

# Sporadic Event-Based Control using Path Constraints and Moments

Toivo Henningsson

**Abstract**—Control is traditionally applied using periodic sensing and actuation. In some applications, it is beneficial to use instead event based control, to communicate or make a change only when necessary. There are no known general closed form solutions to such event based control problems. We consider stationary event-based control problems with mixed continuous/discrete time dynamics and stochastic disturbances. The system is modelled by a set of path constraints, which are converted into constraints on trajectories’ moments up to some order  $N$ ; upper and lower bounds on the control objective for any system that meets the constraints are derived using sum-of-squares techniques and convex semidefinite programming. Joint optimization of upper bound and controller parameters is non-convex in general; approaches to such controller optimization are investigated, including local optimization using bilinear matrix inequalities. Examples show that the bounds are significantly tighter than earlier results obtained using quadratic value functions.

## I. INTRODUCTION

Digital control is traditionally carried out using periodic sampling and actuation. Sometimes, however, there is a bottleneck in the control loop. There may be a fixed cost or a minimum time between *events* such as to transmit a state estimate or change a control signal. In *Event-based control*, the decision when to generate an event is taken dynamically, rather than to pick a fixed sample rate a priori.

Event-based control can mean many different things. It can be phrased in a stochastic, deterministic, or worst-case setting, with linear or nonlinear dynamics, in continuous or discrete time, with the aim to reduce computation, communication or actuation. In a non-stochastic setting, some authors predict the next event time in advance, see e.g [17], [18].

This paper considers systems with linear dynamics and stochastic disturbances, and the objective to reduce communication or actuation. Both continuous time (CT) and discrete time (DT) settings will be considered; in fact, trajectories may switch back and forth between flow (CT) and jump (DT), see Fig. 1.

One way to approach the class of problems considered in this paper is to discretize the system into a Markov chain, and then solve the optimal control problem using dynamic programming [1]; this is applied to single state plants in [7]. This method has exponential complexity in the number of state variables. To deal with more than a few states, we will consider instead value functions up to some fixed polynomial degree  $N$ , which gives polynomial complexity.

This work has been supported by the Swedish Research Council, the LCCC Linnaeus Center, and the ELLIIT Excellence Center.

T. Henningsson is with the Department of Automatic Control LTH, Lund University, Box 118, SE-221 00 Lund, Sweden. Email: toivo.henningsson@control.lth.se

In [2], impulse control of a continuous time (CT) integrator plant with a white noise disturbance was considered. It was shown that the mean event frequency can be reduced to a third by using a threshold based event triggering strategy instead of periodic events, for the same state variance. However, such a control policy is *aperiodic*; the time between two events may be arbitrarily short, making it hard to implement in practice. Several other authors have also investigated aperiodic CT problems, e.g [13], [14], [8]. To get an implementable control law, some authors, e.g [4], [3], [9], [15] have considered event-based control in discrete time (DT), with a cost term for each sample with an event.

We are interested in the slightly broader class of *sporadic* controllers [7], with a guaranteed waiting time between any two events. After this period of *inactive state*, the controller may begin to monitor the plant state continuously, or at some sample rate. CT and DT sporadic control is also considered in [5], (where sporadic CT is called non-uniform control) under the objective of ultimate boundedness.

In the last decade, moment relaxations (see e.g [16]), and their dual, sum-of-squares (SOS) restrictions (see e.g [12], [11]), have gained popularity to approximate nonlinear optimal control problems without closed form solutions. Typically, lower bounds on achievable cost are found, which improve as the problem size grows with relaxation order. This paper is an adaptation of such techniques to event-based optimal control problems. By including the controller in the model, we can also find and optimize upper bounds on cost.

One motivating example that can be (approximately) solved with the methods in this paper is the following sporadic control problem: a classic linear quadratic (LQ) problem with the added constraints that 1) the control signal is zero except for *control events*, when it may be a (vector) Dirac impulse, 2) there is a minimum time  $\Delta T$  between control events. A fixed cost per control event may be added, and a filter on the plant input to shape the control waveform. A jump transition is created by sampling the system for a

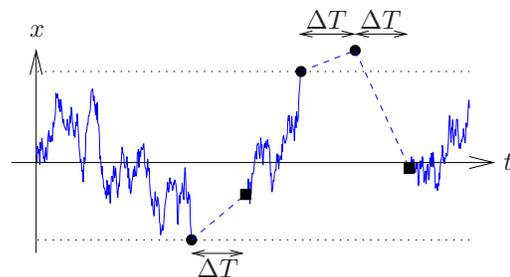


Fig. 1. Example of a mixed flow/jump trajectory. When entering a jump (dots), the system jumps to a new state  $x_+$  and time  $t_+ = t + \Delta T$  (squares).

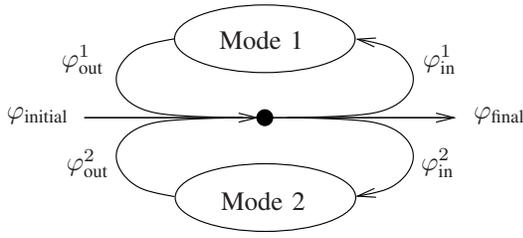


Fig. 2. General mode switching model: each time the trajectory leaves a mode, the controller decides to enter either Mode 1, Mode 2, or terminate.

time  $\Delta T$  after each control event (see Fig. 1), which recasts the sporadic control problem into a mode switching control problem (see Fig. 2). The mode without control may be CT or DT (possibly with a time step  $\neq \Delta T$ ).

