidths U_1, U_2, U_3 to find the lowest total cost. For each plant, we first approximate the cost function $J_i(U)$ by sweeping r for each value of U. We then apply nonlinear optimization to find the optimal shares, yielding $U_1 = 0$, $U_2 = 0.397$, $U_3 = 0.603$ and the total cost $J = 3.49$. It is interesting to note that the long delay associated with FDMA apparently makes it pointless to control the stable plant.

For CSMA, we consider two arbitration mechanisms: random access and static priorities. For the random access scheme, we sweep the three thresholds to find the minimum cost, giving $r_1 = 1.12$, $r_2 = 0.92$, $r_3 = 0.77$, and the total cost $J = 1.96$. The three loops occupy the

76

Table 1. Optimal cyclic schedules for the three asymmetric plants.

Length n	Cyclic schedule	Total cost J
2	$\{3, 2\}$	2.651
3	$\{3, 3, 2\}$	2.708
4	$\{3, 2, 3, 1\}$	2.588
5	$\{3, 2, 3, 2, 1\}$	2.650
6	$\{3, 2, 3, 2, 3, 1\}$	2.563
7	$\{3, 2, 3, 3, 2, 3, 1\}$	2.589
8	$\{3, 2, 3, 2, 3, 2, 3, 1\}$	2.567
9	$\{3, 2, 3, 3, 2, 3, 2, 3, 1\}$	2.591
10	$\{3, 2, 3, 2, 3, 1, 3, 2, 3, 1\}$	2.573
11	$\{3, 2, 3, 3, 2, 3, 1, 3, 2, 3, 1\}$	2.588
12	$\{3, 2, 3, 2, 3, 1, 3, 2, 3, 2, 3, 1\}$	2.563

network on average 14%, 22%, and 38% of the time, while it is idle 26% of the time. The relative shares for the loops are not that different from the ones generated by the optimal cyclic schedule.

For the static priority CSMA case, we assume that the unstable plant has the highest priority, the integrator has medium priority, while the stable plant has the lowest priority. Again sweeping the three thresholds and evaluating the costs gives the optimal thresholds $r_1 = 0.95$, $r_2 = 0.87$, $r_3 = 0.77$, and the total cost $J = 1.94$. The priorities allow for tighter thresholds to be utilized. The three loops occupy the network on average 15%, 25%, and 38% of the time, while it is now idle 22% of the time.

The results under the various access schemes are summarized in Table 2. We can again conclude that CSMA can provide better control performance than both TDMA and FDMA. For this example, CSMA gives 23% percent lower total cost than TDMA and 44% lower cost than FDMA. We further note that there is only a very modest improvement by using priorities, which is good news for wireless systems where random access schemes may be the only realistic choice for the implementation.

Table 2. Optimal costs for the three asymmetric plants under the various medium access schemes.

Scheme	J_1	J_2	J_3	$J = \sum J_i$
TDMA	0.690	0.889	0.984	2.56
FDMA	1.000	1.177	1.319	3.49
CSMA-rand	0.554	0.618	0.772	1.94
CSMA-statprio	0.562	0.641	0.723	1.92

8. Discussion and Conclusion

This paper has studied a prototypical networked control co-design problem, where both the control policy and network scheduling policy have been taken into account. Although very simple mathematical models were used, some interesting conclusions regarding the various medium access schemes could be drawn. CSMA with event-triggered sampling was the superior scheme in all presented examples, while FDMA performed poorly due to the long transmission delay.

The simulation-based design approach adopted in this paper is conceptually easy to extend to higher-order plants and controllers. We have noted that the simulation time required to evaluate the cost with a given accuracy grows slower than the number of states in the system. Rather, the main problem with more realistic systems is the number of controller parameters that need to be optimized. For higher-order systems, it is probably necessary to impose restrictions on the controller structure and only optimize over a small subset of the parameters.

Another interesting approach would be to develop a way to characterize the performance of an event-triggered control loop as a function of its network resource usage pattern. Integrating several control loops, it should be possible to provide guarantees on the worst-case performance of each controller. Apart from higher-order plants and controllers, several other extensions to the work in this paper are possible to imagine, including

- having the controller located in a separate node, meaning that both the transmission from sensor to controller and from controller to actuator need to be scheduled.

- having more detailed models of real network protocols, including, e.g., the random back-offs in CSMA/CD.

- allowing MIMO systems, where each sensor and actuator may reside on a different node in the network.

- modeling measurement noise, variable transmission times, and lost packets.

Acknowledgment

References

Åström, K. J. (1970): *Introduction to Stochastic Control Theory*. Academic Press, New York.

Åström, K. J. and B. Bernhardsson (1999): "Comparison of periodic and event based sampling for first-order stochastic systems." In *Preprints 14th World Congress of IFAC*, vol. J, pp. 301–306. Beijing, P.R. China.

Åström, K. J. and B. Wittenmark (1997): *Computer-Controlled Systems*. Prentice Hall.

Cervin, A., D. Henriksson, B. Lincoln, J. Eker, and K.-E. Årzén (2003): "How does control timing affect performance?" *IEEE Control Systems Magazine*, **23:3**, pp. 16–30.

Cervin, A. and E. Johannesson (2008): "Sporadic control of scalar systems with delay, jitter and measurement noise." In *Proc. 17th IFAC World Congress*. Seoul, Korea.

Hristu-Varsakelis, D. and P. R. Kumar (2002): "Interrupt-based control over a shared communication medium." In *Proc. IEEE Conference on Decision and Control*.

Johannesson, E., T. Henningsson, and A. Cervin (2007): "Sporadic control of first-order linear stochastic systems." In *Proc. 10th International Conference on Hybrid Systems: Computation and Control*, Lecture Notes in Computer Science 4416. Springer-Verlag, Pisa, Italy.

Otanez, P. G., J. R. Moyne, and D. M. Tilbury (2002): "Using deadbands to reduce communication in networked control systems." In *Proc. American Control Conference*.

Rabi, M. (2006): *Packet based Inference and Control*. PhD thesis, Institute for Systems Research, University of Maryland. URL: http://hdl.handle.net/1903/6612.

Rehbinder, H. and M. Sanfridson (2004): "Scheduling of a limited communication channel for optimal control." *Automatica*, **30:3**, pp. 491–500.

Walsh, G. C., H. Ye, and L. Bushnell (1999): "Stability analysis of networked control systems." In *Proc. American Control Conference*.

Paper III

Log-concave Observers

Toivo Henningsson   Karl Johan Åström

Abstract

The Kalman filter is the optimal state observer in the case of linear dynamics and Gaussian noise. In this paper, the observer problem is studied when process noise and measurements are generalized from Gaussian to log-concave. This generalization is of interest for example in the case where observations only give information that the signal is in a given range. It turns out that the optimal observer preserves log-concavity. The concept of strong log-concavity is introduced and two new theorems are derived to compute upper bounds on optimal observer covariance in the log-concave case. The theory is applied to a system with threshold based measurements, which are log-concave but far from Gaussian.

1. Introduction

The Kalman filter (see [Kalman, 1960], [Kalman and Bucy, 1961]) is one of the most widely used schemes for state estimation from noisy measurements. It is optimal for linear measurements and Gaussian noise, but it is often applied in a more general setting. Although the Extended Kalman filter (see [Gelb and Corporation., 1974]) often works well in practice, sometimes it does not, and it is in general not easy to see how altered conditions change the observer problem.

In this paper, a particular generalization is investigated where measurements and noise are allowed to be log-concave (see [Prékopa, 1971], [Prékopa, 1973], [Bagnoli and Bergstrom, 1989], [An, 1996]). The model of log-concave measurements is applicable in many instances where the assumption of independent additive measurement noise is too limited, for instance with heavy quantization, or with the problem of event based sampling discussed in [Åström and Bernhardsson, 2002].

Within this framework, the problem of moving horizon ML/MAP estimation becomes a convex optimization problem, see [Schön *et al.*, 2003]. This paper will however focus on the covariance of the Bayesian Observer, which is investigated and compared with the Kalman filter.

Strongly log-concave functions are introduced as a means to quantify observer properties. Two new theorems are applied to derive upper bounds on optimal observer covariance.

It turns out that the observer problem is still quite well behaved so that, especially with some insight gained in the analysis, a Kalman filter might often be usable for this more general measurement setting. For a more thorough treatment, see [Henningsson, 2005].

The paper is organized as follows. A motivating example is presented in section 2. The notion of log-concavity is introduced in section 3, where we state the main results as theorem 1 and 2. In section 4 we treat the observer problem. The results in section 3 are used to investigate the observer properties. Finally in section 5 the results are applied to the example.

2. Example: A MEMS accelerometer

Consider an accelerometer based on the following design. A test mass is suspended to move freely in one dimension and is affected by an external acceleration. Sensors detect deviations from the origin exceeding a detection threshold and report the sign of the deviation. An input signal is available to accelerate the test mass so as to keep it close to the origin. The aim of the design is to estimate the external acceleration as accurately as possible.

The discrete time dynamics are given by

$$x(k) = \begin{pmatrix} 1 & h \\ 0 & 1 \end{pmatrix} x(k-1) + \begin{pmatrix} \frac{1}{2}h^2 \\ h \end{pmatrix} u(k-1) + v(k-1),$$

where x is the state, u the input signal, v the external acceleration and h the sampling period. The state consists of position x_1 and velocity x_2. With the external acceleration as a white noise disturbance, sampling yields v to be Gaussian white noise with covariance

$$P_N = \sigma^2 \begin{pmatrix} \frac{1}{3}h^3 & \frac{1}{2}h^2 \\ \frac{1}{2}h^2 & h \end{pmatrix},$$

where σ^2 is the process noise intensity.

The measurements are given by

$$y(k) = \begin{cases} \text{sign}(x_1(k)), & |x_1(k)| \geq 1 \\ 0, & \text{otherwise,} \end{cases}$$

which is the only non-classical assumption used in the model. The output $y(k)$ is not readily described as a linear combination of state and uncorrelated measurement noise, making a straightforward application of Kalman filter theory difficult.

In fact, it is not at all obvious what properties to expect for this observer problem; will the observer error remain bounded, how large will it be, how does it depend on the measurement sequence, how complex observer is necessary, and so on. To answer questions about the

observer problem, the Bayesian observer for the system will be analyzed. Other examples where similar measurement conditions apply are when measurements are coarsely quantized or come in the form of level triggered events.

3. Log-concavity

Many results are available on general log-concavity, see for instance [Prékopa, 1971], [Prékopa, 1973], [Bagnoli and Bergstrom, 1989], and [An, 1996]. The book [Boyd and Vandenberghe, 2004] contains much material on convex functions that can easily be transfered to the log-concave case. Here, only the properties that are most relevant in the context of this paper will be stated.

A log-concave function is a function with concave logarithm. Log-concave functions are well suited for applying convexity theory to probability densities; many common densities are log-concave and several useful operations preserve log-concavity. In contrast, no probability density on $\mathbb{R}^n$ is either convex or concave since probability densities have a finite integral while convex and concave functions on $\mathbb{R}^n$ do not.

DEFINITION 1—LOG-CONCAVE FUNCTION
A function $f : \mathbb{R}^n \longrightarrow \mathbb{R}$ is logarithmic concave or *log-concave*, iff $f(x) \geq 0$, f has convex support and $\ln(f(x))$ is concave on this support. ☐

For some simple examples of log-concave functions see figure 1, and for some counterexamples figure 2. Among common log-concave densities are Gaussian and exponential densities.

Log-concave functions are unimodal, meaning that the superlevel sets $\{x; f(x) \geq a\}, a \in \mathbb{R}$ are convex. Many attractive properties of log-concave functions are analogous to those for concave functions. A useful fact is that multiplication takes the place of addition so that the product of two log-concave functions is log-concave. Another very useful result derived by Prékopa is
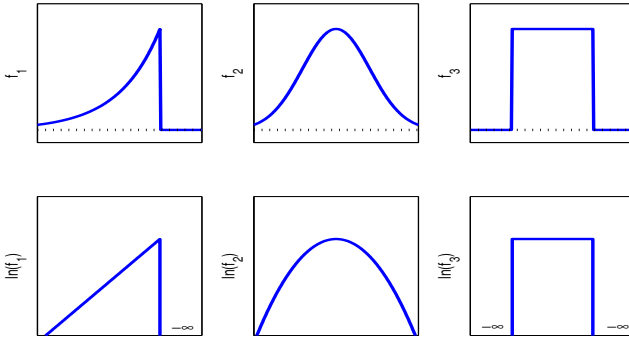
Figure 1. Some examples of log-concave functions in one variable; the function is plotted above and its logarithm below. The dotted line is $f = 0$, and $\ln(0)$ is taken to be $-\infty$. f_1: Truncated exponential function, f_2: Gaussian function, f_3: rectangular window.
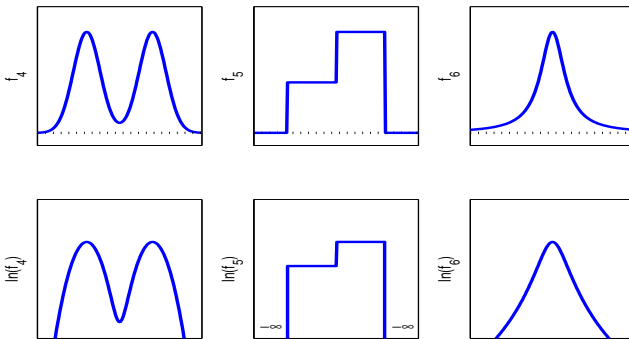

Figure 2. Some examples of functions in one variable that are *not log-concave*; the function is plotted above and its logarithm below. f_4: Not unimodal, f_5: Discontinuous on interior of support, $f_6(x) = \frac{1}{1+x^2}$: sub-exponential decay.

PROPOSITION 1—PRÉKOPA
Let $f(x,y)$ be jointly log-concave in $x \in \mathbb{R}^m, y \in \mathbb{R}^n$. Then the integral

$$g(x) = \int f(x,y)dy$$

is a log-concave function of x. $\qquad\qquad\qquad\qquad\qquad\qquad\qquad$ □

Proof. See [Prékopa, 1971] and [Prékopa, 1973].

This theorem implies for instance that the marginal densities of log-concave densities are log-concave, and that the convolutions of log-concave functions are log-concave. It will be central in the proof of theorems 1 and 2.

3.1 Strong log-concavity

Log-concavity is in its nature only a qualitative property. To allow for quantitative statements, the following class of functions is introduced.

DEFINITION 2—STRONGLY LOG-CONCAVE FUNCTION
Let $P \in \mathbb{R}^{n \times n}$ be positive definite and define the set

$$\mathcal{LC}(P^{-1}) = \left\{ f; f_0(x) = \frac{f(x)}{e^{-\frac{1}{2}x^T P^{-1} x}} \text{ log-concave} \right\}.$$

The function f is *strongly log-concave* of strength P^{-1} iff $f \in \mathcal{LC}(P^{-1})$.
$\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad$ □

All strongly log-concave functions are log-concave, bounded and go to zero as $|x| \to \infty$ at least as fast as a Gaussian function.

Membership in $\mathcal{LC}(P^{-1})$ can be seen as an inequality, in the sense that

$$f \in \mathcal{LC}(P^{-1}), \quad P \leq R$$
$$\implies f \in \mathcal{LC}(R^{-1}).$$

The inclusion $f \in \mathcal{LC}(P^{-1})$ is *tight* iff $\mathcal{LC}(P^{-1})$ is a subset of all $\mathcal{LC}(R^{-1})$ that contain f.

A Gaussian density with covariance P is tightly in $\mathcal{LC}(P^{-1})$, and can be seen as the *corresponding Gaussian* to this class. The definition states that any strongly log-concave function can be written as the product of a log-concave function and a corresponding Gaussian. Also, the following properties hold:

THEOREM 1—ENCAPSULATION PROPERTY
If $f \in \mathcal{LC}(F^{-1})$ and $g \in \mathcal{LC}(G^{-1})$ then

$$f(Ax + b) \in \mathcal{LC}(A^T F^{-1} A)$$
$$(f * g)(x) \in \mathcal{LC}\left((F + G)^{-1}\right)$$
$$f(x) \cdot g(x) \in \mathcal{LC}(F^{-1} + G^{-1}),$$

where $x, b \in \mathbb{R}^n, A \in \mathbb{R}^{n \times n}$ and $f * g$ is the convolution of f and g. ☐

Proof. See appendix A.

