Stability of Nonlinear Control Systems Under Intermittent Information with Applications to Multi-Agent Robotics

Domagoj Tolić

Fakultet Elektrotehnike i Računarstva
Zagreb, Croatia

ACROSS Colloquia
26.10.2012.

FER

ACROSS
Centre of Research Excellence
for Advanced Cooperative Systems

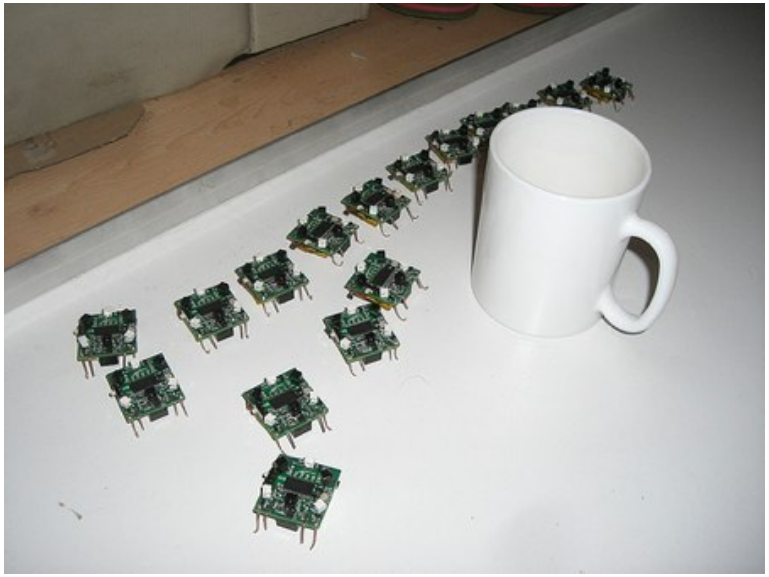
Outline

1. Introduction
   - Motivation
   - Intermittent Information
   - Recent Ideas
2. Stability
   - Problem Formulation
   - Approach
   - Numerical Results
3. Optimal Intermittent Fdbk
   - Motivation
   - Approach
   - Numerical Results
4. Summary

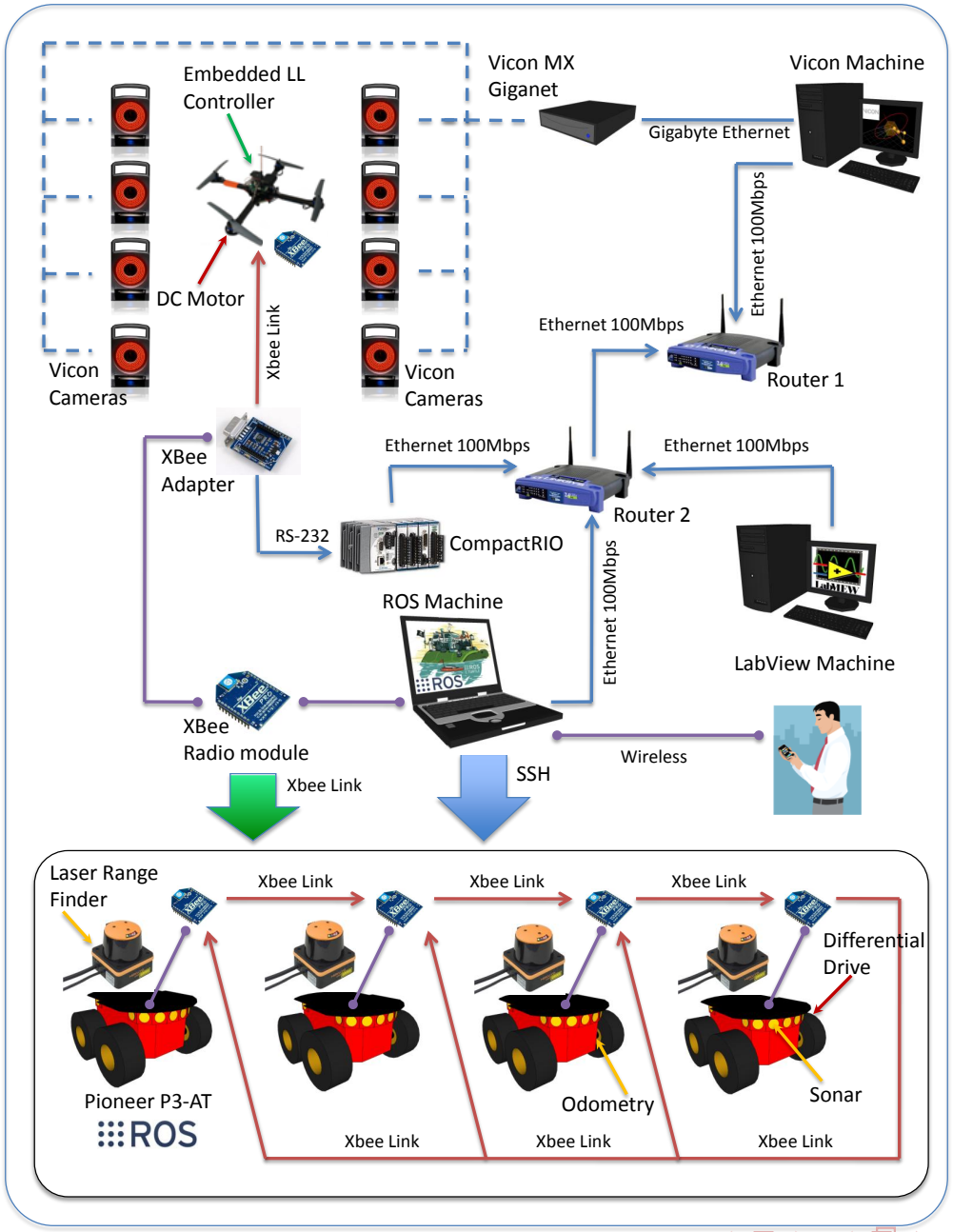
Outline

1. Introduction
   - Motivation
   - Intermittent Information
   - Recent Ideas
2. Stability
3. Optimal Intermittent Fdbk
4. Summary

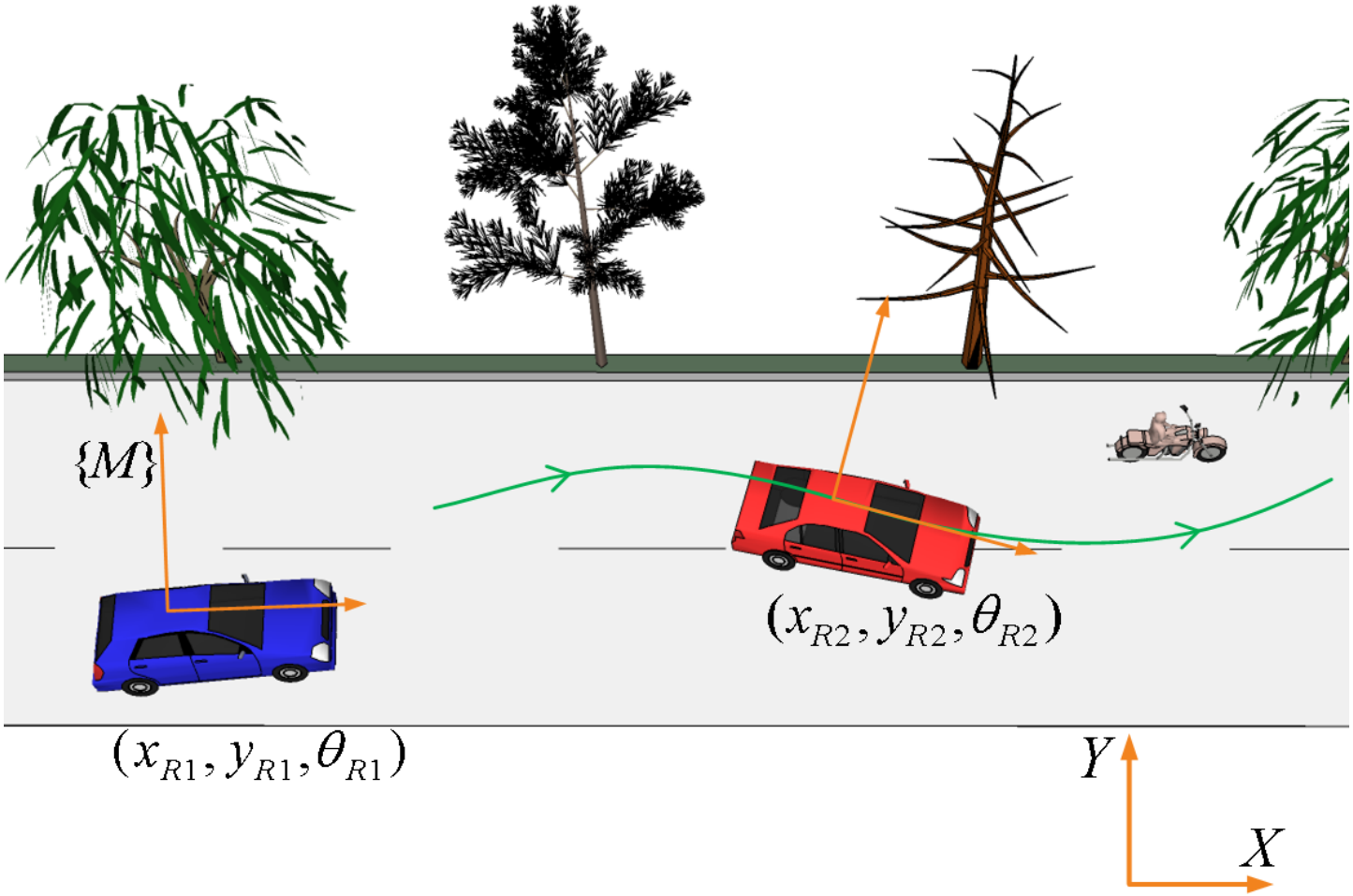
Multi-Agent Systems



Systems of Systems (Heterogeneity)



Illustrative Example



What Do We Mean by Intermittent Information?