The paper is outlined as follows: After preliminaries in Section II, the event-based control problem is formulated in Section III. Path constraints to model the system are described in Section IV, and combined in Section V using convex optimization to show bounds on cost for any system that meets them; these problems are cast as semidefinite programs (SDP:s) in Section VI to facilitate efficient solution.

For lower bound problems, the degrees of freedom of the controller can be left unconstrained; the bound will hold for any controller, including the optimal. For upper bound problems, the controller must be included as a constraint. Section VII considers approaches to joint optimization of controller parameters and upper bounds, which is in general non-convex. Results are presented in Section VIII and conclusions are given in Section IX.

The source code for the toolbox used to produce the numerical results in this paper is available online [6].

## II. PRELIMINARIES

For matrices  $A, B$ , let  $A \succeq B$  denote that  $A - B$  is positive semidefinite. Given that  $\mathcal{X} = \mathbb{R}^n$ : Let  $\mathcal{V}(\mathcal{X})$  be a space of test functions (typically polynomials)  $V : \mathcal{X} \mapsto \mathbb{R}$ . For  $f, g \in \mathcal{V}(\mathcal{X})$ , let  $f \geq g$  denote pointwise inequality:  $f(x) - g(x) \geq 0, \forall x \in \mathcal{X}$ . Let  $\mathcal{V}_+(\mathcal{X}) \subset \mathcal{V}(\mathcal{X})$  be the convex cone of (pointwise) positive functions  $V \geq 0, V \in \mathcal{V}(\mathcal{X})$ .

Let  $\mathcal{V}_N(\mathcal{X})$  be the space of (multivariate) polynomials over  $\mathcal{X}$  of degree  $\leq N$ . Let  $\Sigma_N(\mathcal{X}) \subset \mathcal{V}_N(\mathcal{X})$  be the convex cone of sum-of-squares polynomials of degree  $\leq N$ , i.e. the convex closure of  $V_{N/2}(\mathcal{X}) \cdot V_{N/2}(\mathcal{X})$ . Given a basis  $\psi(x)$  for  $V_{N/2}(\mathcal{X})$ , it is well known that  $\lambda \in \mathcal{V}_N(\mathcal{X})$  is also  $\in \Sigma_N(\mathcal{X})$  iff there is a matrix  $\Lambda \succeq 0$  such that  $\lambda(x) = \psi(x)^T \Lambda \psi(x)$ .

## III. PROBLEM FORMULATION

Consider a system that can switch between two *modes*  $m \in \mathcal{M} = \{\text{flow}, \text{jump}\}$ , with different dynamics for the *state*  $x \in \mathcal{X} = \mathbb{R}^{n_x}$ . A *trajectory* (or path) consists of *parts*  $k \in \mathcal{K} = \{1, 2, 3, \dots\}$ , each within one mode  $m_k \in \mathcal{M}$ . The controller may switch modes freely between parts, see Fig. 2. The trajectory begins at time  $k = 0, t = t_k^{\text{in}} = t_{\text{initial}}$  and state  $x = x_k^{\text{in}} = x_{\text{initial}}$ .

Entering the *flow mode* at time  $t = t_k^{\text{in}}$  and state  $x_k(t_k^{\text{in}}) = x_k^{\text{in}}$ , the state  $x$  evolves until  $t = t_k^{\text{out}}, x_k(t_k^{\text{out}}) = x_k^{\text{out}}$ , by the

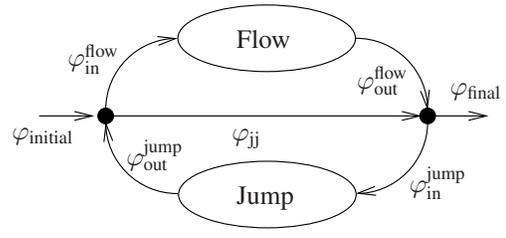


Fig. 3. Flow-jump mode switching model: When the controller decides to exit flow, it must take a jump. After a jump, it may decide either way.

(stochastic differential equation) dynamics

$$dx_k = Ax_k dt + Bu_k^{\text{flow}} dt + dw, \quad E(dw dw^T) = R dt, \quad (1)$$

where  $u_k^{\text{flow}}(t) \in \mathcal{U}_{\text{flow}} = \mathbb{R}^{n_{\text{flow}}}$  is the control signal,  $w$  is a Wiener Process, (independent of the past trajectory), and  $R \succeq 0, A, B$  are model matrices of appropriate dimensions. The controller may decide to exit the flow mode at any time.

Entering the *jump mode* at  $t = t_k^{\text{in}}$  causes a jump that ends at  $t = t_k^{\text{out}} = t_k^{\text{in}} + \Delta T, \Delta T \geq 0$  and state

$$x_k^{\text{out}} = \Phi x_k^{\text{in}} + \Gamma u_k^{\text{jump}} + w_k, \quad w_k \in \mathcal{N}(0, P_{\text{jump}}), \quad (2)$$

where  $u_k^{\text{jump}} \in \mathcal{U}_{\text{jump}} = \mathbb{R}^{n_{\text{jump}}}$  is the control signal, the Gaussian disturbance  $w_k$  is independent of the past trajectory, and  $P_{\text{jump}} \succeq 0, \Phi, \Gamma$  are model matrices of appropriate dimensions. The jump time  $\Delta T$  is also a model parameter.

*Remark 1:* For brevity, we describe only the case with one flow and one jump mode. The switching model of Fig. 3 is appropriate in this case, since it disallows consecutive flow parts; we will still use Fig. 2 in calculations for brevity. The methods in this paper apply also in the case of two jump modes, possibly with different time steps  $\Delta T_i$ .