The inclusions are as narrow as the premises allow, being tight when f and g are the corresponding Gaussians.

THEOREM 2—COVARIANCE BOUND
If $f \in \mathcal{LC}(P^{-1})$ is a probability density then

$$V = \int (x - m_x)(x - m_x)^T f(x) dx \leq P,$$

where $m_x = \int x f(x) dx$. The bound is tight for the corresponding Gaussian. ☐

Proof. See appendix B.

The matrix expressions for strength of log-concavity correspond exactly to the way that the operations propagate inverse covariances for Gaussian functions. By the latter theorem, the inverse strength of log-concavity is an upper bound on the covariance.

The theorems form a chain of inequalities that can be used to propagate upper bounds on covariance under the operations of affine transformation, convolution and multiplication. For more properties of strongly log-concave functions, see [Henningsson, 2005].

4. Log-concave observers

The observer problem that will be considered is for processes with linear dynamics and log-concave noise and measurements, as defined below.

The dynamics are given by

$$x(k) = Ax(k-1) + Bu(k-1) + v(k-1),$$

where x is the state, u the input and v the process noise. The noise has log-concave probability density f_N. The matrices A and B, as well as f_N may be time-varying.

The measurements are described by the stochastic variables $Y(k)$,

$$f_{Y(k)|X(k)}\big(y|x(k)\big) = f_M\big(y, x(k)\big),$$

where the measurement function f_M is log-concave in x for each y and may be time-varying.

4.1 The Bayesian observer

As a basis for the analysis, the online Bayesian observer for estimation of $x(k)$ from the history of y and u will be considered. The state of the observer at any time is fully described by the function

$$f_k(x) = f_{X_k|y_{1:k}, f_{X_0}}(x),$$

where $y_{1:k}$ is the measurement history and f_{X_0} is the assumed initial density.

The observer update from f_{k-1} to f_k is best described in three steps taking into account dynamics, process noise, and measurements:

$$
\begin{align}
f_k^{\mathrm{d}}(x) &\propto f_{k-1}\big(A^{-1}x - A^{-1}Bu(k-1)\big), \tag{1}\\
f_k^{\mathrm{dn}}(x) &= (f_N * f_k^{\mathrm{d}})(x), \tag{2}\\
f_k(x) &\propto f_M(y(k), x) \cdot f_k^{\mathrm{dn}}(x), \tag{3}
\end{align}
$$

where $\propto$ denotes proportionality and A is assumed to be invertible. For the derivation, see [Henningsson, 2005]. The dynamics update corresponds to an affine transformation, the noise update to a convolution with f_N, and the measurement update to a multiplication with $f_M(y(k), \cdot)$. For an illustration, see figures 3, 4 and 5. If f_{X_0}, f_N and $f_M(y, \cdot)$ are Gaussian, the observer updates (1)-(3) reduce to a Kalman filter.
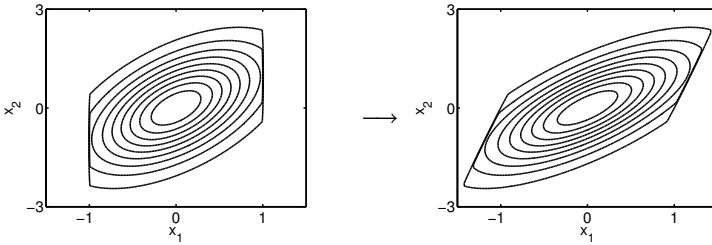
Figure 3.   Illustration of the dynamics update for the MEMS accelerometer observer. The transformation amounts to a shear in this case.
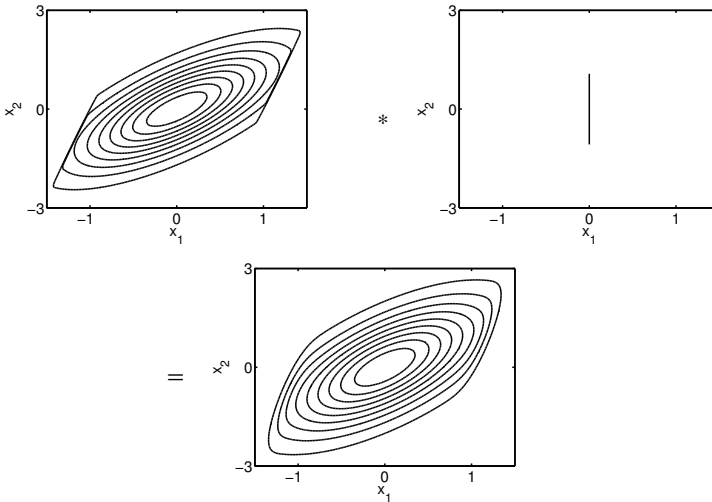
Figure 4.   Illustration of the process noise update for the MEMS accelerometer observer. The Gaussian noise enters almost exclusively in the x_2 direction.

4.2 Properties

Since log-concavity is preserved under affine parameter transformation, convolution, and multiplication, all f_k are log-concave if f_{X_0} is log-concave.

Theorem 1 can be used to propagate upper bounds on observer covariance. This approach can be used to asses the merits of a particular
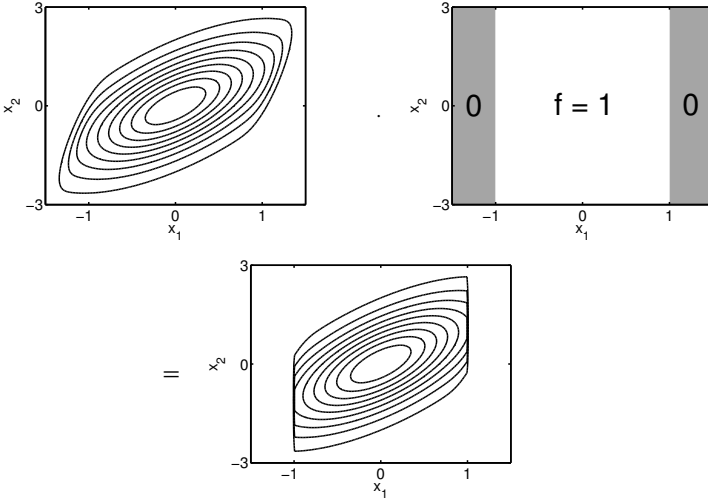
Figure 5.   Illustration of the measurement update for the MEMS accelerometer observer. The measurement is $y = 0$.

sensor setup, or together with some information about the localization of f_k to give state estimates with error bounds. The computations of covariance propagation have the structure of a Kalman filter applied to corresponding Gaussians.

5. An Application

The MEMS accelerometer will now be used to illustrate how the theory can be applied in the analysis of a concrete observer problem.

5.1 Analytical covariance bounds

The accelerometer has linear dynamics and log-concave noise and measurements. The process noise density f_N is Gaussian with covariance P_N, so that $f_N \in \mathcal{LC}(P_N^{-1})$. The measurement function $f_M(y, x)$ is log-concave in x for all y, see figure 6.

Applying theorem 1 directly leads in this case to a highly conservative covariance bound, achieved when completely ignoring the mea-
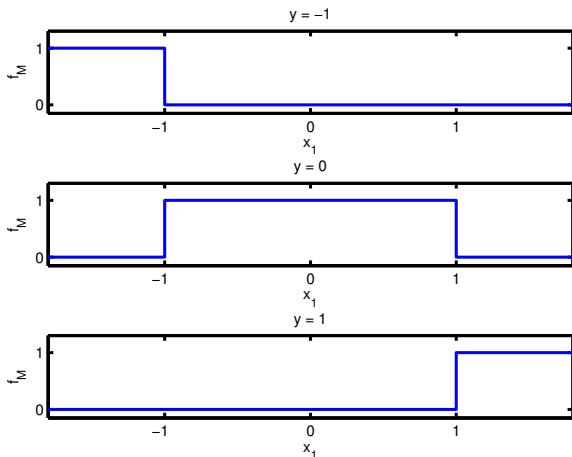
Figure 6. The measurement function $f_M(y, x)$ for the MEMS accelerometer describing the relative probability of state x when $y = -1, 0, -1$. The function is independent of x_2.

surements. The bound grows cubically with time. Grid based finite difference simulations of the Bayesian observer do however indicate that the covariance is bounded, and if the output changes frequently, small.

The reason why the bound is so conservative is that f_M is not strongly log-concave for any y; strength of log-concavity being the only measure of information that the theorem considers. In lack of stronger proven results, a slight approximation will allow to account for the major source of state information.

The most important source of state information under normal conditions is the events when y goes from being 0 to ± 1, at which time x_1 is known to be almost exactly equal to y. This can be modeled as a Gaussian measurement of x_1 with expectation y and variance σ_M^2.

The variance σ_M^2 will depend on the process noise and uncertainty in velocity, but will be small when h is small. The modified measurement function $\hat{f}_M$ can be seen in figure 7. For events, $\hat{f}_M(\pm 1, \cdot) \in$
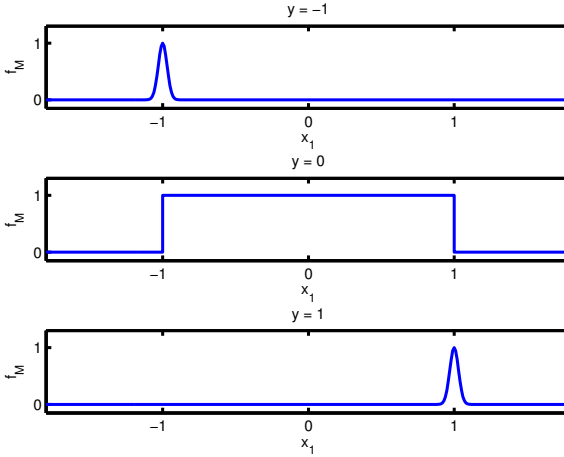
Figure 7. The modified measurement function $\hat{f}_M(y,x)$ for the MEMS accelerometer when $y = -1, 0, -1$. For $y = \pm 1$, the function has been changed to a narrow Gaussian centered on the detection threshold.

$\mathcal{LC}(Q_M)$ where

$$Q_M = \begin{pmatrix} \sigma_M^{-2} & 0 \\ 0 & 0 \end{pmatrix},$$

and otherwise $\hat{f}_M(0, \cdot) \in \mathcal{LC}(0)$.

Under this approximation, the variance of the optimal estimate $\hat{x}_2$ of x_2 right after an event can be shown to satisfy

$$V(\hat{x}_2) \leq \frac{1}{3}\sigma^2 t + 2\sigma_M^2 t^{-2},$$

where t is the time since the last event. For the derivation, see [Henningsson, 2005].

The bound illustrates that the accuracy of the accelerometer depends strongly on the rate of events. If the objective of control is good measurements, the controller should keep the rate above a certain level, for instance sending the test mass bouncing in a ping pong fashion between the detection boundaries.

5.2 Kalman filter approximation

A Kalman filter was tuned to give a reasonable approximation of the Bayesian observer. The crucial issue was to assign the covariance of the measurement $y = 0$. While a single measurement $y = 0$ predicts x_1 to have expectation zero with variance $\sigma^2 = \frac{1}{3}$, there is much less additional information in the measurement $y = 0$ at the next time step.

In this case it is reasonable to design the Kalman filter by choosing the stationary variance p_{11}^{stat} of x_1 when $y = 0$. The variance would typically be $p_{11}^{\text{stat}} = \frac{1}{\sqrt{3}}$ (rectangular distribution) or a little less. From solving the Riccati equation, it is found that the measurement variance $\sigma_o^2 h^{-1}$ for $y = 0$ must be chosen according to

$$\sigma_o = 2^{-1/3}(p_{11}^{\text{stat}})^{2/3}\sigma^{-1/3},$$

where σ^2 is the process noise intensity.

5.3 Simulation

Figure 8 shows a comparison of actual and predicted variances for a simulation of the Bayesian observer. The variance σ_M^2 was chosen so that the approximate upper bound would always be conservative. The upper bound is quite tight some time after each event, but then diverges. The variance of the tuned Kalman filter appears to be an only mildly conservative approximation of the actual variance. As long as the rate of events is reasonably high, the approximate upper bound is very tight.

A simple control law was devised to control the rate of events, and simulations were run for different rates to compare observer performance for the grid filter and the tuned Kalman filter. Figure 9 shows the observer error as a function of mean time between events t_{mean}. The grid filter is slightly better than the tuned Kalman filter and considerably better than the approximate covariance bound down to values of t_{mean} around 0.4.

For lower t_{mean} it seems that the grid filter scheme encounters discretization issues. At the same time, the tuned Kalman filter comes very close to the approximate upper bound which appears to be very tight in this region, indicating that the observer problem is very similar to the Kalman filter case for high rates. This similarity is not
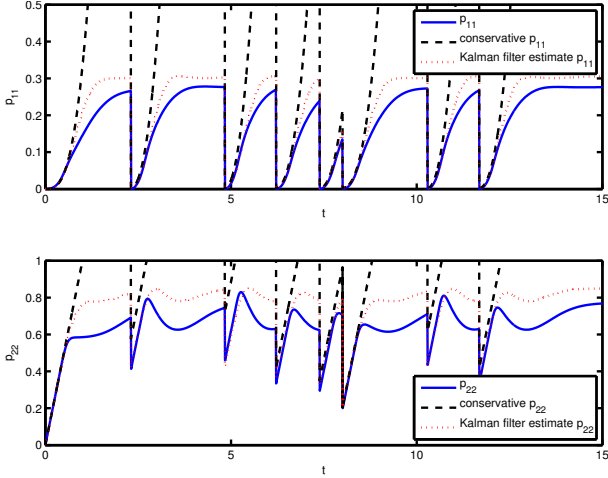
Figure 8. Observer covariances during a simulation for the MEMS accelerometer: grid filter, approximate upper bound and tuned Kalman filter.

surprising since when the covariance is small, the bulk of probability mass is concentrated in a small region which is only seldom affected by the non Gaussian measurements.

Thus it is seen that the upper bound derived from the theory is quite tight when the rate of events is high and that if the inherent correlation in the non Gaussian measurements is considered, a Kalman filter can be applied as a close to optimal observer.

EXAMPLE 1—QUANTIZED MEASUREMENTS
In the previous example it was necessary to rely on approximations because the measurement functions were not strongly log-concave. If the measurement function can be chosen freely, much stronger results are possible.

Consider the general problem of estimating a scalar variable from a series of independent identically distributed quantized measurements. The objective is to find a conditional measurement distribution, or measurement function, that is in some sense optimal. Using strength of log-concavity as an optimality criterion one can formulate the following problem:
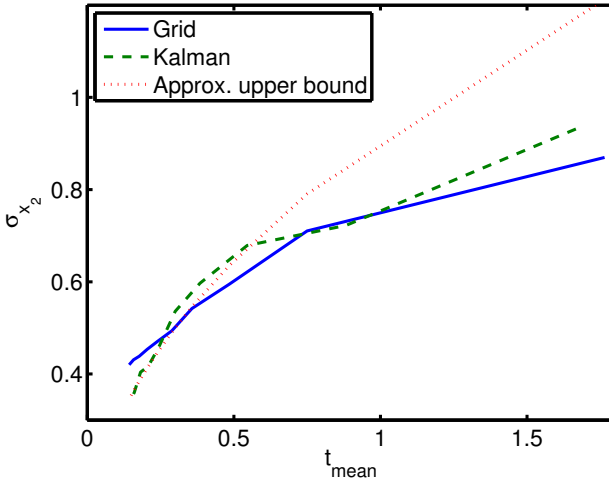
Figure 9. RMS x_2 estimation error as a function of mean time between events: grid filter estimation error, tuned Kalman filter estimation error, and approximate upper bound. For too high event rates, the grid filter suffers from discretization problems.