- How often should **up-to-date information** about systems be collected and sent to **neighbors**?

Intrinsic Properties

- packet collisions, sampling period, network throughput, scheduling protocols, occlusions of sensors, a limited communication/sensing range, etc.

What Do We Mean by Intermittent Information?

- How often should **up-to-date information** about systems be collected and sent to **neighbors**?

Intrinsic Properties

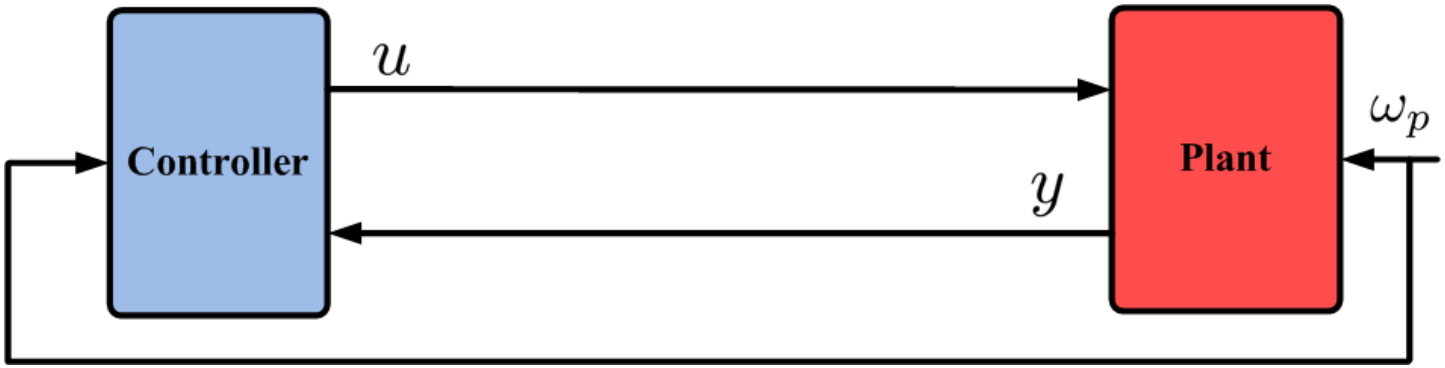
- packet collisions, sampling period, network throughput, scheduling protocols, occlusions of sensors, a limited communication/sensing range, etc.

Intermittent Feedback

- a **user-designed** property of a system
- the goal is to **decrease energy consumption** as well as **processing** and **sensing requirements**
- **flexibility** and **resource efficiency**

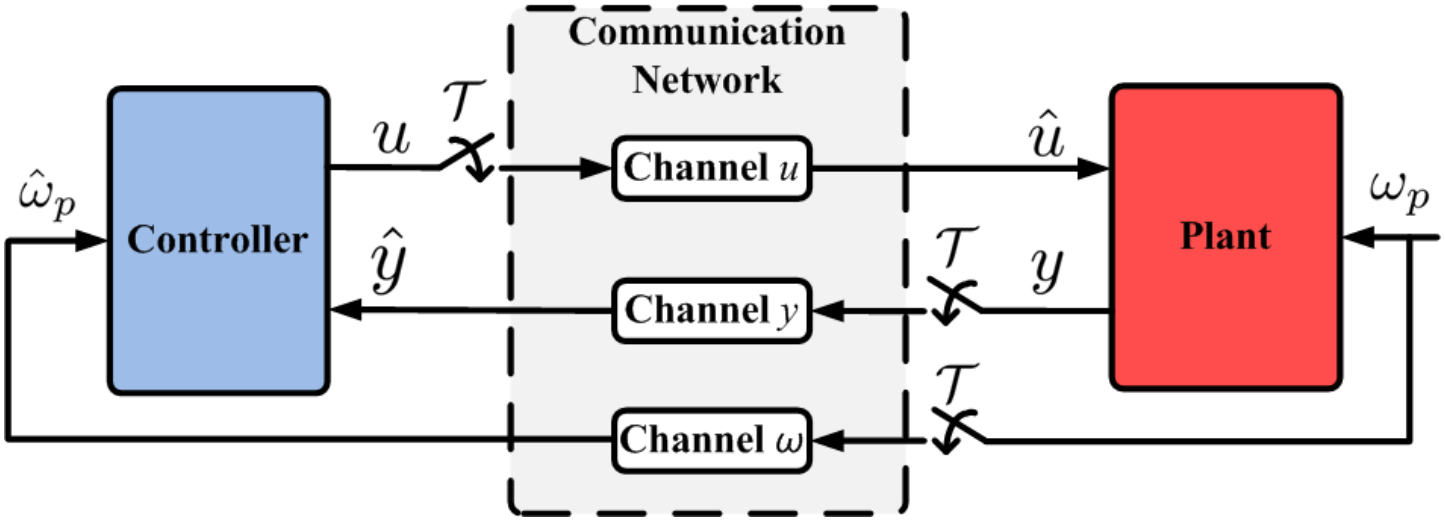
Why Do We Care?

When designing controllers we think of:



Why Do We Care?

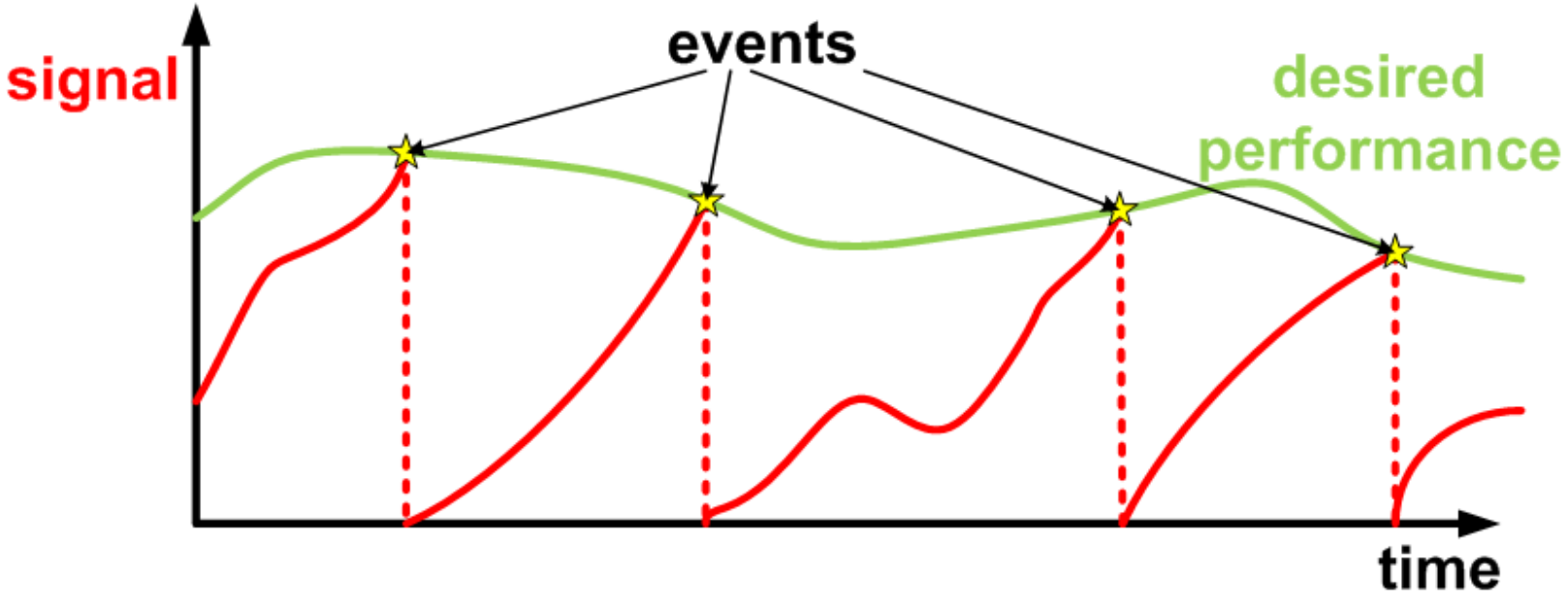
What we really have is:



Event-Triggering vs. Self-Triggering

Event-Triggering

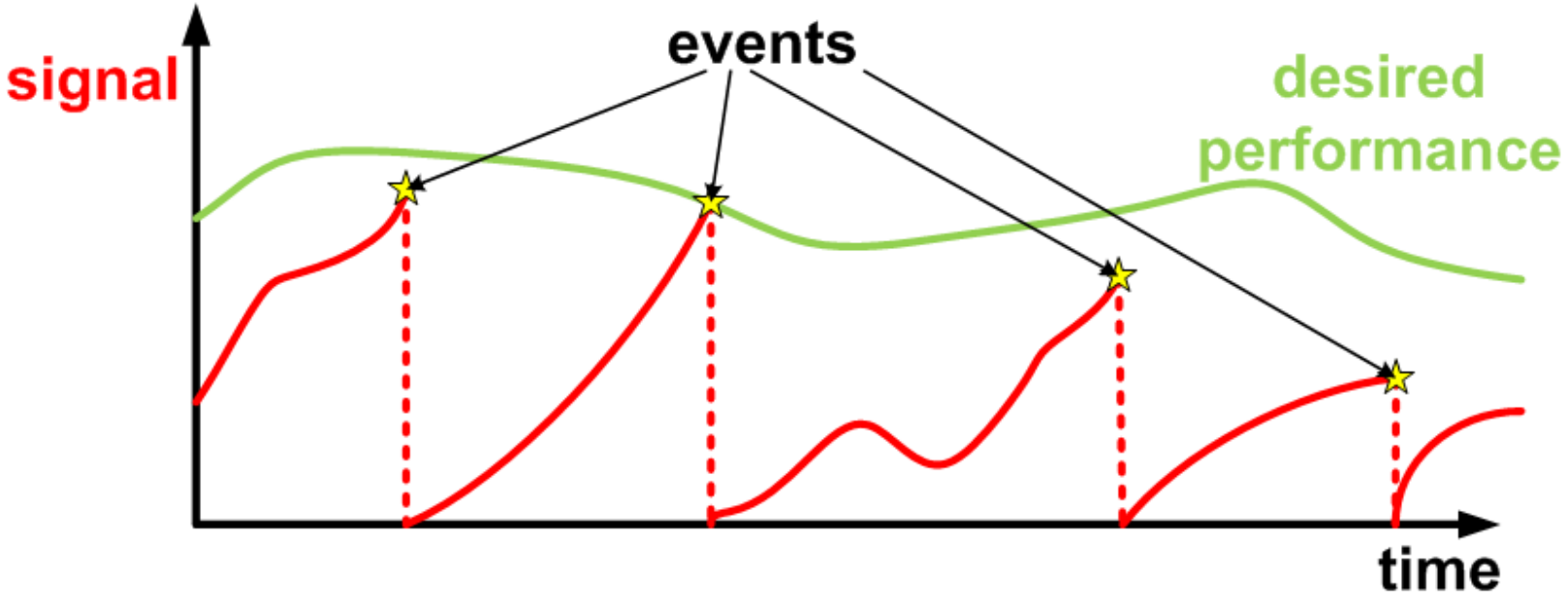
- sampling (i.e., transmission of up-to-date information) is triggered when a **triggering event** occurs



Event-Triggering vs. Self-Triggering

Self-Triggering

- the **current sampling instance** is used to **predict** triggering events and **preclude** undesired performance



Outline

1. Introduction
2. Stability
   - Problem Formulation
   - Approach
   - Numerical Results
3. Optimal Intermittent Fdbk
4. Summary

Related Work

1. Small-gain theorem approaches [Nešić and Teel, 2004], [Tabbara et al., 2007];

Related Work

1. Small-gain theorem approaches [Nešić and Teel, 2004], [Tabbara et al., 2007];
2. Dissipativity or passivity-based approaches [Yu and Antsaklis, 2011a], [Yu and Antsaklis, 2011b];

Related Work

1. Small-gain theorem approaches [Nešić and Teel, 2004], [Tabbara et al., 2007];
2. Dissipativity or passivity-based approaches [Yu and Antsaklis, 2011a], [Yu and Antsaklis, 2011b];
3. Input-to-State Stability (ISS) approaches [Tabuada, 2007], [Anta and Tabuada, 2010], [Lemmon, 2010]; and

Related Work

1. Small-gain theorem approaches [Nešić and Teel, 2004], [Tabbara et al., 2007];
2. Dissipativity or passivity-based approaches [Yu and Antsaklis, 2011a], [Yu and Antsaklis, 2011b];
3. Input-to-State Stability (ISS) approaches [Tabuada, 2007], [Anta and Tabuada, 2010], [Lemmon, 2010]; and
4. Other approaches [Estrada and Antsaklis, 2008], [Li et al., 2007], [Tallapragada and Chopra, 2011].

Closed-Loop Systems

- **first** – design a closed-loop system without taking into account communication networks

  – *plant*:

$$\begin{aligned} \dot{x}_p &= f_p(t, x_p, u, \omega_p), \\ y &= g_p(t, x_p), \end{aligned} \tag{1}$$

  – *controller*:

$$\begin{aligned} \dot{x}_c &= f_c(t, x_c, y, \omega_c), \\ u &= g_c(t, x_c) \end{aligned} \tag{2}$$

- **second** – introduce communication networks

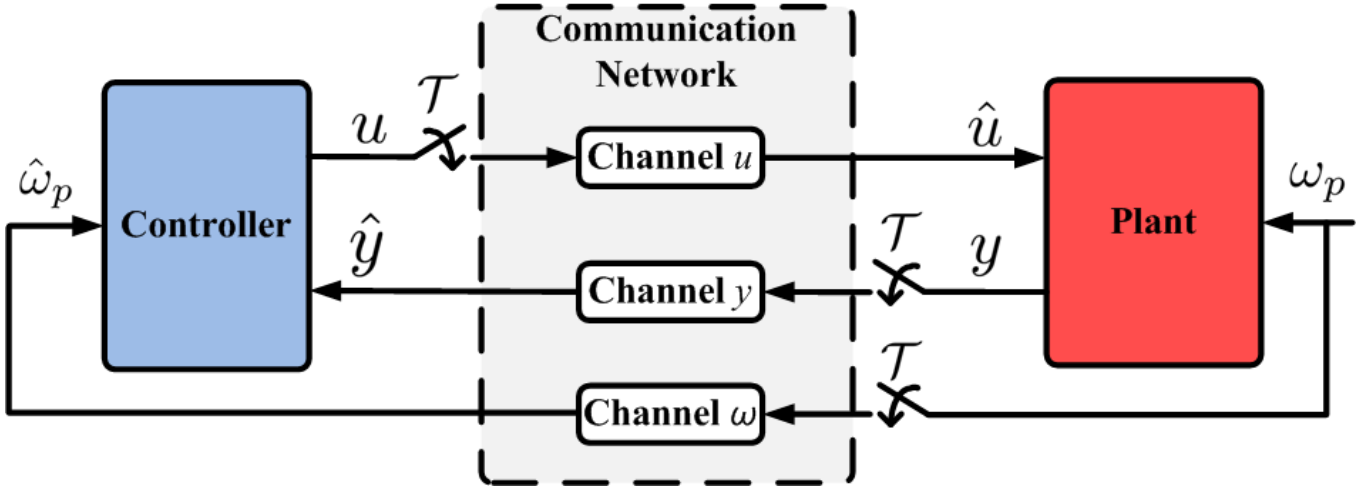
Introducing Communication Networks

- **output error vector**:

$$e := \begin{bmatrix} \hat{y} - y \\ \hat{u} - u \end{bmatrix} =: \begin{bmatrix} e_y \\ e_u \end{bmatrix} \tag{3}$$

- **input error vector**:

$$e_\omega := \hat{\omega}_p - \omega_p \tag{4}$$



Problem Statement

Input-Output Triggering

Based on the value of $\hat{y}$ and $\hat{\omega}_p$ received at t_i, $i \in \mathbb{N}$, find a **time interval** τ_i **until the next transmission** instant of y and ω_p yielding the closed-loop system (1)-(2) **stable in some appropriate sense**.

Contributions

1. The design of an **input-output triggered** sampling policy employing the **small-gain theorem**;

Contributions

1. The design of an **input-output triggered** sampling policy employing the **small-gain theorem**;
2. Consideration of **external inputs**, and **realistic** communication channels and sensors;

Contributions

1. The design of an **input-output triggered** sampling policy employing the **small-gain theorem**;
2. Consideration of **external inputs**, and **realistic** communication channels and sensors;
3. The formulation of novel conditions for $\mathcal{L}_p$**-stability of hybrid systems**; and

Contributions

1. The design of an **input-output triggered** sampling policy employing the **small-gain theorem**;
2. Consideration of **external inputs**, and **realistic** communication channels and sensors;
3. The formulation of novel conditions for $\mathcal{L}_p$**-stability of hybrid systems**; and
4. The design of a novel method for **computing** $\mathcal{L}_p$-gains over a **finite horizon** by resorting to **convex programming**.

Contributions

1. The design of an **input-output triggered** sampling policy employing the **small-gain theorem**;
2. Consideration of **external inputs**, and **realistic** communication channels and sensors;
3. The formulation of novel conditions for $\mathcal{L}_p$**-stability of hybrid systems**; and
4. The design of a novel method for **computing** $\mathcal{L}_p$-gains over a **finite horizon** by resorting to **convex programming**.

Our approach does not require construction of storage nor Lyapunov functions.