The expected cost over trajectories  $j_{\text{acc}}$  is a sum of integrals over each flow interval and a term for each jump:

$$j_{\text{acc}} = \mathbb{E} \left[ \sum_{k \in \mathcal{K}_{\text{flow}}} \int_{T_k} c_{\text{flow}}(z_k(t)) dt + \sum_{k \in \mathcal{K}_{\text{jump}}} c_{\text{jump}}(z_k) \right], \quad (3)$$

where the index sets  $\mathcal{K}_m$  and part intervals  $T_k$  are given by

$$\mathcal{K}_m = \{k \in \mathcal{K}; m_k = m\}, \quad T_k = [t_k^{\text{in}}, t_k^{\text{out}}],$$

the *extended state*  $z$  in flow and jump respectively by

$$z_k(t) = \begin{pmatrix} x_k(t) \\ u_k^{\text{flow}}(t) \end{pmatrix} \in \mathcal{Z}_{\text{flow}}, \quad z_k = \begin{pmatrix} x_k^{\text{in}} \\ u_k^{\text{jump}} \end{pmatrix} \in \mathcal{Z}_{\text{jump}},$$

$\mathcal{Z}_m = \mathcal{X} \times \mathcal{U}_m$ , and the cost functions  $c_m \in \mathcal{V}_+(\mathcal{Z}_m)$ .

*Remark 2:* The function  $c_{\text{flow}}(z)$  is the cost *per time unit* in flow mode, while  $c_{\text{jump}}(z)$  is the cost *per jump*.

The controller consists of two parts:

- A switching law  $\theta(x)$  to choose mode  $m = \text{flow}$  when  $\theta(x) \geq 0$ , and mode  $m = \text{jump}$  otherwise.
- Modal control laws  $u_m = f_m(x), m \in \mathcal{M}$ .

The control objective is to minimize the *average cost*

$$J = \bar{\mathcal{R}}(j_{\text{acc}}) = \limsup_{t_{\text{spent}} \rightarrow \infty} \frac{1}{t_{\text{spent}}} j_{\text{acc}}, \quad (4)$$

where the trajectory duration  $t_{\text{spent}}$  is given by

$$t_{\text{spent}} = t_{\text{final}} - t_{\text{initial}} = \sum_{k \in \mathcal{K}} t_k^{\text{out}} - t_k^{\text{in}}. \quad (5)$$

#### IV. PATH CONSTRAINTS

We will now list a number of path constraints to model the considered system. In order to show bounds on path integrals such as the cost (3) in the next section, nonnegative path integrals are derived from the constraints. We first introduce a compact notation for path integrals using measures.

##### A. Path measures

Define the *occupation measure*  $\mu$  and *jump event measure*  $\varphi$ , with arguments  $f \in \mathcal{V}(\mathcal{Z}_{\text{flow}})$ ,  $f \in \mathcal{V}(\mathcal{Z}_{\text{jump}})$  respectively:

$$\mu(f) = \mathbb{E} \sum_{k \in \mathcal{K}_{\text{flow}}} \int_{T_k} f(z_k(t)) dt, \quad \varphi(f) = \mathbb{E} \sum_{k \in \mathcal{K}_{\text{jump}}} f(z_k).$$

Given a function  $f(z)$  of the extended state  $z = (x, u^m)$ ,  $\mu(f)$  can be thought of as an accumulator that integrates  $f(z)dt$  along the parts of the trajectory in flow, and  $\varphi(f)$  as one that adds up  $f(z)$  for each jump.

Using  $\mu$  and  $\varphi$ , the accumulated cost (3) and trajectory duration (5) can be expressed more compactly as

$$j_{\text{acc}} = \mu(c_{\text{flow}}) + \varphi(c_{\text{jump}}), \quad (6)$$

$$t_{\text{spent}} = \sum_{k \in \mathcal{K}_{\text{flow}}} \int_{T_k} dt + \sum_{k \in \mathcal{K}_{\text{jump}}} \Delta T = \mu(1) + \varphi(\Delta T), \quad (7)$$

where 1 in  $\mu(1)$  means the constant function  $f(z) = 1$ , and in the same way for  $\varphi(\Delta T)$ .

To describe mode switching such as in Figs. 2 and 3, we define, for the initiation and termination events, measures

$$\varphi_{\text{initial}}(f) = \mathbb{E} f(x_{\text{initial}}), \quad \varphi_{\text{final}}(f) = \mathbb{E} f(x_{\text{final}}),$$

and, accumulating mode entry and exit events, measures

$$\varphi_{\text{dir}}^m(f) = \mathbb{E} \sum_{k \in \mathcal{K}_m} f(x_k^{\text{dir}}), \quad m \in \mathcal{M}, \text{dir} \in \{\text{in}, \text{out}\}.$$

Note that the jump event measure  $\varphi$  and jump entry measure  $\varphi_{\text{in}}^{\text{jump}}$  are not the same, since  $\varphi$  is defined over the extended state  $z_{\text{jump}}$ , and  $\varphi_{\text{in}}^{\text{jump}}$  over the state  $x$  only. However, they coincide for  $u_{\text{jump}}$ -independent test functions:

$$\varphi(V) = \varphi_{\text{in}}^{\text{jump}}(V), \quad \forall V \in \mathcal{V}(\mathcal{X}). \quad (8)$$

Having defined the path measures, we will now use them to formulate path constraints and nonnegative path integrals.