Let the independent measurements y be distributed according to

$$f_{Y|x}(y|x) = f(x - y), \quad y \in \mathcal{Z},$$

where x is the variable to be estimated. Find a function $f \in \mathcal{LC}(p^{-1})$, where $p > 0$ is as small as possible, such that

$$f(x) \geq 0,$$
$$f(-x) = f(x),$$
$$\sum_{k=-\infty}^{\infty} f(x - k) = 1.$$
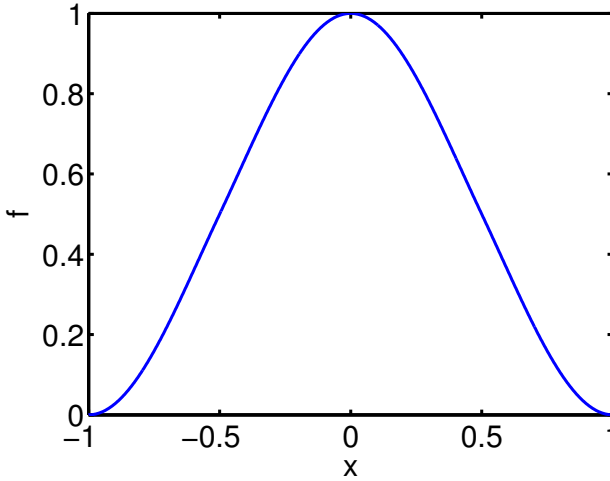
Figure 10.   The measurement function in example 1. The function is Gaussian when $|x| \leq \frac{1}{2}$ and zero when $|x| \geq 1$.

The solution is given by the function

$$
f(x) = \begin{cases} 2^{-4|x|^2}, & |x| \leq \dfrac{1}{2}, \\[2mm] 1 - 2^{-4(1-|x|)^2}, & \dfrac{1}{2} < |x| \leq 1, \\[2mm] 0, & \text{otherwise}, \end{cases}
$$

satisfying $f(x) \in \mathcal{LC}\big(8\ln(2)\big)$. The function is plotted in figure 10, and in log-scale in figure 11. A series of n measurements with f as measurement function is guaranteed to yield a probability density in $\mathcal{LC}\big(n \cdot 8\ln(2)\big)$ and therefore a variance satisfying $\sigma^2 \leq \frac{1}{n \cdot 8\ln(2)}$.

□
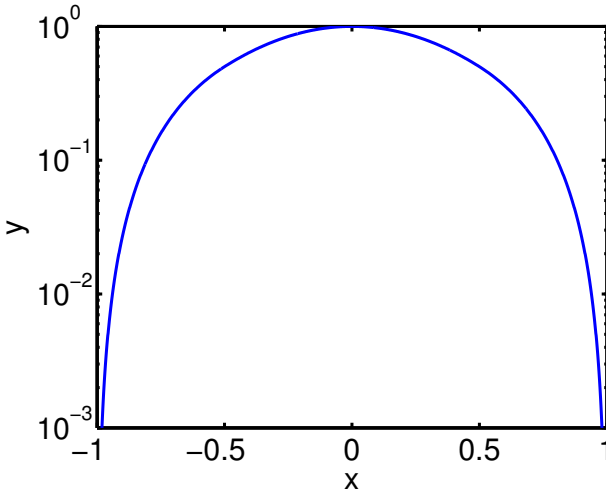
Figure 11. The measurement function in example 1 in log-scale. The logarithm is clearly concave, being quadratic when $|x| \leq \frac{1}{2}$.

6. Conclusion

Log-concavity is a powerful tool when dealing with probability densities. The generalization to allow log-concave densities in the observer widens the range of application considerably compared to the Kalman filter. Although no closed form solution exists in the general case, the observer problem is still very accessible to mathematical treatment.

Regarding observability and observer performance, strongly log-concave functions together with theorems 1 and 2 can be applied to derive simple bounds on achievable observer covariance.

An in-depth treatment of the log-concave case gives a greater understanding of the performance of an Extended Kalman filter. In design of instrumentation, striving for log-concave measurement functions can facilitate the observer problem.

Acknowledgement

References

An, M. (1996): "Log-concave probability distributions : Theory and statistical testing." Papers 96-01. Centre for Labour Market and Social Research, Denmark.

Åström, K. J. and B. Bernhardsson (2002): "Comparison of Riemann and Lebesque sampling for first order stochastic systems." In *Proceedings of the 41st IEEE Conference on Decision and Control*.

Bagnoli, M. and T. Bergstrom (1989): "Log-concave probability and its applications." Papers 89-23. Michigan - Center for Research on Economic & Social Theory.

Boyd, S. and L. Vandenberghe (2004): *Convex Optimization*. Cambridge University Press, New York, NY, USA.

Gelb, A. and A. S. Corporation. (1974): *Applied optimal estimation / written by Technical Staff, Analytic Sciences Corporation ; edited by Arthur Gelb, principal authors, Arthur Gelb ... [et al.]*. M.I.T. Press Cambridge, Mass. :.

Henningsson, T. (2005): "Logarithmic concave observers." Master's Thesis ISRN LUTFD2/TFRT--5747--SE. Department of Automatic Control, Lund University, Sweden.

Kalman, R. E. (1960): "A new approach to linear filtering and prediction problems." *Transactions of the ASME–Journal of Basic Engineering*, **82:Series D**, pp. 35–45.

Kalman, R. E. and R. S. Bucy (1961): "New results in linear filtering and prediction theory." *Transactions of the ASME - Journal of Basic Engineering*, **83**, pp. 95–107.

Prékopa, A. (1971): "Logarithmic concave measures with application to stochastic programming." *Acta Sci. Math. (Szeged)*, **32**, pp. 301–316.

Prékopa, A. (1973): "On logarithmic concave measures and functions." *Acta Sci. math. 34, 335-343*.

Schön, T., F. Gustafsson, and A. Hansson (2003): "A note on state estimation as a convex optimization problem." *Acoustics, Speech, and Signal Processing, 2003. Proceedings. (ICASSP '03). 2003 IEEE International Conference on*, **6**, April, pp. VI–61–4 vol.6.

A. Proof of theorem 1

The proofs are based on the fact that a function f is in $\mathcal{LC}(F^{-1})$ iff it can be factored as

$$f(x) = e^{-\frac{1}{2}x^T F^{-1} x} f_0(x), \tag{4}$$

where $f_0(x)$ is log-concave. This follows from the definition.

The proofs for affine transformation and multiplication are straightforward, while the proof for convolution is a little more involved.

A.1 Affine transformation

Let $f \in \mathcal{LC}(F^{-1})$, $A \in \mathbb{R}^{n \times n}$, $b \in \mathbb{R}^n$ and $y = Ax + b$. Then

$$
\begin{aligned}
g(x) &= f(Ax + b) \\
&= e^{-\frac{1}{2}(Ax+b)^T F^{-1}(Ax+b)} \cdot f_0(y) \\
&= e^{-\frac{1}{2}(x^T A^T F^{-1} Ax + 2b^T F^{-1} Ax + b^T F^{-1} b)} \cdot f_0(y) \\
&= e^{-\frac{1}{2}x^T (A^T F^{-1} A)x} \cdot \underbrace{\left(e^{-\frac{1}{2}b^T F^{-1} b} e^{-(A^T F^{-1} b)^T x} f_0(y) \right)}_{g_0(x)} .
\end{aligned}
$$

We see that g_0 is the product of a constant, an exponential function and a log-concave function, since log-concavity is preserved under affine parameter transformation. Then g_0 is log-concave because each of the factors is log-concave. Thus $g \in \mathcal{LC}(A^T F^{-1} A)$.

A.2 Convolution

For the proof we need the following matrix identity. Let A, B and C be positive definite matrices such that $C^{-1} = A^{-1} + B^{-1}$, or $C = A(A + $

$B)^{-1}B$. Let x, y and $z = y - (A + B)^{-1}Bx$ be vectors. Then

$$z^T(A + B)z = y^T(A + B)y - 2x^TBy + x^TB(A + B)^{-1}Bx$$

and

$$\begin{aligned}
x^TCx + z^T(A + B)z &= x^TA(A + B)^{-1}Bx + z^T(A + B)z \\
&= x^TBx + y^T(A + B)y - 2x^TBy \\
&= y^TAy + (x - y)^TB(x - y),
\end{aligned}$$

that is,

$$y^TAy + (x - y)^TB(x - y) = x^TCx + z^T(A + B)z, \tag{5}$$

which can be seen as completion of squares in x.

Let $f \in \mathcal{LC}(F^{-1})$ and $g \in \mathcal{LC}(G^{-1})$. Then

$$\begin{aligned}
h(x) &= (f * g)(x) \\
&= \int f(y)g(x - y)dy \\
&= \int e^{-\frac{1}{2}y^TF^{-1}y}e^{-\frac{1}{2}(x-y)^TG^{-1}(x-y)} \cdot f_0(y)g_0(x - y)dy \\
&= \int e^{-\frac{1}{2}\left(y^TF^{-1}y + (x-y)^TG^{-1}(x-y)\right)} \cdot f_0(y)g_0(x - y)dy.
\end{aligned}$$

Applying (5) with $A = F^{-1}, B = G^{-1}$ and $C = H^{-1}$ yields $H^{-1} = (F + G)^{-1}$ and

$$\begin{aligned}
h(x) &= \int e^{-\frac{1}{2}\left(x^TH^{-1}x + z^T(F^{-1}+G^{-1})z\right)} \cdot f_0(y)g_0(x - y)dy \\
&= e^{-\frac{1}{2}x^TH^{-1}x} \underbrace{\int e^{-\frac{1}{2}z^T(F^{-1}+G^{-1})z} \cdot f_0(y)g_0(x - y)dy}_{h_0(x)}.
\end{aligned}$$

The integrand is log-concave since it is the product of a Gaussian function and two log-concave functions, and thus h_0 is log-concave according to theorem 1. This proves that $h \in \mathcal{LC}(H^{-1}) = \mathcal{LC}((F + G)^{-1})$.

A.3 Multiplication

Let $f \in \mathcal{LC}(F^{-1})$ and $g \in \mathcal{LC}(G^{-1})$. Then

$$
\begin{aligned}
h(x) &= f(x)g(x) \\
&= e^{-\frac{1}{2}x^T F^{-1} x} e^{-\frac{1}{2}x^T G^{-1} x} \cdot f_0(x)g_0(x) \\
&= e^{-\frac{1}{2}x^T (F^{-1} + G^{-1})x} \cdot h_0(x),
\end{aligned}
$$

where f_0 and g_0 are log-concave and $h_0(x) = f_0(x)g_0(x)$. Thus h_0 is log-concave and so $h \in \mathcal{LC}(F^{-1} + G^{-1})$.

B. Proof of theorem 2

The factorization (4) will be central also in this proof. Consider first the theorem in one dimension. Let $f \in \mathcal{LC}(p^{-1}), p > 0$ be a nonincreasing probability density defined for $x \geq 0$. Then f can be factored as

$$
f(x) = e^{-\frac{1}{2}p^{-1}x^2} f_0(x),
$$

where $f_0(x), x \geq 0$ is log-concave. The right derivative $f'(0)$ exists since any convex function is almost everywhere differentiable which transfers trivially to log-concave functions. Furthermore $f_0'(0) = f'(0) \leq 0$, and since f_0 is log-concave it is nonincreasing for all $x \geq 0$.

Let $C > 0$ be defined such that

$$
\int_0^\infty C e^{-\frac{1}{2}p^{-1}x^2}\,dx = \int_0^\infty \underbrace{e^{-\frac{1}{2}p^{-1}x^2} f_0(x)}_{f(x)}\,dx = 1.
$$

Then, since $f_0(x)$ is nonincreasing, there must exist some $x_0 > 0$ such that

$$
\begin{aligned}
f_0(x) &\geq C, \quad x < x_0 \\
f_0(x) &\leq C, \quad x > x_0.
\end{aligned}
$$

The second moment of f is

$$\int_0^\infty x^2 f(x)dx = \int_0^\infty x^2 C e^{-\frac{1}{2}p^{-1}x^2} dx + \int_0^\infty x^2 \left(f(x) - C e^{-\frac{1}{2}p^{-1}x^2} \right) dx$$

$$= p + \int_0^\infty x^2 e^{-\frac{1}{2}p^{-1}x^2} \left(f_0(x) - C \right) dx$$

$$= p + \int_0^\infty e^{-\frac{1}{2}p^{-1}x^2} \left(x_0^2 + (x^2 - x_0^2) \right) \left(f_0(x) - C \right) dx$$

$$\leq p + x_0^2 \int_0^\infty e^{-\frac{1}{2}p^{-1}x^2} \left(f_0(x) - C \right) dx$$

$$= p,$$

where we have used that $(x^2 - x_0^2)(f_0(x) - C) \leq 0$. Thus the second moment of f around $x = 0$ is $\leq p$.

Now assume that $f(x) \in \mathcal{LC}(p^{-1})$ is an arbitrary strongly log-concave function in one dimension that assumes its maximum value at $x = M_x$. All strongly log-concave functions are bounded and go to zero as $|x| \to \infty$, so if f does not assume its maximum it can be made to do so by changing the value in one point, which does not affect integrals of f and preserves strong log-concavity. Then $g(x) = f(x - M_x)$ can be written as a convex combination of two probability densities in $\mathcal{LC}(p^{-1})$ such that one has support on $x < 0$ and is nondecreasing and one has support on $x \geq 0$ and is nonincreasing. The second moment of g around 0 is a convex combination of the moments of the two densities, and so

$$\int (x - M_x)^2 f(x)dx \leq p.$$

Since the covariance of the density f is the minimum of the second moment around any point,

$$\int (x - m_x)^2 f(x)dx = \min_y \int (x - y)^2 f(x)dx \leq p,$$

where m_x is expectation of the density. This proves the theorem in one dimension.

For the proof in R^n we shall need another matrix inequality. In the Cauchy-Schwartz inequality $(u^T v)^2 \leq (u^T u)(v^T v)$, let $u = P^{-\frac{1}{2}}x$ and

$v = P^{\frac{1}{2}} e_z$, where $P > 0$, $||e_z|| = 1$. This yields

$$\begin{aligned}
(x^T e_z)^2 &\leq (x^T P^{-1} x)(e_z^T P e_z) \\
\implies x^T e_z (e_z^T P e_z)^{-1} e_z^T x &\leq x^T P^{-1} x \\
\implies Q_r = e_z (e_z^T P e_z)^{-1} e_z^T &\leq P^{-1}.
\end{aligned} \tag{6}$$

Now consider a density $f \in \mathcal{LC}(P^{-1}), P > 0$. Without loss of generality, assume the expectation to be zero. The covariance is then

$$V = \int xx^T f(x) dx,$$

and for a given unit vector e_z,

$$e_z^T V e_z = \int (e_z^T x)^2 f(x) dx = \int_{t \in \mathbb{R}} t^2 \underbrace{\int_{y \perp e_z} f(t e_z + y) dy}_{g(t)} dt,$$

where $x = t e_z + y$ and $g(t)$ is the marginal density of f in the e_z direction, having zero expectation. We see that

$$\begin{aligned}
g(t) &= \int_{y \perp e_z} e^{-\frac{1}{2} x^T P^{-1} x} f_0(x) dy \\
&= \int_{y \perp e_z} e^{-\frac{1}{2} x^T Q_r x} e^{-\frac{1}{2} x^T (P^{-1} - Q_r) x} f_0(x) dy \\
&= e^{-\frac{1}{2}(e_z^T P e_z)^{-1} t^2} \underbrace{\int_{y \perp e_z} e^{-\frac{1}{2} x^T (P^{-1} - Q_r) x} f_0(x) dy}_{g_0(t)},
\end{aligned}$$

since $y^T e_z = 0$ so that $x^T Q_r x = t e_z^T Q_r e_z t = (e_z^T P e_z)^{-1} t^2$. From (6) $P^{-1} - Q_r \geq 0$ so that g_0 is log-concave. Thus $g \in \mathcal{LC}\left((e_z^T P e_z)^{-1}\right)$ so that

$$e_z^T V e_z \leq e_z^T P e_z \implies V \leq P,$$

which proves the theorem. The bound is tight for the corresponding Gaussian by definition.

