Event-triggered Feedback in Control, Estimation, and Optimization

M.D. Lemmon[1]

[1] Department of Electrical Engineering
University of Notre Dame

July 9, 2009 / WIDE Ph.D. Summer School

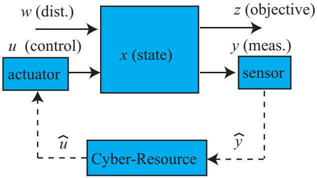
Outline

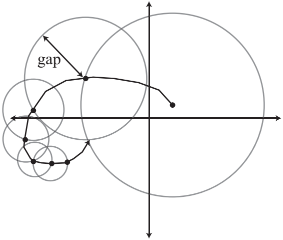
Event-Triggered Feedback for Embedded Control

- Mathematical Preliminaries
- System Model
- ISS and L2 Event-Triggers
- Bounds on Periods and Delays
- Self-triggered implementations

Event-Triggered Sampling

- **Event-Triggered Sampling:** [Arzen 99, Arzen 00]
  - Sensor determines when to sample the system state.
  - The “gap” between current state and past “sampled” state as a measure “novelty” in feedback information
  - Sample the state when the “gap” exceeds a state-dependent threshold
- **Benefits**
  - Reduced usage of computer and communication resources



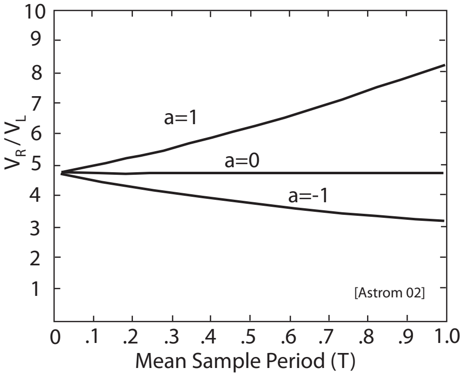


Periodic versus Event-triggered control

Process Model: $dx = axdt + udt + dw$

Variance under Event Sampling = V_L

Variance under Periodic Sampling = V_R



[Astrom 02]

Event-triggering and Computational Resources

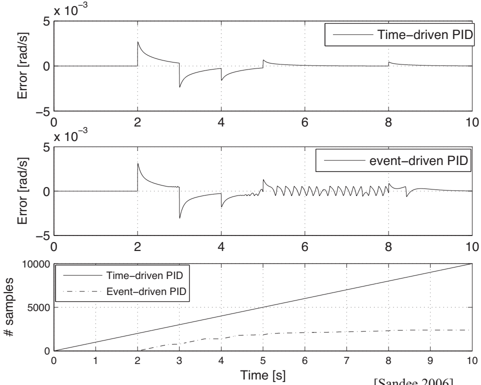
PID controller:

$$\begin{aligned} G(s) &= \frac{A}{(0.33s+1)(0.17s+1)} \\ K(s) &= 30 + \frac{40}{s} + \frac{2s\omega_d}{s+\omega_d} \end{aligned}$$

Event-trigger threshold:

$$|e(t)| = |x(t) - x(r_j)| \leq e_T = \text{threshold}$$

threshold chosen so peak error of both systems are equal



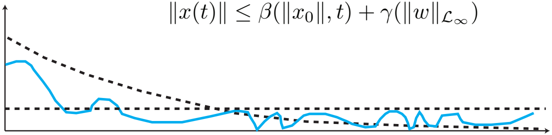
[Sandee 2006]

Input-to-State Stability

- **Process Model:** $\dot{x}(t) = f(x(t), w(t)), \quad x(0) = x_0$
- **Input-to-State Stability (ISS)**

  The system is ISS if there exists $\mathcal{KL}$ function β and class $\mathcal{K}$ function γ such that for any initial condition, $x(0) = x_0$, then the response under any input $w \in \mathcal{L}_\infty$ for all $t \geq 0$ satisfies

$$\|x(t)\| \leq \beta(\|x_0\|, t) + \gamma(\|w\|_{\mathcal{L}_\infty})$$



- **ISS-Lyapunov Function**

  C^1 function $V : \Re^n \rightarrow \Re$ is **ISS-Lyapunov** function if there exist class $\mathcal{K}$ functions $\underline{\alpha}$, $\overline{\alpha}$, γ, and β such that

$$\begin{gathered} \underline{\alpha}(\|x\|) \leq V(x) \leq \overline{\alpha}(\|x\|) \\ \dot{V} \leq -\gamma(\|x\|) + \beta(\|w\|) \end{gathered}$$

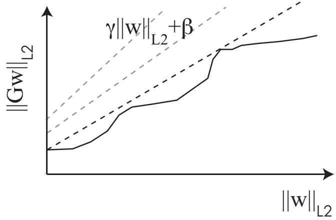
  If V is an ISS-Lyapunov function, then the system is ISS.