##### B. Pointwise path constraints

The simplest form of path constraints express feasible regions of the (extended) state space. (Such algebraic equations can be used for differential-algebraic equation (DAE) systems modelling.) Consider the constraint that  $f(z_{\text{flow}}) = 0$  when the trajectory is in flow mode, for some given function  $f(z)$ . Then also  $f(z_{\text{flow}})V(z_{\text{flow}}) = 0$  for any function  $V \in \mathcal{V}(\mathcal{Z}_{\text{flow}})$ , as is the path integral

$$\mu(fV) = 0, \quad \forall V \in \mathcal{V}(\mathcal{Z}_{\text{flow}}).$$

The same can be done for event measures, e.g.  $\varphi(fV) = 0, \forall V \in \mathcal{V}(\mathcal{Z}_{\text{jump}})$  if  $f(z_{\text{jump}}) = 0$  for all jumps.

Now consider the inequality constraint that  $f(z_{\text{flow}}) \geq 0$  when in flow mode. Then also  $f(z_{\text{flow}})\lambda(z_{\text{flow}}) \geq 0$  for any nonnegative function  $\lambda$ , as is the path integral

$$\mu(f\lambda) \geq 0, \quad \forall \lambda \in \mathcal{V}_+(\mathcal{Z}_{\text{flow}}).$$

The constraint  $f(z) = 1 \geq 0$  apparently holds in any mode, and will be used since it establishes positivity of the path measures.

##### C. Control laws

Control laws can be expressed as path constraints; deterministic ones usually as pointwise ones. Examples:

- A switching law such that  $\theta(x) \geq 0$  in flow and  $\theta(x) \leq 0$  in jump.
- A control law  $u_{\text{jump}} = f_{\text{jump}}(x)$  is equivalent to the constraint that  $g(z_{\text{jump}}) = u_{\text{jump}} - f_{\text{jump}}(x) = 0$  in jumps.
- A random switching law, causing Poisson jumps in flow with a state-dependent intensity such that  $n_{\text{jump}}(x)$  jumps are expected per  $t_{\text{flow}}(x)$  time in flow, where  $n_{\text{jump}}, t_{\text{flow}} \in \mathcal{V}_+(\mathcal{X})$ . Then

$$\mu(\theta n_{\text{jump}}) - \varphi(\theta t_{\text{flow}}) = 0, \quad \forall \theta \in \mathcal{V}(\mathcal{X}). \quad (9)$$

This is not a pointwise constraint since the control law is random, but it holds in expectation, which is what we need.

##### D. Dynamics constraints

Dynamics constraints express how the trajectory may evolve from one instant to another.

1) *Mode switching*: The mode switching dynamics of the model in Fig. 2 are contained in the center point. Since each trajectory initiation and mode exit event is paired with exactly one termination or mode entry event, with the state  $x$  preserved across transitions, the switching constraint

$$\varphi_{\text{initial}} + \varphi_{\text{out}}^{\text{flow}} + \varphi_{\text{out}}^{\text{jump}} - (\varphi_{\text{final}} + \varphi_{\text{in}}^{\text{flow}} + \varphi_{\text{in}}^{\text{jump}}) = 0 \quad (10)$$

holds, where the argument  $V \in \mathcal{V}(\mathcal{X})$  to each measure has been suppressed for brevity. For the mode switching dynamics of Fig. 3 we have two switching points; they are modelled in the same way by pairing inflow and outflow,

$$\begin{aligned} \varphi_{\text{initial}} + \varphi_{\text{out}}^{\text{jump}} - (\varphi_{\text{in}}^{\text{flow}} + \varphi_{\text{jj}}) &= 0, \\ \varphi_{\text{out}}^{\text{flow}} + \varphi_{\text{jj}} - (\varphi_{\text{in}}^{\text{jump}} + \varphi_{\text{final}}) &= 0, \end{aligned} \quad (11)$$

again with the common argument  $V \in \mathcal{V}(\mathcal{X})$  in either equation suppressed. We see that the sum of these two equations is (10), thus (11) is a stronger constraint than (10).

2) *Flow dynamics*: Consider the flow dynamics (1). Given a (twice differentiable) function  $V \in \mathcal{V}(\mathcal{X})$ , the expected change in  $V(x)$  by the dynamics, conditioned on the extended state  $z$ , is (using Itô's Lemma)

$$\begin{aligned} \mathbb{E}(dV|z) &= \mathbb{E}(dx^T) \nabla V(x) + \frac{1}{2} \text{tr} \left( \mathbb{E}(dx dx^T) \nabla^2 V(x) \right) \\ &= \left( (Ax + Bu)^T \nabla V(x) + \frac{1}{2} \text{tr}(R \nabla^2 V(x)) \right) dt \\ &= (\mathcal{A}_{\text{flow}}^* V)(z) dt; \end{aligned}$$

this defines the *backwards flow dynamics operator*  $\mathcal{A}_{\text{flow}}^*$ , a Kolmogorov backwards operator. Equating the expectations of the left and right hand sides over the time spent in flow gives the *flow dynamics constraint*

$$0 = \mathbb{E} \sum_{k \in \mathcal{K}_{\text{flow}}} \int_{T_k} \left( (\mathcal{A}^* V)(z) dt - dV \right) = \mu(\mathcal{A}_{\text{flow}}^* V) - \left[ V \right]_{x_k^{\text{in}}}^{x_k^{\text{out}}} \\ = \mu(\mathcal{A}_{\text{flow}}^* V) + \varphi_{\text{in}}^{\text{flow}}(V) - \varphi_{\text{out}}^{\text{flow}}(V), \quad \forall V \in \mathcal{V}(\mathcal{X}). \quad (12)$$