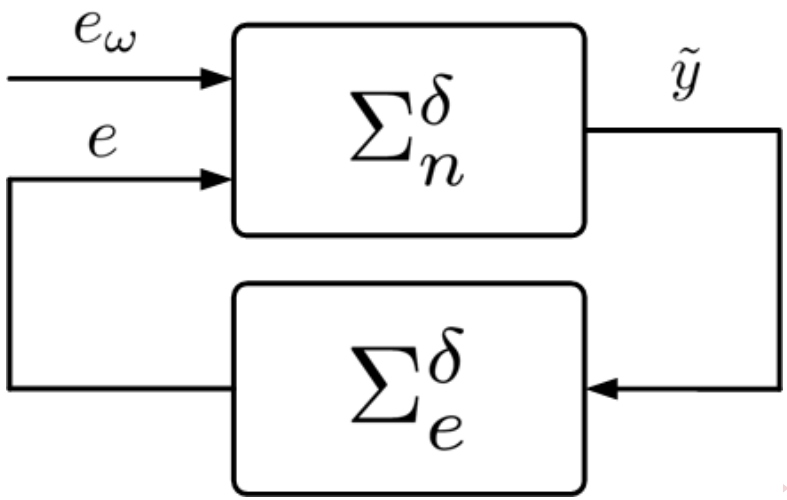
Interconnecting Nominal and Error System I

We write (1)-(2) as the following interconnected hybrid system [Sanfelice, 2007]:

$$\left.\begin{aligned} \dot{x} &= f^{\delta}(t, x, e, e_{\omega}) \\ \dot{e} &= g^{\delta}(t, x, e, e_{\omega}) \end{aligned}\right\} \quad t \in \bigcup_{i \in \mathbb{N}_0} [t_i, t_{i+1}), \tag{5a}$$

$$\left.\begin{aligned} x(t^+) &= x(t) \\ e(t^+) &= h(t, e(t)) \end{aligned}\right\} \quad t \in \mathcal{T}. \tag{5b}$$

where $\delta = \hat{\omega}_p$ and $x = (x_p, x_c)$.



Interconnecting Nominal and Error System II

$$f(t,x,e,\hat{\omega}_p,e_\omega) := \begin{bmatrix} f_p(t,x_p,g_c(t,x_c)+e_u,\hat{\omega}_p-e_\omega) \\ f_c(t,x_c,g_p(t,x_p)+e_y,\hat{\omega}_p) \end{bmatrix},$$

$$h(t_i,e) := \begin{bmatrix} h_y(t_i,e(t_i)) \\ h_u(t_i,e(t_i)) \end{bmatrix},$$

$$g(t,x,e,\hat{\omega}_p,e_\omega) :=$$

$$\begin{bmatrix} \underbrace{\hat{f}_p(t,x_p,x_c,g_p(t,x_p)+e_y,g_c(t,x_c)+e_u,\hat{\omega}_p-e_\omega)}_{\equiv 0 \text{ for zero-order-hold estimation strategy}} - \frac{\partial g_p}{\partial t}(t,x_p) - \frac{\partial g_p}{\partial x_p}(t,x_p) f_p(t,x_p,g_c(t,x_c)+e_u,\hat{\omega}_p-e_\omega) \\ \overbrace{\hat{f}_c(t,x_p,x_c,g_p(t,x_p)+e_y,g_c(t,x_c)+e_u,\hat{\omega}_p)} - \frac{\partial g_c}{\partial t}(t,x_c) - \frac{\partial g_c}{\partial x_c}(t,x_c) f_c(t,x_p,g_p(t,x_p)+e_y,\hat{\omega}_p) \end{bmatrix}$$

$\mathcal{L}_p$-stability of Switched Systems I

Theorem

Consider a hybrid system Σ^δ. Let $K \geq 0$ and $p \in [1, \infty)$. If δ is such that

(i) *There exist constants $\widetilde{K}(\tau_i^\delta)$, $\widetilde{\gamma}(\tau_i^\delta)$ such that*

$$\|y[t_i^\delta, t']\|_p \leq \widetilde{K}(\tau_i^\delta)\|x(t_i^{\delta+})\| + \widetilde{\gamma}(\tau_i^\delta)\|\omega[t_i^\delta, t']\|_p. \tag{6}$$

for all $t' \in [t_i^\delta, t_{i+1}^\delta]$ and all $i \in \mathbb{N}_0$, where $\tau_i^\delta = t_{i+1}^\delta - t_i^\delta$, and such that

$$K_M := \sup_{i \in \mathbb{N}_0} \widetilde{K}(\tau_i^\delta), \tag{7}$$

$$\gamma_M := \sup_{i \in \mathbb{N}_0} \widetilde{\gamma}(\tau_i^\delta), \tag{8}$$

exist, and

$\mathcal{L}_p$-stability of Switched Systems II

Theorem

(ii) *The condition*

$$\sum_{i=1}^{\infty} \|x(t_i^{\delta+})\| \leq K\|x(t_0)\|, \tag{9}$$

holds,

then Σ^δ *is* $\mathcal{L}_p$*-stable from* ω *to* y *with constant* $K_M(K+1)$ *and gain* γ_M *for the given* δ*. For* $p = \infty$*, the same result holds with the constant* $K_M K$ *and gain* γ_M *when (9) is replaced with* $\sup_{i\in\mathbb{N}} \|x(t_i^{\delta+})\| \leq K\|x(t_0)\|$*. In addition, if the state* x *is* $\mathcal{L}_p$ *to* $\mathcal{L}_p$ *detectable, then conditions (i) and (ii) are both sufficient and necessary.*

$\mathcal{L}_p$-stability Over a Finite Horizon for Error System

Theorem

Assume $\delta = \hat{\omega}_p$ is fixed to be constant. Suppose that there exist $A \in \mathcal{A}_{n_e}^+$ such that $\|A\| < \infty$, and a continuous function $\tilde{y} : \mathbb{R} \times \mathbb{R}^{n_x} \times \mathbb{R}^{n_\omega} \times \mathbb{R}^{n_\omega} \to \mathbb{R}_+^{n_e}$ such that the output error dynamics in (5a) satisfies

$$\dot{\bar{e}} = \overline{g(t, x, e, \hat{\omega}_p, e_\omega)} \preceq A\bar{e} + \tilde{y}(t, x, \hat{\omega}_p, e_\omega) \tag{10}$$

for all $e \in \mathbb{R}^{n_e}$ and all $(t, x(t), \hat{\omega}_p, e_\omega(t)) \in \mathcal{S}$ provided that $t \in [t_0, t_0 + \tau]$, where $\mathcal{S} \subseteq \mathbb{R} \times \mathbb{R}^{n_x} \times \mathbb{R}^{n_\omega} \times \mathbb{R}^{n_\omega}$. Then, the output error system is $\mathcal{L}_p$-stable from $\tilde{y}$ to e over a finite horizon $\tau \geq 0$ for any $p \in [1, \infty]$ with

$$\widetilde{K}_e(\tau) = \left(\frac{\exp(\|A\| p\tau) - 1}{p\|A\|} \right)^{1/p}, \tag{11}$$

$$\widetilde{\gamma}_e(\tau) = \frac{\exp(\|A\|\tau) - 1}{\|A\|}. \tag{12}$$

Input-Output Triggering

- **stabilizing intersampling intervals** are given by

$$\tau_i^* = \frac{1}{\|A_i^\delta\|} \ln\left(\kappa \frac{\|A_i^\delta\|}{\gamma_n^\delta} + 1\right) \Longleftrightarrow \gamma_n^\delta \tilde{\gamma}_e(\tau_i^*) = \kappa, \tag{13}$$

where $\kappa \in (0, 1)$.

Input-Output Triggering

Theorem

For $p \in [1, \infty]$ *assume that*

(i) *There exists* $L \geq 0$ *such that* $\overline{\omega}_p \preceq L\, \mathbf{1}_{n_\omega}$,

(ii) *Theorem 3 holds,*

(iii) Σ_n^δ *is* $\mathcal{L}_p$*-stable from* (e_ω, e) *to* $\tilde{y}$ *for every* $\delta \equiv c$, $c \in \mathcal{P}$, *with gain* γ_n^δ *that has an upper- and lower-bound,*

(iv) *Sampling policy (13) yields* x *that satisfies (9) for given* $\delta : [t_0, \infty) \to \mathcal{P}$, *and*

(v) x *is* $\mathcal{L}_p$ *to* $\mathcal{L}_p$ *detectable from* $(\tilde{y}, e_\omega, e)$.

Then, the closed-loop system (1)-(2) is $\mathcal{L}_p$*-stable from* e_ω *to* (x, e) *for given* δ.