Paper IV

Recursive State Estimation for Linear Systems with Mixed Stochastic and Set-Bounded Disturbances

Toivo Henningsson

Abstract

Recursive state estimation is considered for discrete time linear systems with mixed process and measurement disturbances that have stochastic and (convex) set-bounded terms. The state estimate is formed as a linear combination of initial guess and measurements, giving an estimation error of the same mixed type (and causing minimal interference between the two kinds of error). An ellipsoidal over-approximation to the set-bounded estimation error term allows to formulate a linear matrix inequality (LMI) for optimization of the filter gain, considering both parts of the estimation error in the objective. With purely stochastic disturbances, the standard Kalman Filter is recovered. The state estimator is shown to work well for an event based estimation example, where measurements are very coarsely quantized.

1. Introduction

In many control systems, there exist some disturbances that are best modelled as stochastic, and other disturbances that are better modelled as set-bounded uncertainties. The classical approach to state estimation in such cases is to approximate the set-bounded uncertainties by stochastic ones, allowing to use a standard Kalman Filter. Another approach is to approximate the stochastic disturbances by set-bounded ones, and use a state estimator for set-bounded uncertainty.

It is, however, not straightforward to translate between stochastic and set-bounded disturbances, since they do not combine in the same way. Two measurements of the same variable with independent identically distributed (I.I.D.) stochastic noise combine to form an estimate with only half the error variance. Two measurements with set-bounded uncertainty $y_i = x + z_i, |z_i| \leq 1$ may on the other hand be little better than just one if $y_1 \approx y_2$, not uncommon of situations where this kind of disturbance model is applied.

Thus, it is useful to be able to deal with both kinds of disturbances at the same time. The contribution of this paper is the formulation of an estimator that can deal with general state estimation problems with mixed disturbances. The optimization of the filter gain required in each step is expressed as an LMI. Since the basic structure is that of a Kalman Filter, the estimator reduces to a Kalman Filter in the case of purely stochastic disturbances.

There is much previous work for the cases of only stochastic or only set-bounded disturbances, and also some variations on mixing the two. With only stochastic disturbances, the optimal solution is the classical Kalman Filter (see [Kalman, 1960], [Kalman and Bucy, 1961]). State estimation with set bounded disturbances is considered in [Bertsekas and Rhodes, Apr 1971] and [Durieu *et al.*, 2001]. Kalman Filtering with a set-bounded initial expectation in the prior is treated in [Morrell and Stirling, 1988]. For a different approach to mixed disturbance estimation, see [Hanebeck and Horn, 2001] and references therein.

When dealing with set-bounded disturbances, there is the issue of how to represent the uncertainty sets that arise as data is combined. Unlike Gaussian noise, there is no general exact closed form representation of limited complexity. We first present the general equations, which can be used with polytopic uncertainty sets. These will how-

ever grow quickly in complexity. We will thus focus on the ellipsoidal approximation of uncertainty sets; together with a recursive formulation of the estimator this gives a fixed complexity for the estimator operations.

The rest of the paper is laid out as follows. The mixed state estimation problem to be solved is stated in section 2, including the basic estimator structure. Section 3 covers some preliminaries used in the solution. The first step of the solution is taken in section 4, which shows how to decompose the problem into the stochastic part, treated in section 5, and the set-bounded part, treated in section 6. The latter section contains the central theorem to express the set-bounded part of the filter's optimization criterion for a combination of polytopic and ellipsoidal uncertainties, which is proved in the appendix. Section 7 compares the proposed estimator with a grid based Bayesian estimator and a Kalman Filter for an example problem. Conclusions are given in section 8.

2. Problem Formulation

The objective is to perform recursive state estimation for discrete time dynamic systems modelled by

$$x_k = Ax_{k-1} + u_{k-1} + e_{k-1}^{\text{proc.}} \tag{1}$$
$$y_k = Cx_k + e_k^{\text{meas.}} \tag{2}$$

where A and C are the dynamics and measurements matrices, and the state x_k, the known control input u_k, the measurements y_k, the process disturbance $e_k^{\text{proc.}}$, and the measurement disturbance $e_k^{\text{meas.}}$ are vectors. Also A and C may be time dependent.

All error terms e^i are the sum of a stochastic term w^i and a set-bounded term δ^i,

$$e^i = w^i + \delta^i$$
$$\text{E}(w^i) = 0, \qquad \text{E}(w^i (w^i)^T) = R^i$$
$$\delta^i \in \Delta^i$$

for some positive semidefinite covariance matrix R^i and convex uncertainty set Δ^i. The stochastic terms of the process and measurement disturbance $w_k^{\text{proc.}}$ and $w_k^{\text{meas.}}$ for all times are assumed mutually uncorrelated.

Given the system above and an initial state estimate $\hat{x}_0$ with mixed error

$$e_0 = x_0 - \hat{x}_0$$

we want to form a running state estimate as a linear combination of the initial state and the measurements. The dynamics (1) are used to form the *predicted* estimate $\hat{x}_{k|k-1}$ from the previous *filtered* estimate $\hat{x}_{k-1|k-1}$:

$$\hat{x}_{k|k-1} = A\hat{x}_{k-1|k-1} + u_{k-1}. \tag{3}$$

The measurement y_k is then used to form the current filtered estimate

$$\hat{x}_{k|k} = \hat{x}_{k|k-1} + L_k \Big(y_k - C\hat{x}_{k|k-1} \Big)$$

$$= \underbrace{(I - L_k C \quad L_k)}_{X_k} \begin{pmatrix} \hat{x}_{k|k-1} \\ y_k \end{pmatrix} \tag{4}$$

using some suitable filter gain L_k. We wish to choose L_k to minimize the estimation error in some appropriate sense. The matrix X_k specifies how to weigh together the predicted state estimate and the current measurement, and represents the action of the filtering step.

3. Notation and preliminaries

The Minkowski sum of two sets X_k and Y is defined as

$$X + Y = \{x + y; x \in X, y \in Y\}.$$

Similarly, we will let the sum $X + y$ of a set X and a vector y be the translation $X + \{y\}$. The product of a set X and a matrix A will be interpreted as the element-wise product

$$AX = \{Ax; x \in X\}.$$

We will also use the product of two sets X, Y as the stacked Cartesian product

$$X \times Y = \left\{ \begin{pmatrix} x \\ y \end{pmatrix} ; x \in X, y \in Y \right\}.$$

For a matrix A, we denote by $A > 0$ $(A \geq 0)$ that A is positive (semi-)definite. For a block matrix

$$M = \begin{pmatrix} A & B \\ B^T & D \end{pmatrix}$$

with $D > 0$, the conditions that $M \geq 0$ and that the Schur Complement (see [Boyd *et al.*, 1994, ch. 2.1, pp. 7-8]) of D in M

$$\Delta = A - BD^{-1}B^T$$

is positive semidefinite, $\Delta \geq 0$, are equivalent.

4. Problem Decomposition

We begin by decomposing the problem into a stochastic and a set-bounded part. The dynamics (1) combined with the prediction (3) gives the next prediction error

$$e_{k|k-1} = Ae_{k-1|k-1} + e_{k-1}^{\text{proc.}} \tag{5}$$

while the measurement equation (2) combined with the filtering step (4) gives the next filtered error

$$e_{k|k} = X_k \begin{pmatrix} e_{k|k-1} \\ e_k^{\text{meas.}} \end{pmatrix}. \tag{6}$$

The minimization of the expected/worst-case estimation error will guide the selection of the filter gain L_k, which will then be used to update the point estimate according to (4). L_k can be optimized online, or, since it is independent of the point estimate, it can be calculated ahead of time if the disturbance characteristics are known, e.g. if they are periodic or stationary.

The estimation errors $e_{k|k-1}$ and $e_{k|k}$ are composed of a stochastic and a set-bounded part, and are formed by forming each part separately. The two parts will be coupled only in the search for the optimal filter gain L_k in the filtering step, which we find by minimizing the cost function

$$V(L) = \text{tr } W\big(R_{k|k}(L) + \alpha r(L)^2 P(L)\big) \qquad (7)$$

where $W > 0$ is a weight on the estimation error for different states, $\alpha > 0$ is the relative penalty on set-bounded error, $R_{k|k}(L)$ is the filtered error covariance, and $P_k(L)$ and $r(L)$ bound the set-bounded error after filtering $\delta_{k|k} \in \Delta_{k|k}(L)$ inside an ellipsoid:

$$\delta_{k|k}^T P(L)^{-1} \delta_{k|k} \leq r(L)^2 \qquad \forall \, \delta_{k|k} \in \Delta_{k|k}(L). \qquad (8)$$

Either P or r can be fixed for the optimization step, depending on whether we want to prespecify the shape of the ellipsoid circumscribed around $\Delta_{k|k}(L)$.

To carry out the minimization, we take the following steps:

- Form LMI conditions linear in L for

  - the stochastic part: $R \geq R_{k|k}(L)$
  - the set-bounded part: (P, r) satisfying (8)

- Minimize

$$\bar{V} = \text{tr } W(R + \alpha r^2 P)$$

  under these LMI conditions.

When we introduce ellipsoidal approximation of the set-bounded error $\Delta_{k|k}$, we will merge the prediction and filtering steps for this part to reduce conservatism.

5. Stochastic Part

We consider the update and optimization of the stochastic estimation error terms. The prediction and filtering steps (5) and (6) give the

stochastic error covariances

$$R_{k|k-1} = AR_{k-1|k-1}A^T + R_{k-1}^{\text{proc.}} \tag{9}$$

$$R_{k|k} = X_k \underbrace{\begin{pmatrix} R_{k|k-1} & 0 \\ 0 & R_k^{\text{meas.}} \end{pmatrix}}_{R_k^{\text{pm}}} X_k^T \tag{10}$$

for $w_{k|k-1}$ and $w_{k|k}$ respectively, since if $\mathrm{E}(ww^T) = R$,

$$\mathrm{E}\Big((Aw)(Aw)^T\Big) = A\,\mathrm{E}(ww^T)A^T = ARA^T.$$

The prediction step (9) is straightforward. To form an LMI for the filtering step (10), we first factor R_k^{pm} as

$$R_k^{\text{pm}} = SR^{\text{pm0}}S^T, \qquad R^{\text{pm0}} > 0.$$

By the Schur Complement, the condition $R \geq R_{k|k}$ or

$$R - X_k S R^{\text{pm0}} S^T X_k^T \geq 0$$

is then equivalent (since $R^{\text{pm0}} > 0$) to the LMI

$$\begin{pmatrix} R & X_k S \\ S^T X_k^T & (R^{\text{pm0}})^{-1} \end{pmatrix} \geq 0,$$

which is linear in L and R.

6. Set-Bounded Part

We now consider the update and optimization of the set-bounded estimation error terms. The operations are first formulated for general uncertainty sets, and then the case of ellipsoidal over-approximation is treated.

6.1 General Uncertainty Sets

From the prediction step (5), we must have $\delta_{k|k-1} \in \Delta_{k|k-1}$,

$$\Delta_{k|k-1} = A\Delta_{k-1|k-1} + \Delta_{k-1}^{\text{proc.}}.$$

If $\Delta_{k-1|k-1}$ and $\Delta_{k-1}^{\text{proc.}}$ are polytopes, so is $\Delta_{k|k-1}$.

For the filtering step, we have

$$\delta_{k|k} = X_k \underbrace{\begin{pmatrix} \delta_{k|k-1} \\ \delta_k^{\text{meas.}} \end{pmatrix}}_{\delta_k^{\text{pm}}}.$$

The constraint (8) can be expressed for any $\delta_k^{\text{pm}} \in \Delta_k^{\text{pm}} = \Delta_{k|k-1} \times \Delta_k^{\text{meas.}}$ as a second order cone constraint when P is fixed:

$$r \geq ||P^{-\frac{1}{2}}\delta_{k|k}|| = ||P^{-\frac{1}{2}}X_k\delta_k^{\text{pm}}||$$

or in general by the Schur Complement (since $P > 0$) as an LMI

$$r^2 - (\delta_k^{\text{pm}})^T X_k^T P^{-1} X_k \delta_k^{\text{pm}} \geq 0$$
$$\iff \begin{pmatrix} P & X_k\delta_k^{\text{pm}} \\ (\delta_k^{\text{pm}})^T X_k^T & r^2 \end{pmatrix} \geq 0.$$

If Δ_k^{pm} is a polytope, it is enough to consider the constraint at the vertices, since an ellipsoid contains a set of vertices iff it contains the convex hull of those vertices (the polytope).

6.2 Ellipsoidal Uncertainty Sets

Now suppose that the filtered set-bounded error from the previous step $\Delta_{k-1|k-1}$, and possibly the process or measurement disturbance parts $\Delta_{k-1}^{\text{proc.}}$ and $\Delta_k^{\text{meas.}}$, are described by ellipsoids. In this case we can use the ellipsoid (8) to find an ellipsoidal over-approximation for $\Delta_{k|k}$ to use in the next step. To formulate (8) as an LMI in this case, we need the following theorem.

THEOREM 1—ELLIPSOID BOUNDING WEIGHTED ELLIPSOID SUM
Given a number of ellipsoids $\mathcal{E}_i, i = 1 \ldots n$:

$$z_i \in \mathcal{E}_i \iff \begin{cases} z_i = G_i x_i + b_i \\ x_i^T Q_i x_i \le r_i^2 \end{cases}$$

the weighted Minkowski sum

$$\mathcal{A} = X \sum_i \mathcal{E}_i = \left\{ x = X z; z = \sum_i z_i, z_i \in \mathcal{E}_i \, \forall i \right\}$$

can be proved by the S-procedure (see [Boyd *et al.*, 1994, ch. 2.6.3, pp. 23-24]) to be contained in the centered target ellipsoid $\mathcal{E}$,

$$x \in \mathcal{E} \iff x^T P^{-1} x \le r^2 \tag{11}$$

iff the LMI condition

$$\begin{pmatrix} P & XG & Xb \\ G^T X^T & Q\tau & \\ b^T X^T & & r^2 - \sum_i \tau_i r_i^2 \end{pmatrix} \ge 0 \tag{12}$$

is satisfied for some scalars $\tau_i \ge 0$, where $b = \sum_i b_i$, and

$$G = (\, G_1 \quad G_2 \quad \ldots \quad G_n \,), \qquad Q_\tau = \mathrm{diag}\big(\{\tau_i Q_i\}_i\big).$$

If $n = 1$ and $r_1 > 0$, the condition (12) is also necessary for $\mathcal{A} \subseteq \mathcal{E}$. $\quad\square$

Proof: See the appendix.

Using the theorem. We let $P = P$ and $z = \delta_k^{\mathrm{pm}}$, where Δ_k^{pm} is a sum of ellipsoids. With one centered ellipsoid ($b_i = 0$) containing each of the previous filtered error, the process and measurement disturbances:

$$\Delta_{k-1|k-1} \subseteq \mathcal{E}_1, \qquad \Delta_{k-1}^{\mathrm{proc.}} \subseteq \mathcal{E}_2, \qquad \Delta_k^{\mathrm{meas.}} \subseteq \mathcal{E}_3$$

the set-bounded part gets the prediction step $\Delta_{k|k-1} \subseteq A\mathcal{E}_1 + \mathcal{E}_2$ and the filtering step

$$\Delta_{k|k} \subseteq X_k(\Delta_{k|k-1} \times \mathcal{E}_3) \subseteq X_k\big((A\mathcal{E}_1 + \mathcal{E}_2) \times \mathcal{E}_3\big).$$

The ellipsoid sum for $\Delta_{k|k}$ can thus be expressed with the theorem, plugging in the ellipsoids $\mathcal{E}_1, \mathcal{E}_2, \mathcal{E}_3$, and

$$G_1 = \begin{pmatrix} A \\ 0 \end{pmatrix}, \quad G_2 = \begin{pmatrix} I \\ 0 \end{pmatrix}, \quad G_3 = \begin{pmatrix} 0 \\ I \end{pmatrix}.$$

Thus we can use the LMI condition (12) to circumscribe an ellipsoid around $\Delta_{k|k}$.