3) *Jump dynamics*: Consider the jump dynamics (2). Given a function  $V \in \mathcal{V}(\mathcal{X})$ , the expected value of  $V(x^{\text{out}})$  after a jump, conditioned on  $z = (x, u)$  before the jump, is

$$\mathbb{E} \left( V(x^{\text{out}}) \middle| z \right) = \mathbb{E} \left( V(\Phi x + \Gamma u + w) \middle| z \right) \\ = (\phi * V)(\Phi x + \Gamma u), \\ = (\mathcal{H}^* V)(z),$$

where the probability density  $\phi$  is Gaussian  $\sim \mathcal{N}(0, P_{\text{jump}})$ ; this defines the *backwards single jump operator*  $\mathcal{H}^*$ . Summing over all events gives the *jump dynamics constraint*

$$\mathbb{E} \sum_{k \in \mathcal{K}_{\text{jump}}} V(x_k^{\text{out}}) = \mathbb{E} \sum_{k \in \mathcal{K}_{\text{jump}}} (\mathcal{H}^* V)(z_k) \\ \implies \varphi_{\text{out}}^{\text{jump}}(V) = \varphi(\mathcal{H}^* V), \quad \forall V \in \mathcal{V}(\mathcal{X}). \quad (13)$$

## V. BOUNDS ON COST BY CONVEX OPTIMIZATION

To show bounds  $\underline{J} \leq J \leq \bar{J}$  on the average cost (4) of a system, we will show positivity of path integrals such as  $l = j_{\text{acc}} - \underline{J}t_{\text{spent}}$  and  $l = \bar{J}t_{\text{spent}} - j_{\text{acc}}$ , by expressing them as a sum of nonnegative path integrals. In practice, it is sufficient to show that

$$l + \varphi_{\text{initial}}(V) - \varphi_{\text{final}}(V) \geq 0, \quad (14)$$

for some *value function*  $V \in \mathcal{V}(\mathcal{X})$  such that  $\varphi_{\text{final}}(V)$  is uniformly bounded from below as  $t_{\text{spent}} \rightarrow \infty$ . This boundedness can be established in many ways:

- For a lower bound, it may be sufficient that the bound holds for solutions with bounded moments of  $x_{\text{final}}$ ; then  $\varphi_{\text{final}}(V)$  will be bounded as well, for polynomial  $V$ .
- $x_{\text{final}}$  will have bounded moments if the flow region  $\{x \in \mathcal{X}; \theta(x) \geq 0\}$  is bounded and the jump dynamics (2) are exponentially stable.
- $\varphi_{\text{final}}(V)$  is uniformly bounded from below if  $V$  is.

### A. Lower bound

To show the lower bound  $\underline{J} \leq J$ , we want to show that

$$j_{\text{acc}} + \varphi_{\text{final}}(V) - \varphi_{\text{initial}}(V) \geq \underline{J}t_{\text{spent}}. \quad (15)$$

Note that the sign of  $V$  has been chosen opposite from (14). Using first (10), and then (8), (12) and (13), we see that

$$\varphi_{\text{final}}(V) - \varphi_{\text{initial}}(V) \\ = \varphi_{\text{out}}^{\text{flow}}(V) - \varphi_{\text{in}}^{\text{flow}}(V) + \varphi_{\text{out}}^{\text{jump}}(V) - \varphi_{\text{in}}^{\text{jump}}(V) \quad (16) \\ = \mu(\mathcal{A}_{\text{flow}}^* V) + \varphi(\mathcal{H}^* V) - \varphi(V)$$

The inequality (15) is then implied by

$$j_{\text{acc}} + \mu(\mathcal{A}_{\text{flow}}^* V) + \varphi(\mathcal{H}^* V) - \varphi(V) \\ = \underline{J}t_{\text{spent}} + \mu(\lambda_{\text{flow}}) + \varphi(\lambda_{\text{jump}}) \geq \underline{J}t_{\text{spent}}, \quad \lambda_m \in \mathcal{V}_+(\mathcal{Z}_m),$$

where we have used (16) and  $\mu, \varphi \geq 0$ . Collecting terms inside  $\mu$  and  $\varphi$ , this condition is in turn implied by

$$c_{\text{flow}} + \mathcal{A}_{\text{flow}}^* V = \underline{J} + \lambda_{\text{flow}}, \quad (17) \\ c_{\text{jump}} + \mathcal{H}^* V - V = \underline{J}\Delta T + \lambda_{\text{jump}},$$

for some  $\lambda_{\text{flow}} \in \mathcal{V}_+(\mathcal{Z}_{\text{flow}})$ ,  $\lambda_{\text{jump}} \in \mathcal{V}_+(\mathcal{Z}_{\text{jump}})$ .

### B. Lower bound with controller

To add a switching law such that  $\theta(x) \geq 0$  in flow, and  $-\theta(x) \geq 0$  in jump, we use

$$\mu(\theta\nu_{\text{flow}}) - \varphi(\theta\nu_{\text{jump}}) \geq 0, \quad \forall \nu_m \in \mathcal{V}_+(\mathcal{Z}_m). \quad (18)$$

The control law  $u_{\text{jump}} = f_{\text{jump}}(x)$  is incorporated by adding

$$\varphi(gW) = 0, \quad \forall W \in \mathcal{V}(\mathcal{Z}_{\text{jump}}),$$

to the left hand side of (15), where  $g(z_{\text{jump}}) = u_{\text{jump}} - f_{\text{jump}}(x)$ . With these control laws, (17) is strengthened into

$$c_{\text{flow}} + \mathcal{A}_{\text{flow}}^* V = \underline{J} + \lambda_{\text{flow}} + \theta\nu_{\text{flow}}, \quad (19) \\ c_{\text{jump}} + \mathcal{H}^* V - V + gW = \underline{J}\Delta T + \lambda_{\text{jump}} - \theta\nu_{\text{jump}}.$$