Trajectory Tracking

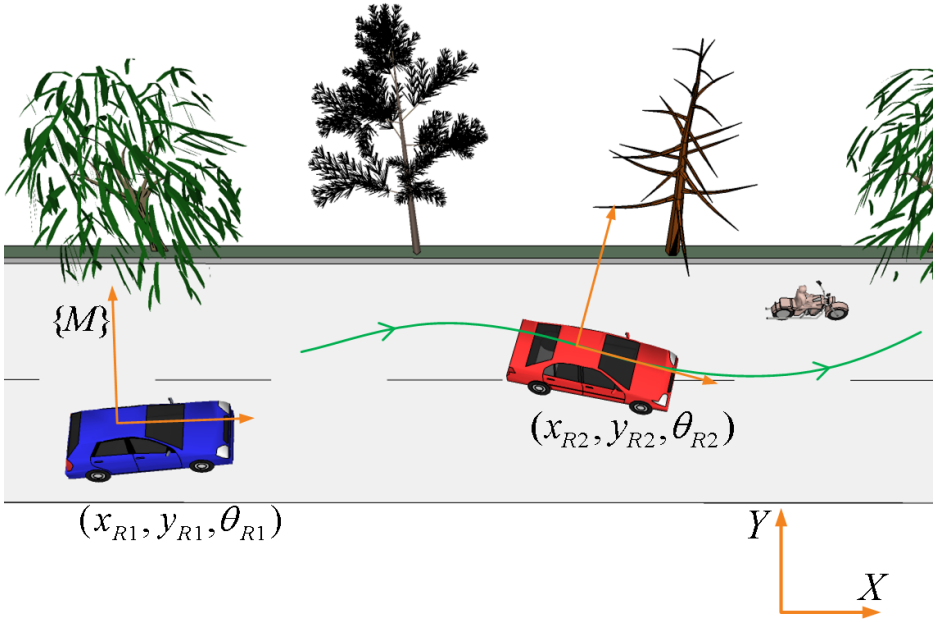
– **tracking error**:

$$\dot{x}_p = \begin{bmatrix} \omega_{R1} x_{p2} - v_{R1} + v_{R2} \cos x_{p3} \\ -\omega_{R1} x_{p1} + v_{R2} \sin x_{p3} \\ \omega_{R2} - \omega_{R1} \end{bmatrix}.$$

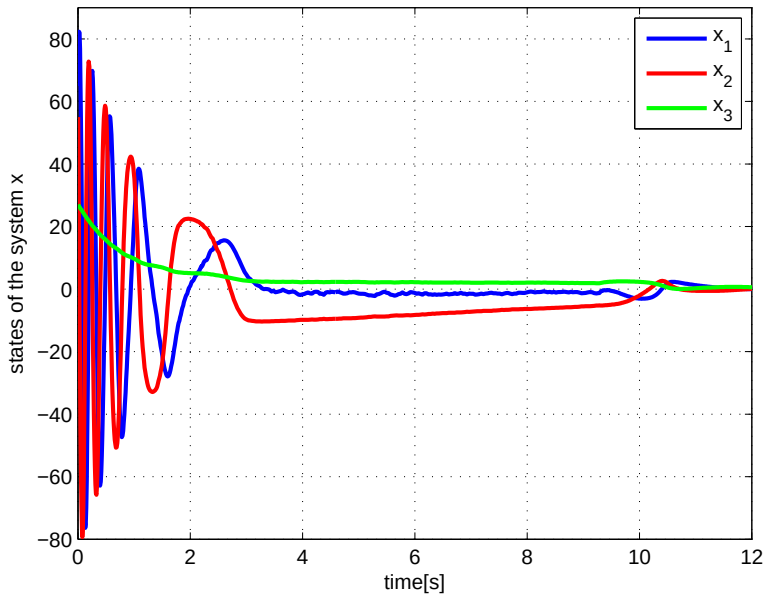
– **controller**:

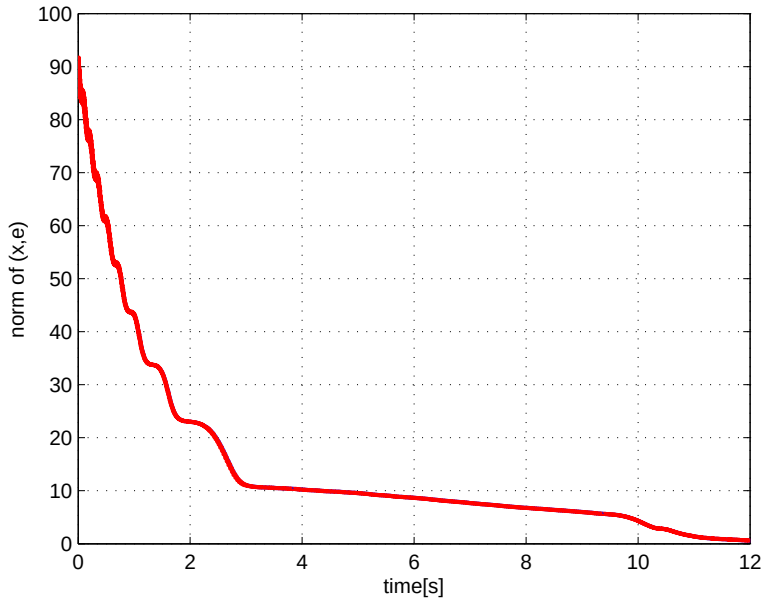
$$v_{R1} = v_{R2} \cos x_{p3} + k_1 x_{p1},$$

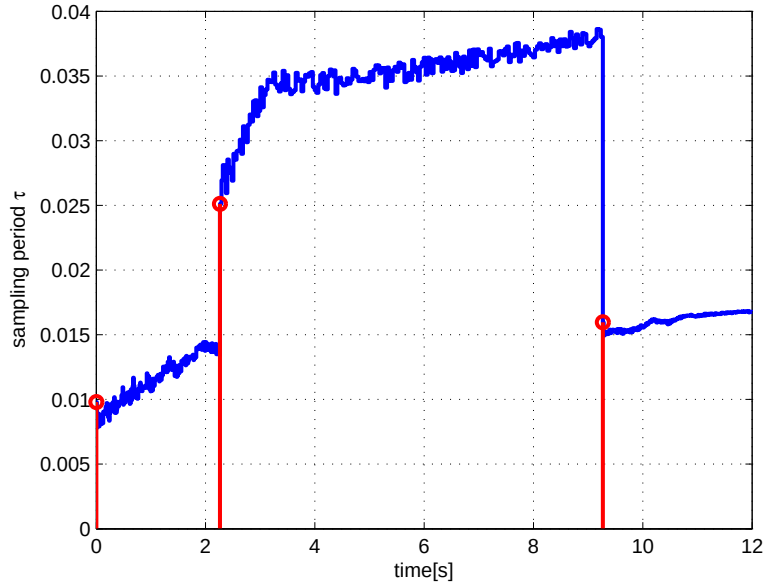
$$\omega_{R1} = \omega_{R2} + k_2 v_{R2} \frac{\sin x_{p3}}{x_{p3}} x_{p2} + k_3 x_{p3}.$$