Variations.   We can use more or fewer ellipsoidal terms for the uncertainty sets Δ_i, and also polytopic terms. For polytopic terms, the sum $\mathcal{P}$ of all such terms is first formed. As in the case with only polytopic terms, the LMI must be written once for each vertex of $\mathcal{P}$. If $\mathcal{P}$ is symmetric, we need only write half as many LMI:s since the centered target ellipsoid $\mathcal{E}$ sees no difference between the vertices v and $-v$. A polytope vertex can be represented by a zero-dimensional ellipsoid with $b_i \neq 0$.

A polytope that is the sum of one-dimensional polytopes (line segments) may expressed more economically as a sum of one-dimensional ellipsoids. However, the result may be more conservative since forming the sum of ellipsoids relies on the S-procedure.

The use of both P and r as variables in the condition (11) for the target ellipsoid may seem redundant, but it allows to state a possibly simpler optimization problem if the shape of the target ellipsoid is fixed. (I.e. to some shape desired in a stationary situation.) It is of course possible to constrain P to other spaces than to be fully free or with a prespecified shape. Another use for r could be to improve the numerical conditioning of the optimization problem by guessing the size of the resulting ellipsoid before optimizing for P.

7. Simulations

7.1 Example System

Consider a double integrator process with dynamics

$$x_{k+1} = \underbrace{\begin{pmatrix} 1 & h \\ 0 & 1 \end{pmatrix}}_{A} x_k + \underbrace{\begin{pmatrix} \frac{1}{2}h^2 \\ h \end{pmatrix}}_{B} u_k + w_k^{\text{proc.}}$$

$$\mathrm{E}\big(w_k^{\text{proc.}}\big) = 0, \quad \mathrm{E}\Big(\big(w_k^{\text{proc.}} w_k^{\text{proc.}}\big)^T\Big) = \frac{1}{4}\underbrace{\begin{pmatrix} \frac{1}{3}h^3 & \frac{1}{2}h^2 \\ \frac{1}{2}h^2 & h \end{pmatrix}}_{R_k^{\text{proc.}}}$$

where $h = 0.1$ is the sample time, $(x_k)_1$ is the position and $(x_k)_2$ the velocity. White process noise enters along with the control acceleration u_k.

The measurements are coarsely quantized:

$$y_k = \mathrm{round}\big(Cx_k\big), \quad C = \begin{pmatrix} 1 & 0 \end{pmatrix},$$

where $\mathrm{round}(x)$ rounds x to the nearest integer. Using the current framework, we can model the measurement by

$$y_k = Cx_k + \delta_k^{\text{meas.}}, \quad \delta_k^{\text{meas.}} \in \Delta_k^{\text{meas.}} = [-\tfrac{1}{2}, \tfrac{1}{2}].$$

With the sampling time h small enough, we may consider $(x_k)_1$ to be almost completely known at all *events*, when y_k changes value. This measurement may be modelled as

$$\frac{1}{2}\big(y_k + y_{k-1}\big) = Cx_k + w_k^{\text{meas.}},$$
$$\mathrm{E}\big(w_k^{\text{meas.}}\big) = 0, \quad \mathrm{E}\Big(\big(w_k^{\text{meas.}} w_k^{\text{meas.}}\big)^T\Big) = R_k^{\text{meas.}}, \tag{13}$$

where $R_k^{\text{meas.}}$ gives a suitable approximation of the error in the guess $Cx_k \approx \frac{1}{2}\big(y_k + y_{k-1}\big)$. We take $R_k^{\text{meas.}} = (R_k^{\text{proc.}})_{11}$.

Since the system is unstable, we stabilize it with the control law

$$u_k = -\begin{pmatrix} 1 & 2 \end{pmatrix}\hat{x}_k,$$
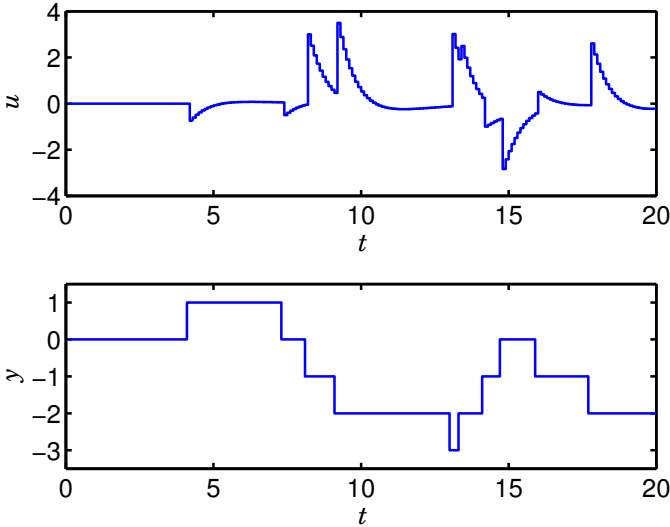
Figure 1.   Test sequence for the observers

which places the poles in approximately $z = e^{-h}$. The state estimate $\hat{x}_k$ is taken from a simple heuristic state estimator that:

- runs in open loop between events

- updates at events:

$$(\hat{x}_k)_1 = \frac{1}{2}\left(y_k + y_{k-1}\right)$$

$$(\hat{x}_k)_2 = \frac{(\hat{x}_k)_1 - (\hat{x}_{k_{\text{last}}})_1}{h(k - k_{\text{last}})}$$

where k_{last} is the time index of the last event or known initial state.

The process was simulated with the heuristic controller to produce the test sequence u_k, y_k in Fig. 1. The corresponding state sequence x_k can seen in Fig. 2. (together with state estimates from different estimators)
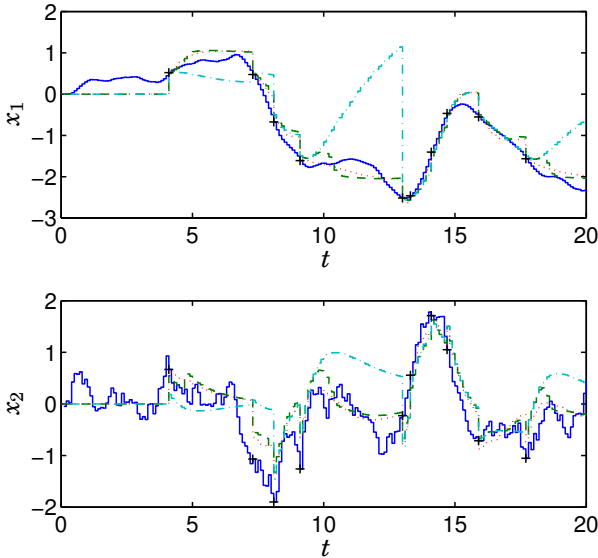
Figure 2. Actual states and state estimates generated by the observers. Actual states (solid), Mixed Estimator (dashed), Grid Filter (dotted), Kalman Filter (dash-dotted). Events are marked with a + sign.

7.2 Estimator Implementation For The Example

In this example, the process noise is purely stochastic, and the set-bounded measurement error $\Delta_k^{\text{meas.}}$ can be represented as an interval symmetric around the origin, so the target ellipsoid $\mathcal{E} \supseteq \Delta_{k|k}$ should enclose the sum of an ellipsoid for $\Delta_{k|k-1}$ and the polytope for $\Delta_k^{\text{meas.}}$. Since we have only one ellipsoid in the sum, (12) is both necessary and sufficient for the target ellipsoid $\mathcal{E}$ to enclose it. Since the polytope $\Delta_k^{\text{meas.}}$ is symmetric with two vertices, we need only one instance of the LMI condition (12).

Table 1.   Mean quadratic errors over a 10^5 time step test sequence.

E_{Mixed}	E_{Grid}	E_{Kalman}
$\begin{pmatrix} 0.054 & 0.063 \\ 0.063 & 0.180 \end{pmatrix}$	$\begin{pmatrix} 0.045 & 0.053 \\ 0.053 & 0.157 \end{pmatrix}$	$\begin{pmatrix} 0.444 & 0.223 \\ 0.223 & 0.285 \end{pmatrix}$

7.3 Performance Comparison

Three filters were compared on the test sequence:

- The Mixed Estimator proposed in this paper using ellipsoidal over-bounding of $\Delta_{k|k}$ in each step, with

$$\alpha = 1, \qquad W = \begin{pmatrix} 1 & -0.3 \\ -0.3 & 0.4 \end{pmatrix}.$$

  The weight matrix W was chosen by letting W^{-1} be roughly proportional to the error covariance of the Grid Filter (see below) a long time after an event.

- A *Grid Filter*; a discretization of the Bayesian Estimator for the system (with approximately 32 000 states). See [Henningsson and Åström, 2006] for more about the Bayesian Estimator for this system.

- A Kalman Filter that uses only the measurements (13) at events, and runs in open loop in between.

Table 1 shows the average estimation error of the filters over a test sequence of 10^5 time steps, evaluated as

$$E = \frac{1}{N} \sum_{k=1}^{N} (x_k - \hat{x}_k)(x_k - \hat{x}_k)^T.$$

The Mixed Estimator is seen to come quite close to the Grid Filter performance, but the Kalman Filter is far behind. Fig. 2 shows actual state trajectories together with the estimates. Events are marked with $+$ signs. When events are frequent, all estimators seem to follow the state trajectories reasonably well, especially for the position x_1. When
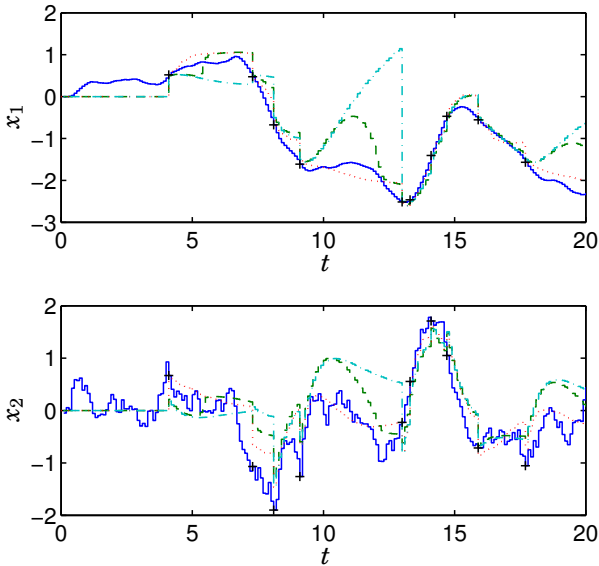
Figure 3.   Actual states and state estimates, with $\alpha = 10$ for the Mixed Estimator, which makes it follow the Kalman Filter for too long.

there is longer time between events, the Kalman Filter seems to lose track. The Mixed Filter is much better at following the Bayesian estimate. The strategy it uses seems to be something like:

- At an event, update the state estimate.

- Continue by open loop predictions some time after each event, while the prediction error is small.

- When the prediction error becomes too large, start to incorporate the imprecise measurements available.

Fig. 3 shows the same simulation with $\alpha = 10$ for the Mixed Estimator. The weight α adjusts the tradeoff between stochastic and set-bounded estimation error. With higher α it is seen that the Mixed Filter waits longer to incorporate the uncertain measurements after each events. The value $\alpha = 1$ used in Fig. 2 seems to give a more reasonable tradeoff.
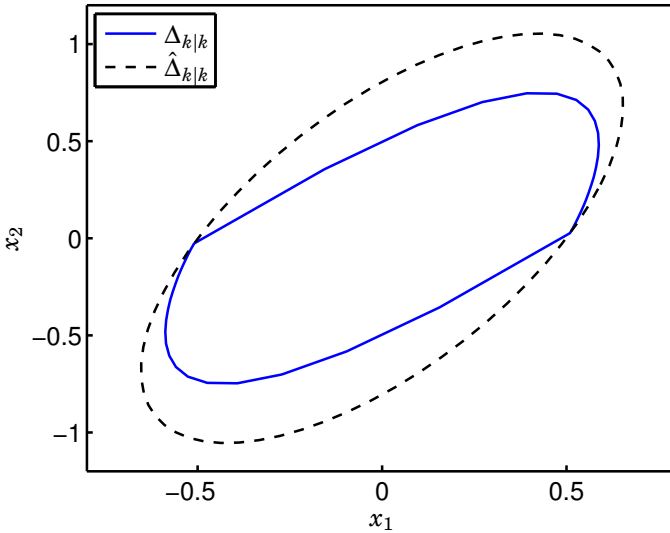
Figure 4.   Actual set bounded error and ellipsoidal approximation used by the Mixed Filter at $t = 12.9$, just before an event.

The uncertainty set $\Delta_{k|k}$ (a polytope in this example) and the recursive ellipsoidal over-approximation $\hat{\Delta}_{k|k}$ used by the mixed filter can be seen in Fig. 4, just prior to the event at $t = 13$. The actual set takes up perhaps $\frac{2}{3}$ of the ellipsoid's volume, and that they more or less touch at the sharpest corners of the polytope.

8. Conclusion

This paper describes the design of a state estimator for linear systems with process and measurement disturbances containing both stochastic and set bounded terms. The estimator structure that is borrowed from the Kalman Filter is optimal for purely stochastic disturbances, and allows the two parts of the estimation error to be treated efficiently and almost independently. The filter gain is optimized by solving a Linear Matrix Inequality (LMI) problem.

The estimator can value the usefulness of measurements corrupted by different amounts of stochastic and set bounded disturbances, with a parameter α that can be used to tune the tradeoff between the two kinds of error. An example shows that the estimator performs quite close to an optimal Bayesian Estimator, and that α can be used to adjust how long to wait after receiving a good measurement before incorporating measurements with interval uncertainty.

The estimator reproduces the behavior of the Kalman Filter with set-bounded initial expectation in [Morrell and Stirling, 1988] under the circumstances assumed in that work, when the weight α goes to zero. When α is nonzero, the estimator applies a higher filter gain to eliminate the set-bounded uncertainty faster.

An open issue is how to choose the state weighting matrix W in a systematic fashion.

Acknowledgments

References

Bertsekas, D. and I. Rhodes (Apr 1971): "Recursive state estimation for a set-membership description of uncertainty." *Automatic Control, IEEE Transactions on*, **16:2**, pp. 117–128.

Boyd, S., L. El Ghaoui, E. Feron, and V. Balakrishnan (1994): *Linear Matrix Inequalities in System and Control Theory*, vol. 15 of *Studies in Applied Mathematics*. SIAM, Philadelphia, PA. Also available at http://www.stanford.edu/~boyd/lmibook/.

Durieu, C., E. Walter, and B. Polyak (2001): "Multi-input multi-output ellipsoidal state bounding." *Journal of Optimization Theory and Applications*, **111**, November, pp. 273–303(31).

Hanebeck, U. and J. Horn (2001): "New estimators for mixed stochastic and set theoretic uncertainty models: the general case." *American*

Control Conference, 2001. Proceedings of the 2001, **6**, pp. 5040–5045 vol.6.

Henningsson, T. and K. J. Åström (2006): "Log-concave observers." In *Proceedings of the 17th International Symposium on Mathematical Theory of Networks and Systems*. Kyoto, Japan.

Kalman, R. E. (1960): "A new approach to linear filtering and prediction problems." *Transactions of the ASME–Journal of Basic Engineering*, **82:Series D**, pp. 35–45.

Kalman, R. E. and R. S. Bucy (1961): "New results in linear filtering and prediction theory." *Transactions of the ASME - Journal of Basic Engineering*, **83**, pp. 95–107.

Morrell, D. and W. Stirling (1988): "Set-valued filtering and smoothing." *Twenty-Second Asilomar Conference on Signals, Systems and Computers, 1988.*

A. Proof of theorem 1

This development is based on [Boyd *et al.*, 1994, ch. 3.7.4, pp. 46-47]. The construction is extended to be linear in the transformation X, to handle ellipsoids that are flat in some dimensions, and to specify the centers b_i separately, but is reduced in that we are only interested in centered target ellipsoids $\mathcal{E}$.