### C. Upper bound with controller

To show the upper bound  $J \leq \bar{J}$ , we want to show that

$$j_{\text{acc}} + \varphi_{\text{final}}(V) - \varphi_{\text{initial}}(V) \leq \bar{J}t_{\text{spent}}.$$

We proceed as before, but now all inequality terms have to be introduced with opposite sign. With controller constraints, the conditions (19) are turned into

$$c_{\text{flow}} + \mathcal{A}_{\text{flow}}^* V = \bar{J} - \lambda_{\text{flow}} - \theta\nu_{\text{flow}}, \quad (20) \\ c_{\text{jump}} + \mathcal{H}^* V - V + gW = \bar{J}\Delta T - \lambda_{\text{jump}} + \theta\nu_{\text{jump}}.$$

We see that the bound conditions above are convex, since they are linear with convex constraints on  $\{\lambda_m\}, \{\nu_m\}$ . Thus maximization of  $\underline{J}$  subject to (17) or (19) is a convex problem, as is minimization of  $\bar{J}$  subject to (20).

## VI. PRACTICAL OPTIMIZATION

To get problems that can be solved by a convex programming solver, we must choose some finite basis for the test functions  $V, \{\lambda_m\}, \{\nu_m\}$  and  $W$ . We will use polynomials up to some degree  $N$  of trajectory moments. A sum-of-squares restriction yields semidefinite programs (SDP:s).

We let the terms in (17), (19), and (20) be polynomials of degree  $\leq N$ . Since it is in general hard to determine the global positivity of a polynomial, we use  $\Sigma_N \subset \mathcal{V}_{N,+}$  to assure positivity; this can be expressed as a linear matrix inequality (LMI). The optimal bound can only improve with increasing  $N$ , as a solution to the bounds with lower  $N$  is still valid with higher  $N$ .

Making sure that no term in (17), (19), and (20) has higher degree than  $N$ , we can optimize over  $\underline{J} \in \mathbb{R}$  or  $\bar{J} \in \mathbb{R}$ , and

$$V \in \mathcal{V}_N(\mathcal{X}), \quad \lambda_m \in \Sigma_N(\mathcal{Z}_m), \\ \nu_m \in \Sigma_{N-\deg \theta}(\mathcal{Z}_m), \quad W \in \mathcal{V}_{N-\deg g}(\mathcal{Z}_{\text{jump}}).$$

The conditions (20) still give an SDP if we fix  $\nu_m$  and  $W$ , and include instead as optimization variables

$$\theta \in \mathcal{V}_{N-\max_{m \in \mathcal{M}} \deg \nu_m}(\mathcal{X}), \quad g \in \mathbb{G} \subseteq \mathcal{V}_{N-\deg W}(\mathcal{Z}),$$

where the space  $\mathbb{G}$  is chosen to give a desirable form for the  $u_{\text{jump}}$  controller, e.g. linear feedback.

## VII. CONTROLLER OPTIMIZATION

Now that we can model a system and derive upper and lower bounds  $\underline{J} \leq J \leq \bar{J}$  on the average cost  $J$ , how can we optimize for good controllers? We would like to prescribe a form for the switching law and modal controllers such as  $\theta \in \mathcal{V}_{N_\theta}(\mathcal{X})$ ,  $\{f_m \in \mathcal{V}_{n_f}(\mathcal{Z}_m)\}_{m \in \mathcal{M}}$ , and then find the controller parameters that give the lowest cost.

Since the actual cost  $J$  is unknown, we have to content with minimizing an upper bound  $\bar{J}$  instead. Unfortunately, joint optimization of upper bound and controller is generally non-convex because of the product terms between controller parameters and dual variables that appear in controller constraints, such as  $\theta \nu_{\text{flow}}$  in (20).

These product terms make the controller optimization into a bilinear matrix inequality (BMI) problem; we can still optimize locally given an initial guess. The formulation also allows various structural constraints on the controller such as limited polynomial degrees of  $\theta$  and  $\{f_m\}$ , or sparsity constraints, e.g. limiting the set of states that a control signal may depend on.

The controller optimization problem becomes convex if we fix enough decision variables so that no product terms with free variables remain. It is then possible to do global optimization by gridding over remaining variables. By making the problem simple, with low relaxation order  $N$  and few constraints, few parameters have to be scanned.

We next give some results relating tightness of the upper bound  $\bar{J}$  and problem complexity, and consider especially the case when global optimization can be done by scanning over a single real parameter.

### A. Mixing controllers

Consider a deterministic switching controller modelled by

$$\theta(x) \geq 0, \quad \text{in flow}, \quad -\theta(x) \geq 0, \quad \text{in jump}, \quad (21)$$

and a controller stochastically mixing time in flow:jump as  $t_{\text{flow}}(x) : n_{\text{jump}}(x)$ . By section IV-C, the positive path integral given by the former is

$$\mu(\theta \nu_{\text{flow}}) - \varphi(\theta \nu_{\text{jump}}) \geq 0, \quad \forall \{\nu_m \in \mathcal{V}_+(Z_m)\}_{m \in \mathcal{M}}.$$

This is exactly the same term as (9), if we identify  $n_{\text{jump}} = \nu_{\text{flow}}$ ,  $t_{\text{flow}} = \nu_{\text{jump}}$ . The bound derived from the deterministic switching constraint (21) can thus be achieved by a stochastically mixing controller with  $n_{\text{jump}} = \nu_{\text{flow}}$ ,  $t_{\text{flow}} = \nu_{\text{jump}}$ ! Since we expect the optimal switching law to be deterministic, this gives a hint of how tight the upper bound can be as function of the polynomial order  $\deg \nu_m \leq N - \deg \theta$ .