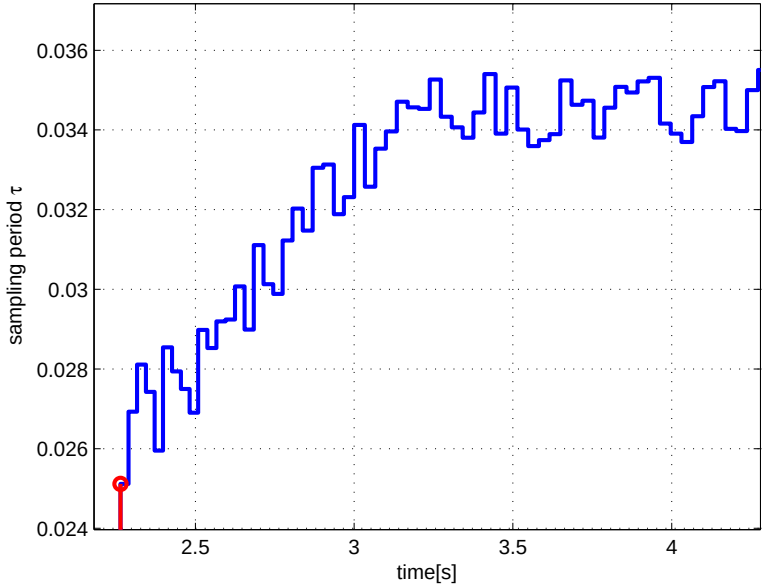


Trajectory Tracking - with Switched Modeling

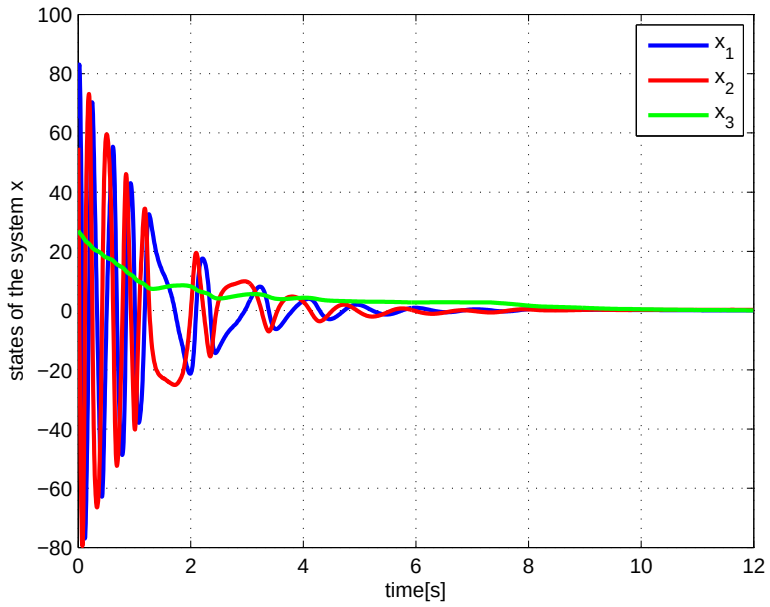


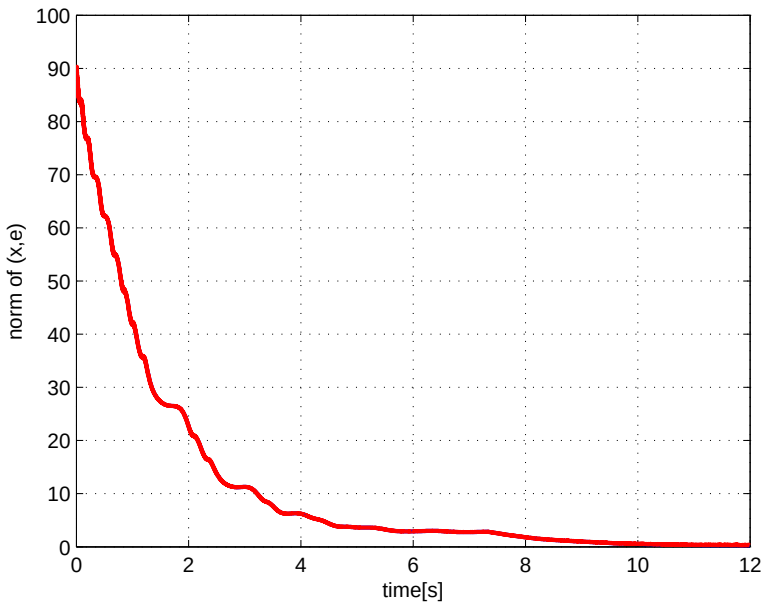


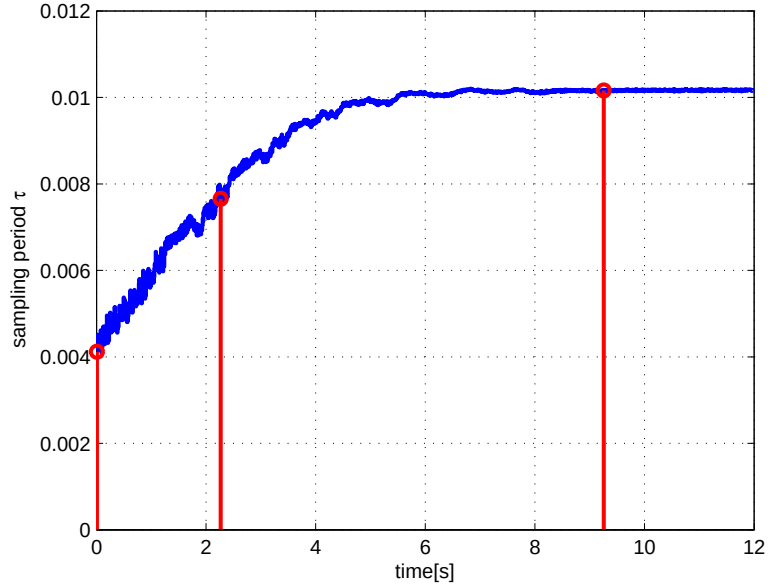




Trajectory Tracking - without Switched Modeling







Outline

1. Introduction
2. Stability
3. Optimal Intermittent Fdbk
   - Motivation
   - Approach
   - Numerical Results
4. Summary

What do we Mean by Optimal Intermittent Feedback?

- think of an airplane driven by an autopilot system designed to follow the **shortest path** between two points
- any **deviation** from the shortest path caused by intermittent feedback **increases** total **fuel consumption**
- this increase in fuel consumption is probably **more costly** than the cost of energy saved due to intermittent feedback

What do we Mean by Optimal Intermittent Feedback?

- think of an airplane driven by an autopilot system designed to follow the **shortest path** between two points
- any **deviation** from the shortest path caused by intermittent feedback **increases** total **fuel consumption**
- this increase in fuel consumption is probably **more costly** than the cost of energy saved due to intermittent feedback
- we encode these **energy consumption trade-offs** in a cost function, and design an **Approximate Dynamic Programming** (ADP) approach that yields **optimal intertransmission intervals** with respect to the cost function

Contributions

1. Formulation of the **optimal self-triggering problem** as a Dynamic Programming (DP) problem;

Contributions

1. Formulation of the **optimal self-triggering problem** as a Dynamic Programming (DP) problem;

2. Employment of **Particle Filters** (PFs) fed by intermittent feedback to account for **partially observable states**; and

Contributions

1. Formulation of the **optimal self-triggering problem** as a Dynamic Programming (DP) problem;
2. Employment of **Particle Filters** (PFs) fed by intermittent feedback to account for **partially observable states**; and
3. Formulation of properties that **successful approximation architectures** in ADP approaches satisfy.

Contributions

1. Formulation of the **optimal self-triggering problem** as a Dynamic Programming (DP) problem;
2. Employment of **Particle Filters** (PFs) fed by intermittent feedback to account for **partially observable states**; and
3. Formulation of properties that **successful approximation architectures** in ADP approaches satisfy.

- Tolić, D.; Fierro, R.; Ferrari, S.; , "Optimal Self-Triggering for Nonlinear Systems via Approximate Dynamic Programming," 2012 IEEE Multiconference on Systems and Control (MSC 2012), 2012, accepted for publication

Performance Index - Dynamic Programming I

Goal

Minimize the following **cost function** $V : \mathbb{R}^{n_x} \to \mathbb{R}$

$$V_{\tau_i}(x_0) = \mathop{\mathbb{E}}_{e_\omega}\left\{ \sum_{i=1}^{\infty} \gamma^i \Big[\underbrace{\int_{t_{i-1}}^{t_i} (x_p^T Q x_p + u^T R u)\mathrm{d}t + S}_{l(x_p,u,\tau_i)} \Big] \right\} \tag{14}$$

over all **sampling policies** τ_i and for all initial conditions $x_0 \in \mathbb{R}^{n_x}$.

Performance Index - Dynamic Programming I

Goal

Minimize the following **cost function** $V : \mathbb{R}^{n_x} \to \mathbb{R}$

$$V_{\tau_i}(x_0) = \mathop{\mathbb{E}}_{e_\omega}\left\{ \sum_{i=1}^{\infty} \gamma^i \Big[\underbrace{\int_{t_{i-1}}^{t_i} (x_p^T Q x_p + u^T R u)\mathrm{d}t + S}_{l(x_p,u,\tau_i)} \Big] \right\} \tag{14}$$

over all **sampling policies** τ_i and for all initial conditions $x_0 \in \mathbb{R}^{n_x}$.

This cost function captures **performance vs. energy** trade-offs.

Performance Index - Dynamic Programming II

- minimization of (14) is equivalent to **Hamilton-Jacobi-Bellman equation**

$$V^*(z) = \inf_{\tau \in [0, \tau^{\max}]} \Big(l(z, u, \tau) + \gamma \mathop{\mathbb{E}}_{e_\omega} \{ V^*(f(z, u, \tau, \hat{\omega}_p, e_\omega)) \} \Big).$$

Performance Index - Dynamic Programming II

- minimization of (14) is equivalent to **Hamilton-Jacobi-Bellman equation**

$$V^*(z) = \inf_{\tau \in [0, \tau^{\max}]} \Big(l(z, u, \tau) + \gamma \mathop{\mathbb{E}}_{e_\omega} \{ V^*(f(z, u, \tau, \hat{\omega}_p, e_\omega)) \} \Big).$$

- $V^*(z)$ is called the **optimal value function** (or optimal cost-to-go function)
- $V^*(z)$ represents the **cost** incurred by an **optimal policy** τ^* when the initial condition in (14) is z.

Performance Index - Dynamic Programming III

- introduce the **Bellman operator** $\mathcal{M}$ as

$$\mathcal{M}g = (\mathcal{M}g)(z) = \inf_{\tau \in [0, \tau^{\max}]} \left(l(z, u, \tau) + \gamma \mathop{\mathbb{E}}_{e_\omega} \{ g(f(z, u, \tau, \hat{\omega}_p, e_\omega)) \} \right)$$

for any $g : \mathbb{R}^{n_x} \to \mathbb{R}$

Performance Index - Dynamic Programming III

- introduce the **Bellman operator** $\mathcal{M}$ as

$$\mathcal{M}g = (\mathcal{M}g)(z) = \inf_{\tau \in [0, \tau^{\max}]} \left(l(z, u, \tau) + \gamma \mathop{\mathbb{E}}_{e_\omega} \{ g(f(z, u, \tau, \hat{\omega}_p, e_\omega)) \} \right)$$

for any $g : \mathbb{R}^{n_x} \to \mathbb{R}$

- since $\gamma \in (0, 1)$, therefore $\mathcal{M}$ is a **contraction**, i.e.,

$$\|\mathcal{M}u - \mathcal{M}v\|_s \leq \gamma \|u - v\|_s$$

where $\|v\|_s = \sup_{z \in \mathbb{R}^{n_p}} v(z)$

Performance Index - Dynamic Programming III

- introduce the **Bellman operator** $\mathcal{M}$ as

$$\mathcal{M}g = (\mathcal{M}g)(z) = \inf_{\tau \in [0, \tau^{\max}]} \left(l(z, u, \tau) + \gamma \mathop{\mathbb{E}}_{e_\omega} \{ g(f(z, u, \tau, \hat{\omega}_p, e_\omega)) \} \right)$$

  for any $g : \mathbb{R}^{n_x} \to \mathbb{R}$
- since $\gamma \in (0, 1)$, therefore $\mathcal{M}$ is a **contraction**, i.e.,

$$\|\mathcal{M}u - \mathcal{M}v\|_s \leq \gamma \|u - v\|_s$$

  where $\|v\|_s = \sup_{z \in \mathbb{R}^{n_p}} v(z)$
- the set $\mathcal{B}$ of all bounded, real valued functions with the norm $\| \cdot \|_s$ is a **Banach space**