To handle the Minkowski sum of ellipsoids, we need a condition for when one ellipsoid contains the intersection of a number of ellipsoids. Given a set of quadratic functions $\{f_i(x)\}_i, i = 1 \ldots n$, one sufficient condition to verify that a quadratic function $f(x) \geq 0$ whenever all $f_i(x) \geq 0$ is given by the S-procedure:

$$\exists \tau_i \geq 0, i = 1 \ldots n : \quad f(x) \geq \sum_i \tau_i f_i(x) \quad \forall x.$$

The condition is also necessary e.g. when $n = 1$ and $f_1(x) > 0$ for some x, see [Boyd *et al.*, 1994, ch. 2.6.3, pp. 23-24].

The condition (12) which we seek to derive is formed by first constructing an extended space where each term of the ellipsoid sum has

its own coordinates, and forming the set where all coordinates are within their respective ellipsoids, which is the intersection of ellipsoidal cylinders. We then used the S-procedure to circumscribe an ellipsoidal cylinder parametrized in the sum coordinates.

Let

$$x^T = (\, x_1^T \quad x_2^T \quad \ldots \quad x_n^T \,), \qquad z = \sum_i z_i.$$

Then, according to the definitions in the theorem,

$$z = Gx + b = \underbrace{(\, G \quad b \,)}_{G_e} \underbrace{\begin{pmatrix} x \\ 1 \end{pmatrix}}_{x_e} = G_e x_e.$$

We take the first step of the S-procedure (using $\tau_i \geq 0 \; \forall i$) by forming the condition

$$\sum_i \tau_i (r_i^2 - x_i^T Q_i x_i) = \left(\sum_i \tau_i r_i^2 \right) - x^T Q_\tau x \geq 0 \tag{14}$$

which will always be fulfilled when $z_i \in \mathcal{E}_i \; \forall i$.

The condition for the target ellipsoid, $x \in \mathcal{E}$, $x = Xz = X G_e x_e$ is equivalent to

$$r^2 - x_e^T G_e^T X^T P^{-1} X G x_e \geq 0. \tag{15}$$

Subtracting (14) from (15), we form our S-procedure condition, which can clearly only be fulfilled for all x if (15) is fulfilled whenever (14) is:

$$x_e^T \left(\underbrace{\begin{pmatrix} Q_\tau & \\ & r^2 - \sum_i \tau_i r_i^2 \end{pmatrix}}_{Q_e} - G_e^T X^T P^{-1} X G_e \right) x_e \geq 0.$$

As we assume x to be arbitrary, we might as well assume x_e to be arbitrary since scaling of x_e with a nonzero constant does not affect whether the condition holds. The case when the last entry of x_e is zero is approached when $||x|| \to \infty$. Thus we can equivalently consider positive semidefiniteness of the matrix that stands between x_e^T and x_e above.

By the Schur Complement, since $P^{-1} > 0$, this condition is equivalent to

$$\begin{pmatrix} P & XG_e \\ G_e^T X^T & Q_e \end{pmatrix} \geq 0,$$

which is exactly (12).

Paper V

Comparison of LTI and Event-Based Control for a Moving Cart with Quantized Position Measurements

Toivo Henningsson   Anton Cervin

Abstract

Traditional linear time-invariant (LTI) control design assumes that measurements are taken at regular time intervals and have independent additive noise. A common practical case that violates this assumption is the use of encoders that give quantized position measurements; when the quantization is appreciable the measurement noise is far from LTI. This paper develops a simple event-based controller based on simplifying a joint maximum a posteriori estimator, which is applied to a moving cart with quantized position measurements. The payoff for implementing the somewhat more complex event-based controller is to drastically reduce the effect of quantization noise in the experiments. A sequence of simpler (LTI) to better adapted controllers are described and compared according to experimental performance and implementation complexity. Implementation issues on the microcontroller are discussed.

1. Introduction

The majority of all feedback controllers today are implemented using computers, relying on periodic sampling, computation, and actuation. For linear time-invariant (LTI) systems, sampled-data control theory [Åström and Wittenmark, 1997] provides powerful tools for direct digital design, while implementations of nonlinear control designs tend to rely on discretization combined with fast periodic sampling.

There are however situations where it could be advantageous to use other activation schemes. For first-order linear stochastic systems, it has been shown that event-triggered sampling can provide better regulation performance and/or lower average activation rates than time-triggered sampling [Åström and Bernhardsson, 1999; Henningsson *et al.*, 2008]. This can be useful for networked embedded control systems with constrained communication, computation, or energy resources. With similar arguments, heuristic event-based PID controllers have been proposed in [Årzén, 1999; Vasyutynskyy and Kabitzsch, 2007].

Another motivation for event-based control are systems where the events are inherent in the physics. Examples include wheel encoders and accelerometers that deliver pulse trains rather than continuous measurement signals. Previous case studies have shown that accurate control can be accomplished even with very low-resolution encoders if the controller is activated at measurement events rather than at regular time intervals [Sandee *et al.*, 2007].

In this paper, we study the practical problem of implementing a velocity control system for a moving cart using a low-resolution position encoder and a low-end 8-bit microcontroller. Each time the quantized position measurement changes value is considered an *event*, to be given special consideration by the controller. Since the friction is appreciable and varying, we want to utilize also the information contained in *the absence of events*.

It is well known that the problem of optimal estimation and control with quantized measurements is extremely difficult [Curry, 1970]. For instance, there is no separation theorem in the general case. Even the pure estimation problem is computationally intractable and essentially requires the on-line solution of a partial differential equation.

Seeking simpler, sub-optimal solutions, we start from a joint max-

imum a posteriori (JMAP) estimator [Cox, 1964]. This formulation is powerful enough to model quantized measurements, yet yields a tractable problem. The estimator is greatly simplified and adapted to the control problem at hand in order to be implementable on the small microcontroller. The final controller is based on periodic state feedback from an event-based observer implemented using fast sampling. The state feedback design can be reused from the LTI controllers designed for comparison, since the practical challenge lies in state estimation.

The rest of the paper is laid out as follows. The setup is explained in Section 2. Section 3 attempts a first control design using LTI methods, giving important insight into the tradeoffs involved. The JMAP estimator is introduced in Section 4 as a systematic means for state estimation with quantized measurements and is simplified in Section 5 into something that can run online. Microcontroller implementation issues are described in Section 6. Section 7 compares different LTI and event-based controllers experimentally. The conclusions are given in Section 8.

2. Setup

2.1 The Moving Cart

The process is a moving cart driven by a DC motor, see Fig. 1. The control signal is the motor voltage, governed by a direction bit and a 29 kHz PWM signal. A rotary encoder on the motor axis is used for position sensing. The encoder output is two square waves as a function of the position, 90° out of phase. The measurements can be modeled as

$$y = \Delta p_{\text{quant.}} \cdot \text{round} \left(\frac{p}{\Delta p_{\text{quant.}}} \right), \ \Delta p_{\text{quant.}} = 5 \cdot 10^{-5} \, \text{m}, \qquad (1)$$

where p is the position of the cart. Lower encoder resolutions can be emulated in software; most tests are run with $\Delta p_{\text{quant.}}$ emulated to 3.2 mm.

The cart is equipped with an ATmega16 8-bit AVR microcontroller clocked at 14.7 MHz, which handles encoder sampling, motor drive, filtering, control, and communication with a PC over a serial link.
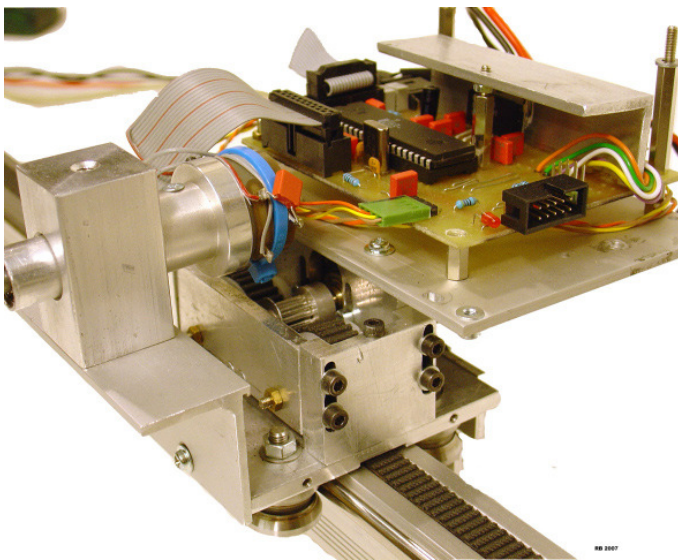
Figure 1.   The moving cart with the microcontroller mounted on top. The cart can move a little more than one meter along the track.

2.2 Process Model

A simple dynamical model for the cart was postulated as

$$\begin{pmatrix} \dot{p} \\ \dot{v} \end{pmatrix} = \begin{pmatrix} 0 & 1 \\ 0 & -T_d^{-1} \end{pmatrix} \begin{pmatrix} p \\ v \end{pmatrix} + \begin{pmatrix} 0 \\ 1 \end{pmatrix} (u + u_{\text{bias}}),$$

where p is the position, v is the velocity, $u = K_u u_V$ is the control signal, u_V is the control signal in units of full motor voltage, u_{bias} is the disturbance from friction and other sources, and K_u and T_d are process parameters. A sequence of step response experiments were made and the parameters estimated by linear regression to

$$T_d = 0.23\,\text{s}, \qquad K_u = 25.5\,\text{m/s}^2.$$

The disturbance u_{bias} was assumed constant within each step response.
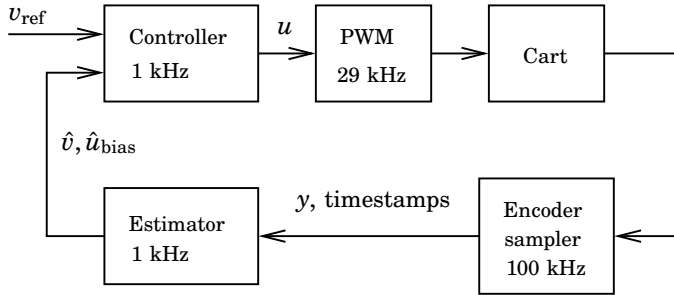
128

Figure 2. The implementation structure. The rate of the encoder sampler is set to keep up with cart speeds of 5 m/s.

As the controllers to be designed will have cross-over frequency $\omega_c \geq 5$ rad/s, T_d^{-1} is approximated to 0 for simplicity. With this approximation the process is a pure double integrator $G_p(s) = \frac{1}{s^2}$, and the cross-over frequency of a control design can be adjusted by just changing the time scale. The time scale of the process is adjusted by additional gain in the controller.

2.3 Control Objectives

We will design velocity controllers for the moving cart. The objectives are:

- Fast reference tracking and attenuation of process disturbances (friction).

- Low noise in the control signal.

- Reasonable robustness.

Reasonable robustness is an absolute demand. Given this constraint, the controllers should try to optimize for the first two objectives, using the cross-over frequency ω_c to adjust the tradeoff.

2.4 Implementation Structure

An overview of the implementation is given in Fig. 2. A multi-rate structure is used, where fast sampling at 100 kHz is used to read the

encoder, while the control output is generated at 1 kHz. For simplicity, the implementation is not event-triggered *per se*; rather, the estimator may use time stamps of the latest measurement events (i.e., changes in encoder value) when forming its estimate.

3. LTI Control Design

In this section, we design two PI controllers under simplifying assumptions. The design is carried out in continuous time, ignoring the fact that the measurement disturbance comes from quantization. Still, the design provides important insight into robustness issues.

3.1 Pure PI Controller

To do velocity control, we need to estimate the cart velocity. We use a simple first-order filter on the position output:

$$\hat{V} = \frac{sY}{sT_{\text{filter}} + 1}. \tag{2}$$

A PI controller

$$U = K(\beta V_{\text{ref}} - \hat{V}) - K\frac{\hat{V} - V_{\text{ref}}}{sT_i} \tag{3}$$

gives the following controller transfer function from $-y$ to u:

$$G_c(s) = \frac{K}{T_i} \frac{sT_i + 1}{sT_{\text{filter}} + 1}$$

This is in effect a lead filter that lifts the phase of the loop gain around the cross-over frequency ω_c above the constant phase $G_p(i\omega) = -180°$ of the process.

We want to place the zero and pole far apart to get a large phase margin, but on the other hand we want short T_i and long T_{filter} for good rejection of process disturbances and measurement noise. The best trade-off is achieved by placing the zero and pole on either side of the cross-over frequency ω_c at equal logarithmic distance,

$$T_{\text{filter}} = r^{-1}\omega_c^{-1}, \quad T_i = r\omega_c^{-1},$$
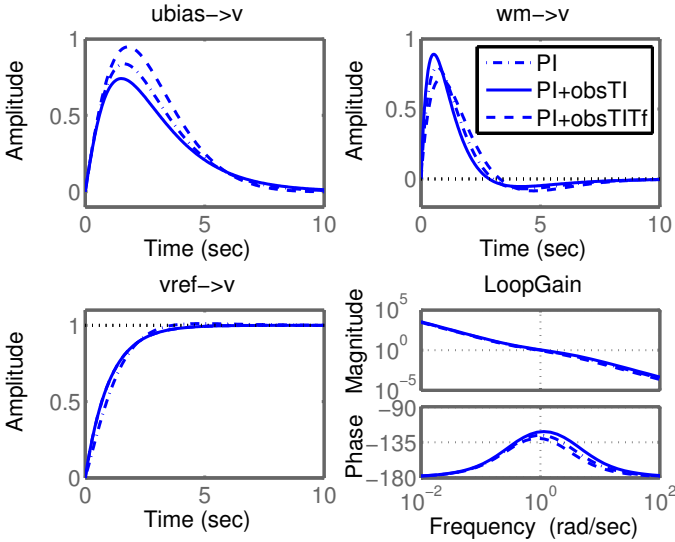
Figure 3. Comparison of the PI designs: simulated step response from u_{bias}, measurement noise w_m, and v_{ref} to v, and loop transfer functions.

where $r > 1$ determines the phase margin ϕ_m. We take $\phi_m \approx 50° \implies r \approx 3$ as a reasonable compromise between robustness and disturbance rejection.

The cross-over frequency ω_c is left as a design parameter, to be varied in the experiments. The loop transfer function for $\omega_c = 1$ and $\phi_m = 51°$ can be seen in Fig. 3. The reference weighting parameter $\beta = 0.5$ is used to eliminate overshoot in the reference step response.

3.2 Observer-Based PI Controller

The standard PI controller makes no use of the process model. To exploit our process knowledge, we can instead use the control law

$$U = \underbrace{K(V_{\text{ref}} - \hat{V})}_{U_P} - \underbrace{K\frac{sY - \hat{V}}{sT_i}}_{\hat{U}_{\text{bias}}}, \tag{4}$$

where the I-part now only integrates the difference between actual and estimated velocity, and the velocity estimate

$$\hat{V} = \frac{sY + T_{\text{filter}}U_P}{sT_{\text{filter}} + 1} \tag{5}$$

includes feed forward from the P-part. Since the I-part provides the bias estimate, it should not be fed forward into $\hat{V}$. Reacting only to differences between the model prediction and measurements, the I-part will no longer wind up during reference steps, which eliminates the need for the β tuning parameter. Using $U + \hat{U}_{\text{bias}}$ instead of U_P for feed forward in the velocity filter, this disturbance estimation scheme also provides anti-windup.

With the observer-based control law, the controller transfer function becomes

$$G_c(s) = \frac{K}{T_i} \frac{s(T_i + T_{\text{filter}}) + K T_{\text{filter}}}{sT_{\text{filter}} + 1 + K T_{\text{filter}}},$$

i.e., a lead filter with slower zero and faster pole than for the pure PI controller. The observer gives extra phase lead around the cross-over frequency, which can be exploited by more aggressive tuning. We take

- $T_i' = T_i/4$ for improved disturbance rejection.