The result does not hold in general if we introduce more constraints for the deterministic switching law, such as

$$\theta \varphi_{\text{out}}^{\text{flow}} = 0, \quad \theta \varphi_{\text{in}}^{\text{flow}} \geq 0, \quad \theta \varphi_{\text{ij}} \leq 0,$$

where  $\theta \varphi_{\text{out}}^{\text{flow}} = 0$  holds only in the mixed flow/jump setting. These tighter constraints have been used to produce the upper bounds in the results, except for when the equivalence to random switching has been exploited.

### B. Single parameter sweep: Poisson controller

We will now describe a case when global optimization can be performed by scanning over a single real variable. Consider the upper bound problem with constraints (20), no modal control law  $u_m$  (i.e.  $gW = 0$ ),  $N = 2$  and quadratic threshold  $\theta \in \mathcal{V}_2(\mathcal{X})$  to be optimized. Since  $\deg \nu_m \leq N - \deg \theta = 0$ , the polynomials  $\nu_m$  are constants, e.g.  $\nu_m \in \mathbb{R}$ . The problem can thus be solved globally by sweeping the ratio  $\nu_{\text{flow}} : \nu_{\text{jump}}$  (a common scaling can be accommodated in  $\theta$ ). This is the procedure outlined in [3] for the case of two jump modes with the same  $\Delta T$ .

Since  $\deg \nu_m = 0$ , the upper bound  $\bar{J}$  optimized in this formulation can be achieved by a *Poisson controller*; a random switching controller with state independent switching ratio! Still, the derived threshold  $\theta$  may realize a better cost than the Poisson controller, and can be used as an initial guess for local optimization.

[3] considers also the case with modal control law  $u_{\text{jump}} = -Kx$ . This can be accommodated in our formulation by solving the lower bound problem with Poisson switching constraint, since the solution turns out to be exact for  $N = 2$  in this case.

## VIII. RESULTS

Consider an integrator process (with state  $x \in \mathcal{X} = \mathbb{R}$ )

$$dx = udt + dw, \quad E(dw^2) = dt, \quad (22)$$

where  $w$  is a Wiener Process. The control input  $u$  is a train of Dirac pulses with minimum time between them  $\Delta T = 1$ ,

$$u(t) = \sum_{i=1}^{n_{\text{events}}} u_i \delta(t - t_i), \quad t_{i+1} - t_i \geq \Delta T.$$

We let  $u_i = -x(t_i - 0)$  to immediately reset the state at any control event  $t_i$ . The cost function is

$$j_{\text{acc}} = \int_T x(t)^2 dt + \rho n_{\text{events}},$$

where  $T$  is the interval of time spent in the system. We want to find an event triggering strategy to minimize  $J = \bar{\mathcal{R}}(j_{\text{acc}})$ , the average cost as  $t_{\text{spent}} = |T| \rightarrow \infty$ .

To achieve the minimum inter-event time  $\Delta T$ , the jump mode is constructed as an immediate reset to  $x = 0$ , followed by the dynamics (22) sampled for time  $\Delta T$ . The flow mode is just (22) with control input  $u = 0$ .

Figure 4 shows the optimal average cost  $J$  as a function of event cost  $\rho$  (calculated in [7] for this problem), the cost of periodic control with optimal period  $h \geq \Delta T$ , and lower and upper bounds, which fit quite tightly around the optimum. The upper bound  $\bar{J}_{N=4}$  was found by BMI optimization using the solver PENBMI [10]. The curve  $\bar{J}_{N=6}, \theta_{\text{BMI}_4}$ , calculated with the same thresholds, show that they are in fact almost optimal. The cost of optimal Poisson Sampling lies far above the other bounds, almost coinciding with the cost of periodic control. (In fact, they both choose periodic sampling with  $h = \Delta T$  when  $\rho \leq 0.5$ ) The upper bound  $\bar{J}_{N=6}, \theta_{\text{Poisson}}$  shows that the thresholds from Poisson control are considerably better than the bound.

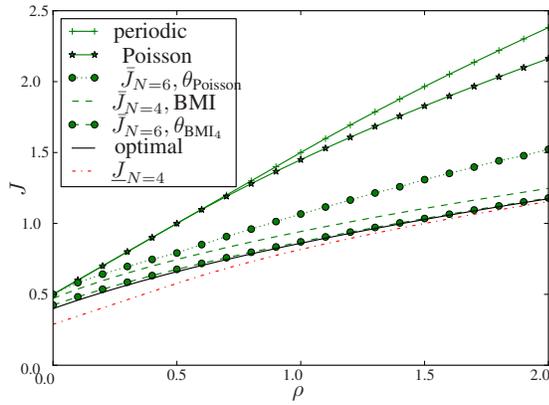


Fig. 4. Cost  $J$  as a function of event cost  $\rho$  for the integrator with  $\Delta T = 1$ .

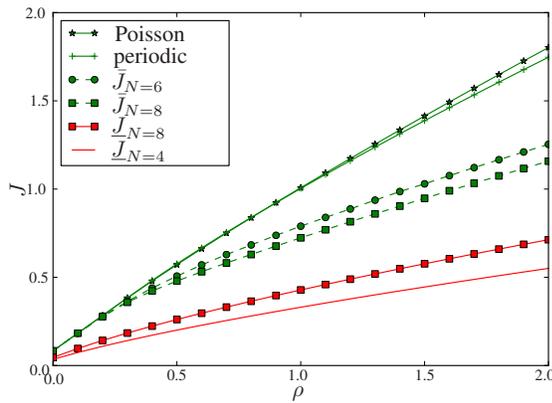


Fig. 5. Cost  $J$  as a function of event cost  $\rho$  for the double integrator with  $\Delta T = 1$ . Thresholds for upper bounds are from Poisson control.