Performance Index - Dynamic Programming III

- introduce the **Bellman operator** $\mathcal{M}$ as

$$\mathcal{M}g = (\mathcal{M}g)(z) = \inf_{\tau \in [0, \tau^{\max}]} \left(l(z, u, \tau) + \gamma \mathop{\mathbb{E}}_{e_\omega} \{ g(f(z, u, \tau, \hat{\omega}_p, e_\omega)) \} \right)$$

  for any $g : \mathbb{R}^{n_x} \to \mathbb{R}$
- since $\gamma \in (0, 1)$, therefore $\mathcal{M}$ is a **contraction**, i.e.,

$$\|\mathcal{M}u - \mathcal{M}v\|_s \leq \gamma \|u - v\|_s$$

  where $\|v\|_s = \sup_{z \in \mathbb{R}^{n_p}} v(z)$
- the set $\mathcal{B}$ of all bounded, real valued functions with the norm $\| \cdot \|_s$ is a **Banach space**
- therefore, for each initial $V^0 \in \mathcal{B}$, the **sequence of value functions** $V^{n+1} = \mathcal{M}V^n = \mathcal{M}^{n+1}V^0$ converges to V^*

Issues

1. "Curses of dimensionality"

$$(\mathcal{M}g)(z) = \inf_{\tau \in [0,\tau^{\max}]} \left(l(z,u,\tau) + \gamma \mathop{\mathbb{E}}_{e_\omega} \{ g(f(z,u,\tau,\hat{\omega}_p, e_\omega)) \} \right)$$

2. Approximation architecture

3. Partially observable states

Issues

1. "Curses of dimensionality"

$$(\mathcal{M}g)(z) = \inf_{\tau \in [0,\tau^{\max}]} \left(l(z,u,\tau) + \gamma \mathop{\mathbb{E}}_{e_\omega} \{ g(f(z,u,\tau,\hat{\omega}_p, e_\omega)) \} \right)$$

2. Approximation architecture

3. Partially observable states

This is why we use **Approximate Dynamic Programming**.

Curses of Dimensionality

$$(\mathcal{M}g)(z) = \inf_{\tau \in [0,\tau^{\max}]} \left(l(z,u,\tau) + \gamma \mathop{\mathbb{E}}_{e_\omega} \{ g(f(z,u,\tau,\hat{\omega}_p, e_\omega)) \} \right)$$

1. Uncountable and multi-dimensional state space
   –> choose a **finite set** of points $\mathcal{X} \subset \mathcal{C}_x \subset \mathbb{R}^{n_x}$

2. Computing expectation
   –> a sum of a **quadrature approximation** (e.g., Simpson formula)

3. Optimization
   –> **gradient search** methods with constraints starting from different initial points

Approximation Architecture I

- introduce an approximate value function $\hat{V}^i$ of V^i
- compute $\hat{V}^{i+1}$ only for the points in $\mathcal{X}$
- **generalize/interpolate** for $\hat{V}^{i+1}$ for $\mathcal{C}_x \setminus \mathcal{X}$

Approximation Architecture I

- introduce an approximate value function $\hat{V}^i$ of V^i
- compute $\hat{V}^{i+1}$ only for the points in $\mathcal{X}$
- **generalize/interpolate** for $\hat{V}^{i+1}$ for $\mathcal{C}_x \setminus \mathcal{X}$

Properties

(i) $\hat{V}^{i+1}(x') = (\mathcal{M}\hat{V}^i)(x')$;

(ii) $\mathrm{supp}(\hat{V}^{i+1} - \hat{V}^i) = \mathcal{C}_i$, where $\mathcal{C}_i \subset \mathcal{C}_x$ is a convex and compact neighborhood of x'; and

(iii) for any $c \in \partial\mathcal{C}_i$ the following holds

$$\hat{V}^{i+1}[S] \subseteq [\hat{V}^{i+1}(c), \hat{V}^{i+1}(x')],$$

where $\hat{V}^{i+1}[S]$ is the image of the segment $\mathcal{S}$ connecting x' and c.

Approximation Architecture II

- **online learning** of Neural Networks (NNs)

Approximation Architecture II

- **online learning** of Neural Networks (NNs)
- for **batch learning**, properties (i), (ii) and (iii) cannot be guaranteed since NNs are **expansion approximators**

Approximation Architecture II

- **online learning** of Neural Networks (NNs)
- for **batch learning**, properties (i), (ii) and (iii) cannot be guaranteed since NNs are **expansion approximators**
- exceptions are **kernel-based** NNs and **recurrent** NNs in certain settings

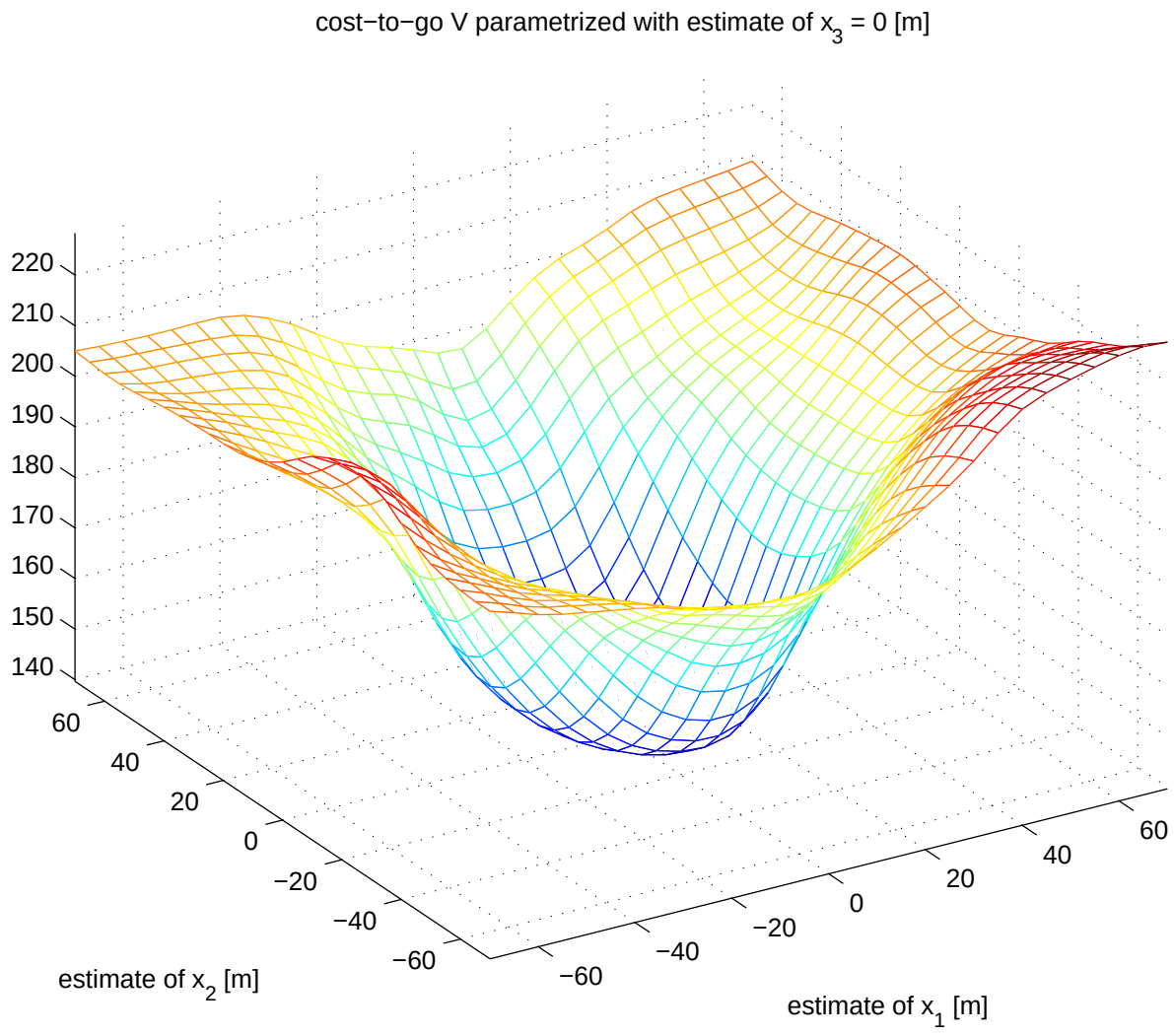
Approximation Architecture II

- **online learning** of Neural Networks (NNs)
- for **batch learning**, properties (i), (ii) and (iii) cannot be guaranteed since NNs are **expansion approximators**
- exceptions are **kernel-based** NNs and **recurrent** NNs in certain settings
- **randomly** pick points $x_i' \in \mathcal{C}_x$ in each step
  - we do not have to specify $\mathcal{X}$
  - we avoid the problem of **exploration vs. exploitation** (see [Powell, 2007] and [Sutton and Barto, 1998])