- $T_i' = T_i/2, T_{\text{filter}}{}' \approx 2T_{\text{filter}}$ for improved measurement noise rejection. This is useful for the PI controller since it is bad at handling the measurement quantization. We do not want to make T_{filter} slower than ω_c^{-1}, since this would impede process disturbance rejection.

Fig. 3 compares the original PI designs and the two versions with observer. It is seen that the original design is somewhere in the middle, while $T_i' = T_i/4$ gives slightly faster disturbance rejection and $T_{\text{filter}}{}' = 2T_{\text{filter}}$ gives a smaller response to measurement disturbances. The observer-based controllers respond slightly faster to v_{ref} changes, since the P-part gets to act on the full reference step.

As is seen in the figure, the pure and observer-based PI controllers can be tuned to more or less the same behaviour in closed loop. The real gain of using an observer will be evident when we introduce the event-based controller, where the control runs in open loop as long as the measurements do not contradict the observer's predictions.

4. The JMAP Estimator

A problem with the PI controller is that it does not exploit the fact that the main source of measurement noise comes from quantization. In this section, we explore how to model quantization in the state estimator, and the properties that follow. Insight into the behavior of the estimator will be used to simplify it in the next section.

4.1 Process Model

Consider a system in discrete time,

$$x(k+1) = Ax(k) + Bu(k) + w(k), \tag{6}$$

where x is the state, u is the control signal, and w is a zero-mean white Gaussian noise process with variance R. The available measurements specify an interval for the output at each sample:

$$y(k) - \Delta y \leq Cx(k) \leq y + \Delta y. \tag{7}$$

The initial state may be fully known or Gaussian distributed.

4.2 The Estimation Problem

We consider the joint maximum a posteriori (JMAP) approach to state estimation: find the most probable trajectory of the state x conditioned on the measurements, and use the state at the current time as state estimate. With additive Gaussian measurement noise, this approach yields the Kalman Filter (see [Cox, 1964]). With the quantized measurements (7), the solution is a bit more complex.

The log-likelihood $l(x)$ of a state trajectory $x(k), k_0 \leq k \leq k_1$, considering the dynamics (6) and initial distribution of the state is given by

$$l(x) = -\frac{1}{2} \left(||x(k_0) - x_0||^2_{R_0^{-1}} + \sum_{k=k_0}^{k_1-1} ||w(k)||^2_{R^{-1}} \right), \tag{8}$$

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

133

when $x(k_0)$ has a Gaussian distribution with mean x_0 and variance R_0, and where $||x||_R^2 = x^T R x$ and $w(k)$ has been solved for from (6). The most probable state trajectory x is found by maximizing $l(x)$ subject to the linear measurement constraints (7) and any known initial conditions. This is a quadratic program, which can be solved reasonably fast on a PC. Ideally, the history $k_0 \leq k \leq k_1$ should go as far back as possible to use all measurements in the estimation. An approach often used in practice is to fix $k_1 - k_0 = \Delta k$, resulting in *moving horizon estimation*, see [Rawlings and Bakshi, 2006].

At any sample k, the constraint (7) is considered *active* if the optimal trajectory would be different without it. If it is known which constraints are active, the optimal solution can be obtained by fixing the constrained variables at their constraints and optimizing freely over the rest. This is the same solution as we would get from a Kalman Filter when the available measurements are perfect position measurements at the active constraints.

4.3 Time Update

When moving forward one sample to $k = k_1 + 1$ without adding new measurements, the estimate update is very simple. Suppose that there is a unique optimal state trajectory. The new cost term in (8) will add no cost iff $w(k_1) = 0$, i.e. the next state is predicted by the dynamics (6) with the disturbance set to zero.

4.4 Measurement Update

If the constraint (7) from the new measurement agrees with the current state estimate, i.e. if the trajectory from the time update is still feasible, it is also still optimal. Otherwise, the estimator must add the most probable correction to the trajectory so as to satisfy (7), over the entire horizon. This correction will move the estimated position the shortest distance necessary, i.e. to the constraint.

The most influential measurements will be at *events*, when the quantized position measurement changes value. The position is then known to be exactly half way between the two quantization levels some time during the short period between the two measurements.

5. Simplified Event-Based Estimator

Although the JMAP estimation problem is tractable, it is far too demanding to solve in real time on a small microcontroller. In this section, we derive a realistic estimator by simplifying the JMAP estimator. The key simplifications are:

- In the JMAP approach, u_{bias} would be a state variable. We assume that it varies slowly, and can be estimated separately from p and v.

- Active position measurement constraints are considered only at the current time and last event, which is considered as a known position at a known time.

- When applying measurement updates, it is preferred to change the state estimate as little as necessary. This rule is needed since the simplified estimation problem would otherwise be underdetermined.

The states of the estimator will be estimates of the current state $\hat{p}$, $\hat{v}$, and $\hat{u}_{\text{bias}}$ together with a history described by the time of the last event.

5.1 Time Update

As in the JMAP case, the time update is simply the dynamics of the process model without noise

$$\begin{pmatrix} \hat{p}(k+1) \\ \hat{v}(k+1) \end{pmatrix} = \underbrace{\begin{pmatrix} 1 & h \\ 0 & 1 \end{pmatrix}}_{A} \begin{pmatrix} \hat{p}(k) \\ \hat{v}(k) \end{pmatrix} + \underbrace{\begin{pmatrix} \frac{1}{2}h^2 \\ h \end{pmatrix}}_{B} \left(u(k) + \hat{u}_{\text{bias}}(k) \right). \quad (9)$$

There is no systematic drift of u_{bias} in the model, so $\hat{u}_{\text{bias}}$ is not changed in the time update. There is no reason to believe that u_{bias} has changed unless indicated by the measurements.

5.2 Measurement Update

When the current position measurement y disagrees with the current position estimate $\hat{p}$, i.e. y differs from the quantization of $\hat{p}$ according

to (1), a measurement update is applied. The most probable cause of estimation error is taken to be an error Δv in the velocity estimate at the last event—the only error that requires no disturbance after the last event to explain it. By superposition, this leads to a velocity error Δv and a position error $\Delta p = \Delta v \Delta t$ at the current time, where Δt is the time since the last event. Adjusting the position estimate to lie at the constraint $\hat{p} = p$, i.e. to just agree with the measurement y, the measurement update becomes

$$
\begin{aligned}
\hat{p}^+ &= p = \hat{p} + \Delta p, \\
\hat{v}^+ &= \hat{v} + \frac{\Delta p}{\Delta t}.
\end{aligned}
\tag{10}
$$

As in the PI controller with observer, $\hat{u}_{\text{bias}}$ is formed from the time integral of the difference between estimated and measured velocity, i.e., it accumulates the difference between estimated and measured position. The measurement update for $\hat{u}_{\text{bias}}$ thus becomes

$$
\hat{u}_{\text{bias}}^+ = \hat{u}_{\text{bias}} + \frac{K}{T_i}\Delta p,
\tag{11}
$$

in analogy to $\hat{U}_{\text{bias}} = \frac{K}{T_i}(Y - \frac{1}{s}\hat{V})$ in (4).

5.3 Fixes

The gains from position error Δp to velocity and disturbance corrections Δv and Δu_{bias} given above usually work well. Since the estimator is a considerable simplification and because of some unmodeled effects, there is a need to fix some corner cases.

LTI Mode Timeout. When there is a long time between events, a position error only indicates a small velocity correction, and the gain from Δp to Δv goes down (as Δt^{-1}). This works as intended for steady motion.

The low gain should no longer be used, however, when the position errors become big, e.g. when the cart is stuck due to friction. The estimator time update will predict a high velocity due to prolonged control signal activity, but the measurement update should actually keep $\hat{v}$ down.

To handle this case, a timeout of $T_{\text{timeout}} = 2T_{\text{filter}}$ is used. If measurement constraints have been active for the last T_{timeout} time, the time constant $\Delta t = T_{\text{filter}}$ of the LTI velocity estimators is used in the measurement update, giving a relatively high, and fixed, gain from Δp to Δv.

Measurement Gain Limitation. The case when the time Δt between events is very short is also problematic. Since the gain from errors in Δt and y to Δv becomes very high, any measurement noise that is not captured by the quantization model will be heavily amplified. Since we use the K and T_i parameters from the PI controller, which has been designed to tolerate a lag of T_{filter} in the velocity estimate, we limit Δt in (10) to be no shorter than T_{filter}. Beyond this limit, $\hat{v}$ will behave more like the first order filters of the PI controllers, introducing some low pass filtering on the measurements.

6. Implementation Issues

In this section, a variety of the issues encountered when implementing the LTI and event-based controllers on a low-end microcontroller are discussed.

6.1 Choice of Time Constants

The values of T_i, T_{filter} and ω_c are adjusted so that all multiplications in the Pure PI controller can be implemented with bit shifts. Different power of 2 ratios were tried to find a design with good tradeoff between robustness and disturbance rejection.

1. To avoid excessive aliasing, T_{filter} is chosen $\approx 8h$. The ratio is adjusted to place the pole in $z = b = 7/8$ in the discretization of the velocity filter in (2),

$$H_{\text{filter}}(z) = \frac{1-b}{z-b}\frac{z-1}{h}. \tag{12}$$

2. The integral time is taken as $T_i = 64h$.

3. The cross-over frequency ω_c is chosen to make the gain $\frac{K\Delta p_{\text{quant.}}}{K_u h}$ from velocity error in units of $\Delta p_{\text{quant.}}/h$ to the motor voltage u_V a power of 2. The suitable cross-over frequencies are $\omega_c =$ 5, 10, 20, 40, 80 rad/s, which decides the sample rate h.

Keeping a fixed ratio between T_i, T_{filter}, and h means that the fixed-point scalings need only small adjustments when changing ω_c.

6.2 Discretization of LTI Controllers

PI Controller. The velocity filter is implemented using (12) as $\hat{V} = H_{\text{filter}}(z)Y$. The control law (3) is implemented as

$$U = K(V_{\text{ref}} - \hat{V}) - \frac{K}{T_i}\left(Y - \frac{zh}{z-1}V_{\text{ref}}\right),$$

where the unfiltered difference $\frac{z-1}{zh}Y$ is used instead of $\hat{V}$ in the I-part. This is favourable since Y is better known than V.

PI Controller with Observer. The velocity filter (5) with feed forward from the P-part u_P is discretized as

$$\hat{V} = \frac{hU_P + (1-b)(z-1)/h}{z-b},$$

and the control law (4) as

$$U = \underbrace{K(V_{\text{ref}} - \hat{V})}_{U_P} - \frac{K}{T_i}\left(Y - \frac{zh}{z-1}\hat{V}\right).$$

6.3 Multi-rate Sampling

The encoder must be sampled every 10 μs = 128 clock cycles to be able to follow cart speeds of up to 5 m/s. There is not much time for calculation in 128 clock cycles and no need to run the controller that often, so it executes at a slower rate. The estimator first updates $\hat{v}$ and $\hat{u}_{\text{bias}}$, and then the controller computes $u = K(\beta v_{\text{ref}} - \hat{v}) - \hat{u}_{\text{bias}}$.

At each invocation of the estimator, it reads the current position. For the event-based estimators, the encoder sampler also saves the

direction and time stamp of the last event, allowing the estimator to use a much higher time resolution than its own sampling period. If there was an event during the last sample, the time update is taken up to the event, the measurement update applied, and the time update then taken for the remaining part of the time step. Otherwise, the time update is taken for the whole time step, and the measurement update applied afterwards, if needed.

6.4 Fixed-Point Arithmetic

All variables are represented in fixed-point arithmetic, most in 32 bits but some in 16 bits. Distance is measured in quantization steps and time in encoder sampler periods. All fixed gains in the Pure PI controller can be implemented with bit shifts. The process gain K_u is quite uncertain and is represented with 5 bits of precision, which saves on computation.

6.5 Division

The AVR microcontroller has no hardware support for integer division, though there is a fast hardware multiplier. Division by Δt in the event-based measurement update is thus implemented by first approximating $x = \Delta t^{-1}$ and then multiplying.

1. Δt is first normalized by shifting to be within the range $[1, 2)$, in $1 : 15$ fixed point representation.

2. A 256-entry inverse table gives an 8 bit initial guess, which is extended to 15 bits.

3. A Newton iteration for the function $f(x) = x^{-1} - \Delta t$ is applied. Though the operation is carried out in 16 bit precision, the only multiplications that are needed is one 8x16 bit and one 8x8 bit. The initial guess has only 8 bits of precision, and the error in the initial guess is small enough to fit in 8 bits.

The result is an inverse of about 14 bits accuracy, which was deemed sufficient.

6.6 Concurrency

The tasks that the microcontroller has to handle concurrently are listed in Table 1. For the encoder sampling and control tasks, we must meet

Table 1.   Characteristics of the controller tasks.

Task	Period	Priority
Encoder sampling	$10\,\mu s$	high
Encoder support	$1280\,\mu s$	medium
Serial receive	$\geq 87\,\mu s$	medium
Serial transmit	$\geq 87\,\mu s$	medium
Control	$1000\,\mu s$	low

certain rates to keep track of the position and achieve the desired control. The serial communication is used to receive the set point from and send measurements to a PC. For these tasks, we want to come close to the speed limit imposed by the hardware. The tasks are split into three priorities:

High.   Since the encoder must be sampled very fast, it is given a dedicated timer interrupt and optimized in assembly language. To minimize execution time, only 8 bit quantities are updated by the interrupt: the lowest bits of the position, time and time stamps. An encoder support task will propagate changes in the 8 bit quantities to 32 bit quantities.

Medium.   A second timer interrupt at a lower rate is used to poll the medium priority tasks. To maximize the serial communication bandwidth, the period is set at the lowest multiple of the encoder sample period that guarantees that one byte can be sent and received at each invocation. The phase is adjusted so that the polling interrupt is triggered right after a trigger of the encoder interrupt, to minimize interference with the encoder sampling.

Low.   The controller sample rate is relatively low, and the latency is high, so the estimator and controller are run in the main loop. This also allows experimentation with control algorithms without concerns for breaking the timing of the other tasks.

6.7 Step by Step Estimator Implementation

The implementation of the event-based estimator requires a number of steps, but with the sequence suggested here, the controller can be

verified to work after each. The Basic Event Estimator is an approximation that should be good when events are quite frequent, and is refined gradually. It is naturally implemented starting from an implementation of the Pure PI controller, which supplies the control law that is used throughout:

$$u = K(\beta v_{\text{ref}} - \hat{v}) - \hat{u}_{\text{bias}}$$

(though β is set to one in the end).

Basic Event Estimator. The estimates $\hat{p}$ and $\hat{v}$ are constant between events: there is no time update, and measurement updates are at events only. The measurement update (10) is simplified with

$$\hat{v}^+ = \frac{\Delta p}{\Delta t}, \quad \Delta p = \hat{p}^+ - \hat{p},$$

to work when there is no time update for $\hat{p}$. The $\hat{u}_{\text{bias}}$ update mirrors the Pure PI controller's $\hat{U}_{\text{bias}} = \frac{K}{sT_i}(\hat{V} - V_{\text{ref}})$ in (3):

$$\hat{u}_{\text{bias}}(k) = \hat{u}_{\text{bias}}(k-1) + \frac{K}{T_i}\Big(\hat{p}(k) - \hat{p}(k-1) - hv_{\text{ref}}(k)\Big).$$

Position Prediction. Now $\hat{v}$ is used to predict the evolution of $\hat{p}$ between events, introducing as time update the relevant part of (9):

$$\hat{p}(k+1) = \hat{p}(k) + h\hat{v}(k).$$

With this time update, the intended form (10) of the measurement update should be used. It is now possible to lower limit Δt by T_{filter} according to section 5.3; the correction form of the measurement update (10) will ensure that $\hat{v}$ eventually converges to the correct value. The only other visible difference from taking this step is to remove the I-part windup between events when $v \approx v_{\text{ref}}$.