Now consider a double integrator process (with state  $x \in \mathcal{X} = \mathbb{R}^2$ )

$$dx_1 = x_2 dt, \quad dx_2 = u dt + dw, \quad E(dw^2) = dt,$$

with immediate reset to  $x = 0$  at events, minimum time between them  $\Delta T$ , and the cost function

$$j_{\text{acc}} = \int_T x_1(t)^2 dt + \rho n_{\text{events}}.$$

Figure 5 shows upper and lower bounds for the cost  $J$  as a function of event cost  $\rho$ . All upper bounds were found using thresholds from Poisson control. We see that Poisson control and periodic control are comparable, but that the Poisson thresholds perform distinctly better. Still, the gap between upper and lower bounds suggests that there is room to realize a lower cost with better thresholds.

## IX. CONCLUSIONS AND FUTURE WORK

We have modelled a broad class of event based optimal control problems using path constraints, and shown how to derive interval bounds on the control objective from these using convex semidefinite programming. Joint optimization of upper bound and controller parameters is non-convex in general; approaches to it using global and local optimization have been investigated. The examples show that the bounds

are significantly tighter than previous results using quadratic value functions; they also clearly demonstrate that event-based control is superior to periodic control in the examples.

Interesting directions for future work include further case studies and extension to other kinds of stochastic hybrid control problems, improved controller optimization and numerical conditioning.

## ACKNOWLEDGMENT

The author would like to thank A. Cervin, B. Bernhardsson and A. Ghulchak for fruitful discussions on the topic of event-based control, C. Savorgnan for a helpful introduction to polynomial and moment approximations, and M. Kocvara and M. Stingl for letting me use their solver PENBMI.

## REFERENCES

- [1] A. Arapostathis, V. S. Borkar, E. Fernández-Gaucherand, M. K. Ghosh, and S. I. Marcus. Discrete-time controlled Markov processes with average cost criterion: a survey. *SIAM J. Control Optim.*, 31:282–344, March 1993.
- [2] K. J. Åström and B. Bernhardsson. Comparison of periodic and event based sampling for first-order stochastic systems. In *Preprints 14th World Congress of IFAC*, volume J, pages 301–306, Beijing, P.R. China, July 1999.
- [3] R. Cogill. Event-based control using quadratic approximate value functions. In *Conference on Decision and Control*, pages 5883–5888, 2009.
- [4] R. Cogill, S. Lall, and J. P. Hespanha. A constant factor approximation algorithm for optimal estimation subject to communication costs. In *Proc. of the 2007 Amer. Contr. Conf.*, July 2007.
- [5] W. P. M. H. Heemels, J. H. Sandee, and P. P. J. V. D. Bosch. Analysis of event-driven controllers for linear systems. *International Journal of Control*, 81:571–590, 2008.
- [6] T. Henningsson. EventMually toolbox. <http://www.control.lth.se/Research/tools/eventmually.html>.
- [7] T. Henningsson, E. Johannesson, and A. Cervin. Sporadic event-based control of first-order linear stochastic systems. *Automatica*, 44(11):2890–2895, Nov. 2008.
- [8] D. Hristu-Varvakelis and P. Kumar. Interrupt-based feedback control over a shared communication medium. In *Decision and Control, 2002, Proceedings of the 41st IEEE Conference on*, volume 3, pages 3223 – 3228 vol.3, 2002.
- [9] O. C. Imer and T. Basar. Optimal estimation with limited measurements. *Int. J. Syst., Control Commun.*, 2:5–29, January 2010.
- [10] M. Kocvara and M. Stingl. PENBMI. See [www.penopt.com](http://www.penopt.com).
- [11] S. Prajna, A. Papachristodoulou, P. Seiler, and P. A. Parrilo. Sostools and its control applications. In D. Henrion and A. Garulli, editors, *Positive Polynomials in Control*, volume 312 of *Lecture Notes in Control and Information Sciences*, pages 580–580. Springer Berlin / Heidelberg, 2005.
- [12] S. Prajna, P. Parrilo, and A. Rantzer. Nonlinear control synthesis by convex optimization. *IEEE Transactions on Automatic Control*, 49(2):310 – 314, 2004.
- [13] M. Rabi. *Packet based Inference and Control*. PhD thesis, Institute for Systems Research, University of Maryland, 2006. URL: <http://hdl.handle.net/1903/6612>.
- [14] M. Rabi, K. Johansson, and M. Johansson. Optimal stopping for event-triggered sensing and actuation. In *Proc 47th IEEE Conference on Decision and Control*, 2008.
- [15] J. H. Sandee, W. P. M. H. Heemels, and P. P. J. van den Bosch. Case studies in event-driven control. In *Proc. Hybrid Systems: Computation and Control*, pages 762–765, 2007.
- [16] C. Savorgnan, J. B. Lasserre, and M. Diehl. Discrete-time stochastic optimal control via occupation measures and moment relaxations. In *Conference on Decision and Control*, pages 519–524, 2009.
- [17] P. Tabuada. Event-triggered real-time scheduling of stabilizing control tasks. *IEEE Transactions on Automatic Control*, 52(9):1680 –1685, 2007.
- [18] X. Wang and M. Lemmon. Self-triggered feedback control systems with finite-gain  $L_2$  stability. *IEEE Transactions on Automatic Control*, 54(3):452 –467, 2009.