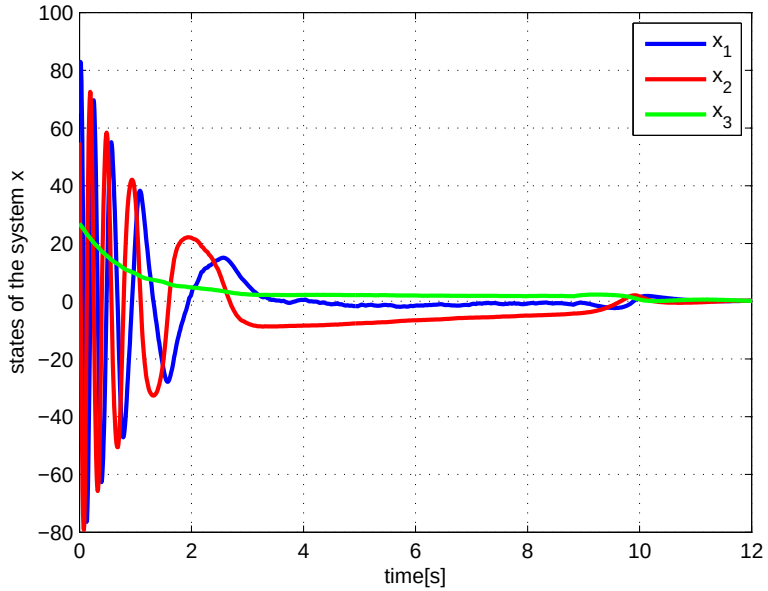
Partially Observable States

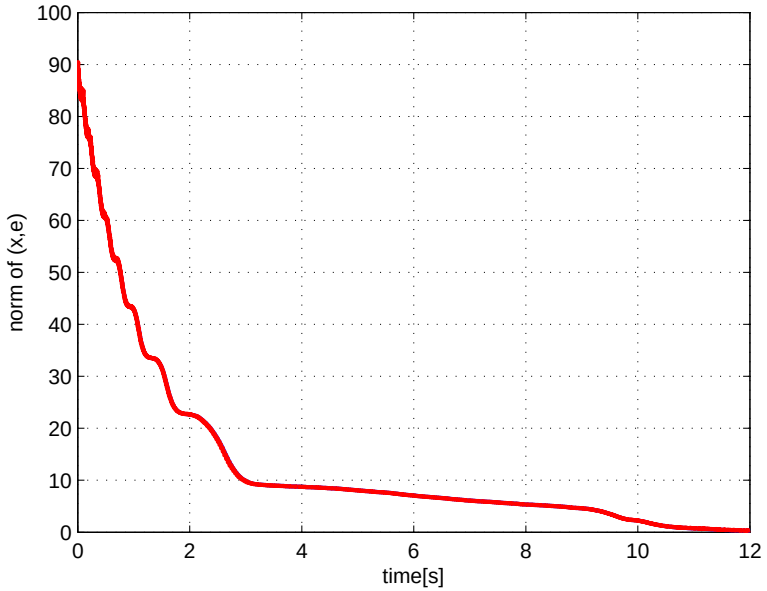
- we assume that the controller can access its state x_c
- consequently, the controller can access u at any given time
- the controller does not have access to the state of the plant x_p but merely to $\hat{\omega}_p$ and $\hat{y}$
- we propose a **particle filter** to extract $\hat{x}_p$ from $\hat{\omega}_p$ and $\hat{y}$ and feeds the controller

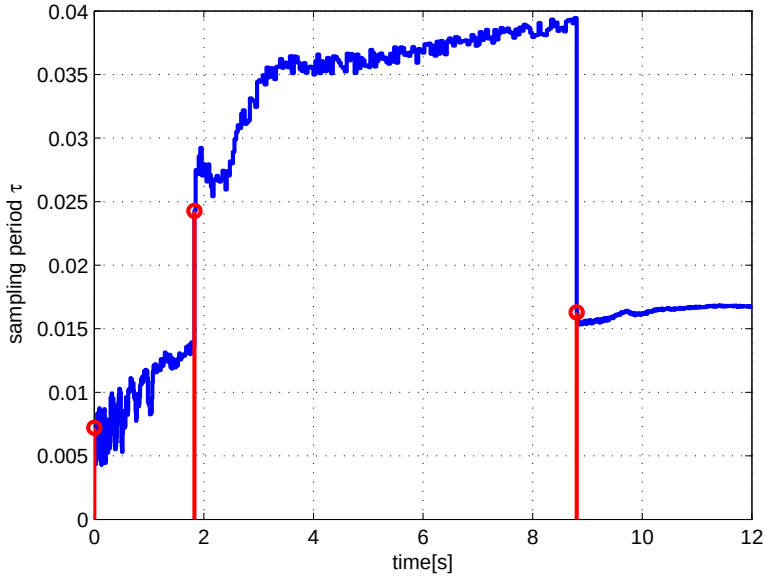
Trajectory Tracking

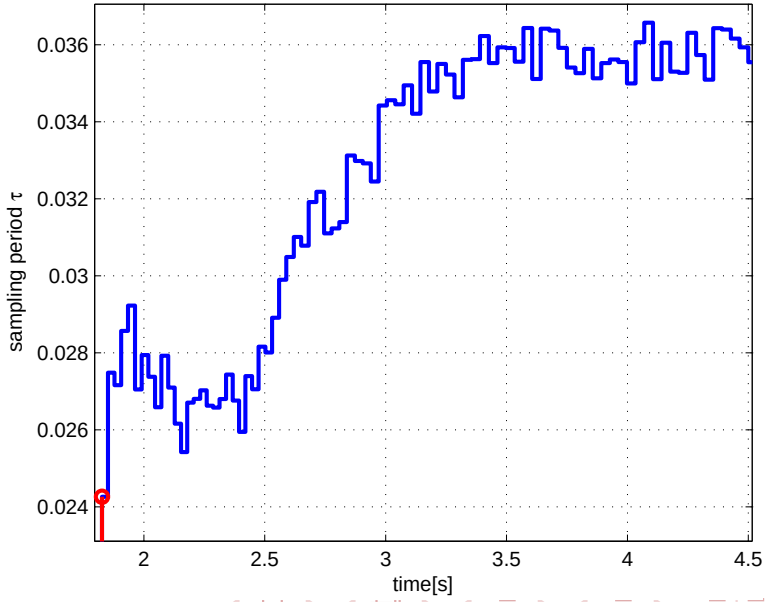


Results - Trajectory Tracking









Outline

1. Introduction
2. Stability
3. Optimal Intermittent Fdbk
4. Summary

Summary

- **input-output triggered control** for **nonlinear systems**
- the **small-gain theorem** is employed to prove stability
- $\mathcal{L}_p$-gains over a **finite horizon**
- novel results regarding $\mathcal{L}_p$-stability of **hybrid systems** are presented
- **optimal** self-triggering via **Approximate Dynamic Programming**

References I

Anta, A. and Tabuada, P. (2010).
To sample or not to sample: Self-triggered control for nonlinear systems.
IEEE Transactions on Automatic Control, 55(9):2030 – 2042.

Estrada, T. and Antsaklis, P. J. (2008).
Stability of model-based networked control systems with intermittent feedback.
In *Proc. of the 17th IFAC World Congress on Automatic Control*, pages 12581–12586.

Lemmon, M. (2010).
Event-triggered Feedback in Control, Estimation, and Optimization, volume 405 of *Lecture Notes in Control and Information Sciences*.
Springer Verlag.

Li, C., Feng, G., and Liao, X. (2007).
Stabilization of nonlinear systems via periodically intermittent control.
IEEE Trans. on Circuits and Systems – II: Express Briefs, 54(11):1019 – 1023.

Nešić, D. and Teel, A. R. (2004).
Input-output stability properties of Networked Control Systems.
IEEE Transactions on Automatic Control, 49(10):1650–1667.

Powell, W. B. (2007).
Approximate Dynamic Programming: Solving the Curses of Dimensionality.
Wiley Series in Probability and Statistics. John Wiley and Sons, Inc., Hoboken, NJ.

Sanfelice, R. (2007).
Robust Hybrid Control Systems.
PhD thesis, University of California Santa Barbara.

References II

Sutton, R. and Barto, A. (1998).
Reinforcement Learning.
The MIT Press, Cambridge, Massachusetts.

Tabbara, M., Nešić, D., and Teel, A. R. (2007).
Stability of wireless and wireline networked control systems.
IEEE Transactions on Automatic Control, 52(9):1615–1630.

Tabuada, P. (2007).
Event-triggered real-time scheduling of stabilizing control tasks.
IEEE Transactions on Automatic Control, 52(9).

Tallapragada, P. and Chopra, N. (2011).
On event triggered trajectory tracking for control affine nonlinear systems.
In *Proceedings of the IEEE Conference on Decision and Control*, pages 5377–5382.

Yu, H. and Antsaklis, P. (2011a).
Event-triggered real-time scheduling for stabilization of passive and output feedback passive systems.
In *Proceedings of the American Control Conference*, pages 1674 – 1679, San Francisco, CA.

Yu, H. and Antsaklis, P. J. (2011b).
Output Synchronization of Multi-Agent Systems with Event-Driven Communication: Communication Delay and Signal Quantization.
Department of Electrical Engineering, University of Notre Dame.
technical report.

Questions? Comments? Suggestions?

Questions? Comments? Suggestions?

Thank You for Your attention!!