Measurement Updates Between Events. Now it is straight forward to apply the measurement update (10) as soon as the position measurement disagrees with $\hat{p}$, allowing the controller to react faster to drops in speed.

LTI Mode Timeout.   The LTI Mode Timeout fix of section 5.3 is added. This is needed in the last implementation step to avoid the controller becoming too soft during startups.

Full Event Estimator.   The full time update (9) is implemented by adding the $\hat{v}$ part, and $\hat{u}_{\text{bias}}$ is now updated only in the measurement update, using (11). Finally, the analysis of the Observer-Based PI controller in section 3.2 applies, so we set $\beta = 1$ and make T_i four times faster to improve disturbance rejection.

7. Experimental Comparison

The controllers to be compared are the Pure PI controller, the Observer-Based PI controller, the Basic Event Estimator, the Event Estimator with Position Prediction, and the Full Event Estimator. Since it takes a only minor implementation effort, the Event Estimator with Position Prediction has measurement updates also between events.

To compare the performance of the different controllers under different conditions, a number of step response experiments were performed. In each experiment, the cart begins at rest at position $p = 0$ with all estimates at zero. At time $t = 0$, a step in v_{ref} is made. The cart is allowed to run for 1 m, counting the first 0.3 m as startup and the rest as stationarity. After 1 m, v_{ref} is stepped back to zero.

To have an accurate way to compare the velocity trajectories for the different controllers, the full resolution of the encoder was used for offline evaluation, while in most of the experiments, all feedback was based on a software emulated encoder with the $q = 6$ lowest bits dropped. To reconstruct the velocity trajectory from an experiment, a simple form of the JMAP estimator is used. Only $\hat{p}$ and $\hat{v}$ are used as states, and the effect of u is ignored. Since there is some jitter in the serial communication, the measurement constraints (7) are relaxed to allow that each measurement may have arrived one sample to soon or too late. The optimization problem for the whole trajectory conditioned on all measurements is solved simultaneously.

Fig. 4 shows typical experimental results for the Pure PI and Full Event controllers at $q = 6$, $\omega_c = 20$ rad/s, $v_{\text{ref}} = 0.8$ m/s. We see that
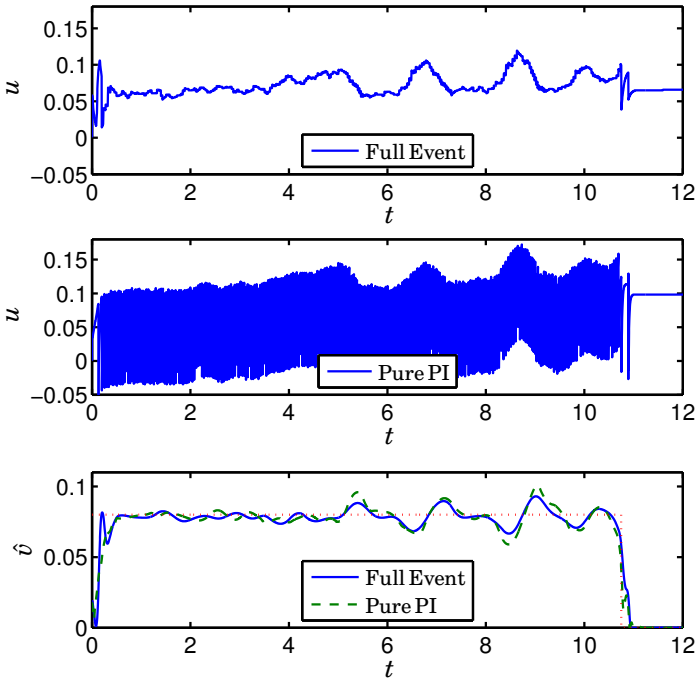
Figure 4. Experimental results for the Pure PI and Full Event controllers with $\omega_c = 20$ rad/s. The dotted line shows v_{ref}. The friction varies faster at the far end of the track, degrading the velocity tracking in the later half.

the PI controller spends a considerably greater control effort, and that the Full Event controller has slightly better reference tracking.

7.1 Performance Metrics

To measure the control effort, we use

$$\sigma_u = \frac{\mathrm{std}(u)}{\mathrm{mean}(u)},$$
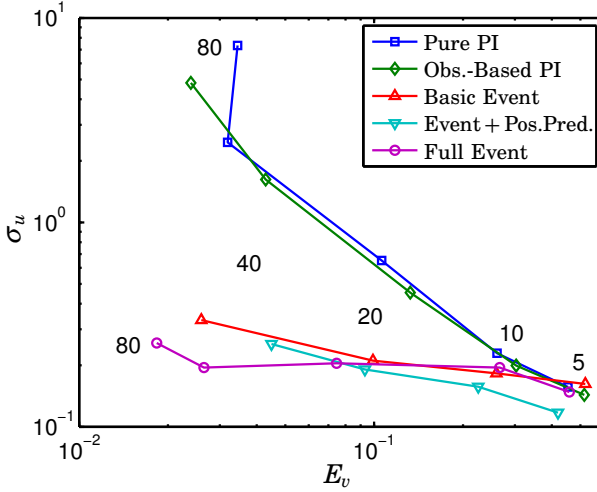
143

Figure 5. Experimental comparison of control signal activity σ_u versus velocity RMS error E_v for the LTI controller and three development stages of the event-based controller. Each controller is used with cross-over frequency $\omega_c = 5, 10, 20, 40, 80$ rad/s, (indicated by numbers in the figure) giving successively lower velocity error but higher control signal activity. At $\omega_c = 80$ rad/s, the two simplified event-based controllers become unstable in this case. Each dot corresponds to one experiment.

where the standard deviation and mean are taken over the last 0.7 m of the trajectory. The mean is taken over all experiments with the same v_{ref}. The reference tracking error is measured by the RMS error over the last 0.7 m, normalized by v_{ref}:

$$E_v = \sqrt{\frac{1}{N} \sum_{k=1}^{N} \left(\frac{\hat{v}(k)}{v_{\text{ref}}} - 1 \right)^2}.$$

Startup time is measured by

$$t_{\text{startup}} = \frac{t_{p=p_{\text{startup}}}}{t_{\text{ref}}} - 1, \qquad t_{\text{ref}} = \frac{p_{\text{startup}}}{v_{\text{ref}}},$$
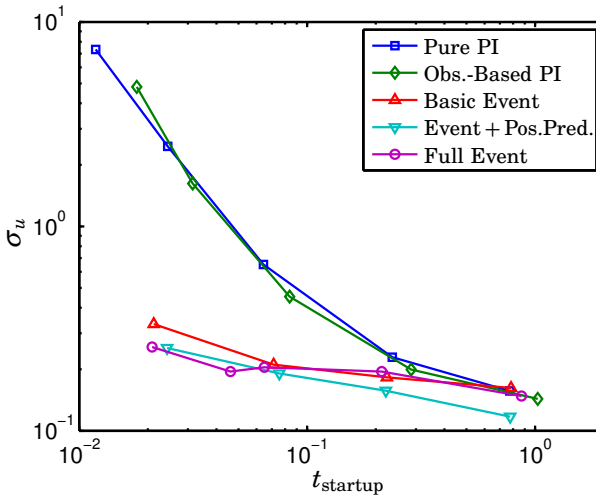
where $p_{\text{startup}} = 0.3$ m.

Figure 6. Experimental comparison of control signal activity σ_u versus startup delay t_{startup} for the experiments in Fig. 5. The full event-based controller has a higher delay due to its different disturbance estimate.

7.2 Controller Comparison

To compare the controllers, experiments were made with $v_{\text{ref}} = 0.8$ m and $q = 6$, varying $\omega_c = 5, 10, 20, 40, 80$ rad/s. This gives about 1.7 events/T_{filter} for the Pure PI controller at $\omega_c = 5$ rad/s, and decreasing. For the Full Event and Observer-Based PI controllers, which had reduced T_i, T_i had to be lower bounded to the value used by the Pure PI Controller at $\omega_c = 80$ rad/s to avoid instability.

Fig. 5 compares the control effort and velocity tracking for different controllers and cross-over frequencies. We see that as the control loops become faster, the control effort of the LTI controllers rises much steeper than for the event-based controllers. At first, the velocity tracking improves in much the same way for all controllers, but eventually the Full Event controller wins out.

The greatest gain comes from going from LTI control to the Basic Event controller, but for high bandwidth, there is more to gain with the Full Event controller. The break point where event-based control gives lower control effort than LTI seems to be around one event per
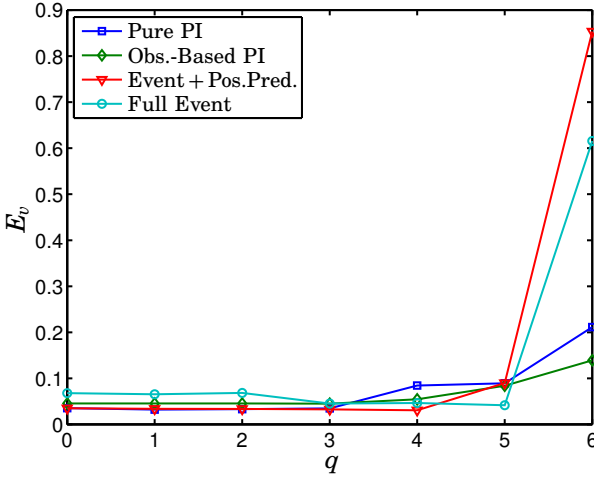
Figure 7.   Experimental comparison of velocity RMS error E_v for LTI and event-based controllers versus number of discarded encoder bits q.

T_{filter}, somewhere between $\omega_c = 5$ rad/s and $\omega_c = 10$ rad/s in this experimental setup.

Fig. 6 compares the control effort and t_{startup} for the same cases as in Fig. 5. Here, the PI controller is a bit faster for high ω_c, though at a considerable control effort.

7.3 Quantization Dependence

To explore quantization effects, the experiments with $\omega_c = 40$ rad/s above were rerun with $v_{\text{ref}} = 0.4$ m/s, varying the quantization as $q = 0 \ldots 6$. The Basic Event Estimator was excluded, since its inability to lower bound Δt made the control signal very noisy at low quantization q.

Figs. 7 and 8 show the tracking error E_v and control effort σ_u with the different encoder resolutions. As the quantization decreases, both E_v and σ_u generally improve, seeming to settle at a quantization free level. The best tracking performance is almost achieved already at $q = 5$, at which point the Full Event controller has also achieved
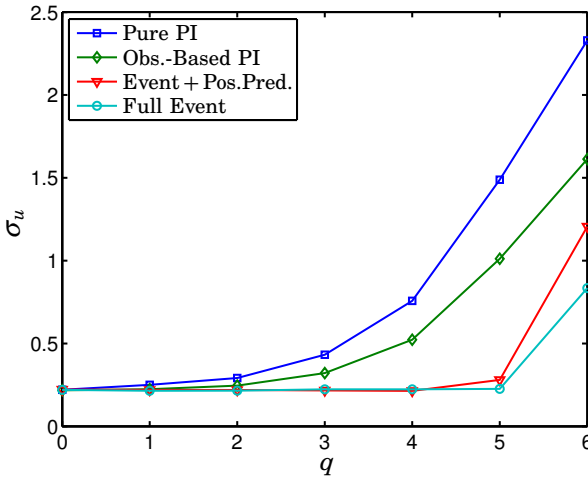
Figure 8. Experimental comparison of control signal activity σ_u for LTI and event-based controllers versus number of discarded encoder bits q.

minimum control effort. The Event controller with Position Prediction achieves minimum σ_u at $q = 4$. The control effort of the LTI controllers decreases only gradually, the Observer-Based PI controller being a bit more gentle.

The Full Event controller actually performs at its best with some extra quantization in this case, so there is some room for improvement to make it behave more like the other controllers when events are frequent.

8. Conclusion

This paper presented a simple event-based state estimator for a moving cart with quantized position measurements, derived as a simplification of a joint maximum a posteriori (JMAP) estimator. Velocity control based on the event-based estimator was compared experimentally to classical linear time-invariant (LTI) controllers. In the experiments,

it was seen that the benefits of event-based control begin to appear when the LTI controllers are unable to filter out the quantization noise efficiently, around the point of one quantization step per velocity filter time constant.

The foremost benefit with event-based control is to greatly reduce the noise in the control signal. The lowered control effort makes it practical to use a much higher gain in the control loop, improving disturbance rejection. Already a simplified event-based controller comes a long way compared to the LTI controllers, but with high controller gain, the full event-based estimator shows superior performance.

Acknowledgment

References

Årzén, K.-E. (1999): "A simple event-based PID controller." In *Proc. 14th IFAC World Congress*.

Åström, K. J. and B. Bernhardsson (1999): "Comparison of periodic and event based sampling for first-order stochastic systems." In *Proc 14th IFAC World Congress*.

Åström, K. J. and B. Wittenmark (1997): *Computer-Controlled Systems*, 3rd edition. Prentice Hall.

Curry, R. E. (1970): *Estimation and Control with Quantized Measurements*. Research Monograph No. 60. M.I.T. Press.

Henningsson, T., E. Johannesson, and A. Cervin (2008): "Sporadic event-based control of first-order linear stochastic systems." *Automatica*, **44:11**, pp. 2890–2895.

Miskowicz, M. (2007): "Asymptotic effectiveness of the event-based sampling according to the integral criterion." *Sensors*, **7:16–37**.

Rawlings, J. and B. Bakshi (2006): "Particle filtering and moving horizon estimation." *Computers and Chemical Engineering*, **30:10-12**, pp. 1529–1541.

Sandee, J. H., W. P. M. H. Heemels, and P. P. J. van den Bosch (2007): "Case studies in event-driven control." In *Proc. Hybrid Systems: Computation and Control*, pp. 762–765.

Vasyutynskyy, V. and K. Kabitzsch (2007): "Simple PID control algorithm adapted to deadband sampling." In *Proc. 12th IEEE Conference on Emerging Technologies and Factory Automation (ETFA'07)*, pp. 932–940.

Department of Automatic Control **Lund University** **Box 118** **SE-221 00 Lund Sweden**	*Document name* LICENTIATE THESIS
	Date of issue November 2008
	Document Number ISRN LUTFD2/TFRT--3244--SE
Author(s) Toivo Henningsson	*Supervisor* Anton Cervin Anders Rantzer
	Sponsoring organisation VR, ARTIST2

Title and subtitle

Event-Based Control and Estimation with Stochastic Disturbances

Abstract

This thesis deals with event-based control and estimation strategies, motivated by certain bottlenecks in the control loop. Two kinds of implementation constraints are considered: closing one or several control loops over a data network, and sensors that report measurements only as intervals (e.g. with quantization). The proposed strategies depend critically on *events*, when a data packet is sent or when a change in the measurement signal is received. The value of events is that they communicate new information about stochastic process disturbances.

A data network in the control loop imposes constraints on the event timing, modelled as a minimum time between packets. A thresholdbased control strategy is suggested and shown to be optimal for firstorder systems with impulse control. Different ways to find the optimal threshold are investigated for single and multiple control loops sharing one network. The major gain compared to linear time invariant (LTI) control is with a single loop a greatly reduced communication rate, which with multiple loops can be traded for a similarly reduced regulation error.

With the bottleneck that sensors report only intervals, both the theoretical and practical control problems become more complex. We focus on the estimation problem, where the optimal solution is known but untractable. Two simplifications are explored to find a realistic state estimator: reformulation to a mixed stochastic/worst case scenario and joint maximum a posteriori estimation. The latter approach is simplified and evaluated experimentally on a moving cart with quantized position measurements controlled by a low-end microcontroller.

The examples considered demonstrate that event-based control considerably outperforms LTI control, when the bottleneck addressed is a genuine performance constraint on the latter.

Key words

event-based control, event-based estimation, control over networks, quantized measurements, stochastic control, sporadic control, embedded control

Classification system and/or index terms (if any)

Supplementary bibliographical information

ISSN and key title 0280–5316	*ISBN*

Language English	*Number of pages* 152	*Recipient's notes*
Security classification		