$\mathcal{L}_2$ Stability

- **Process Model:** $G : \mathcal{L} \rightarrow \mathcal{L}$
- **L2 Stability**

  The system map G is $\mathcal{L}_2$ stable if for all $w \in \mathcal{L}_2$, $\|Gw\|_{\mathcal{L}_2} \leq \gamma \|w\|_{\mathcal{L}_2} + \beta$
- **Induced L2 Gain**

$$\|G\| = \inf \{ \gamma \in \Re \ : \ \|Gw\|_{\mathcal{L}_2} \leq \gamma \|w\|_{\mathcal{L}_2} + \beta \}$$



Hamilton Jacobi Inequality and $\mathcal{L}_2$ Gain

- **Process Model:** $$\begin{aligned}\dot{x}(t) &= A(x(t)) + B_1(x(t))w(t) + B_2(x(t))u(t)\\ z(t) &= \left[\begin{array}{cc} x(t) & u(t)\end{array}\right]^T\end{aligned}$$
- **Hamilton Jacobi Inequality:**

  Assume there exists γ≥0 and positive definite function V: $\Re^n \rightarrow \Re$ such that

  $$\frac{\partial V}{\partial x}A(x) + \frac{1}{2}\frac{\partial V}{\partial x}\left[\frac{1}{\gamma^2}B_1(x)B_1^T(x) - B_2(x)B_2^T(x)\right]\frac{\partial V}{\partial x}^T + \frac{1}{2}x^Tx \le 0$$
- **L2 Controller**

  If the control ,$u(t)$, is selected so that $u = -B_2^T(x)\frac{\partial V(x)}{\partial x}^T$

  We can then show that

  $$\dot{V} \le -\frac{1}{2}\left(\|u\|_2^2 + \|x\|_2^2 - \gamma^2\|w\|_2^2\right) = -\frac{1}{2}\|z\|_2^2 + \frac{1}{2\gamma^2}\|w\|_2^2$$

  Integrating the above inequality yields

  $$\|z\|_{\mathcal{L}_2} \le \gamma\|w\|_{\mathcal{L}_2} + \sqrt{2\gamma V(x(0))}$$

  which implies the L2 induced gain is less than γ.

Hamilton Jacobi Inequality and $\mathcal{L}_2$ Gain

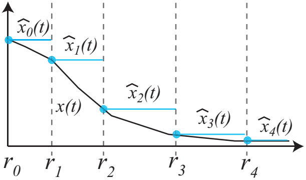
- Consider the directional derivative of the function V that satisfies the earlier Hamilton-Jacobi Bellman inequality
- $$\begin{aligned} \dot{V} &= \frac{\partial V}{\partial x}\left[A(x) + B_1(x)w + B_2(x)u\right] \\ &\leq -\frac{1}{2}\frac{\partial V}{\partial x}\left(\frac{1}{\gamma^2}B_1B_1^T - B_2B_2^T\right)\frac{\partial V}{\partial x}^T - \frac{1}{2}\|x\|_2^2 + \frac{\partial V}{\partial x}B_2u + \frac{\partial V}{\partial x}B_1w && \text{HJI} \\ &= \cancel{\frac{1}{2}\left\|u + B_2^T\frac{\partial V}{\partial x}^T\right\|_2^2} - \frac{1}{2}\|u\|_2^2 - \cancel{\frac{1}{2}\left\|\gamma w - \frac{1}{\gamma}B_1^T\frac{\partial V}{\partial x}^T\right\|_2^2} + \frac{\gamma^2}{2}\|w\|_2^2 - \frac{1}{2}\|x\|_2^2 && \text{Complete Square} \end{aligned}$$

If we let $u = -B_2^T(x)\frac{\partial V(x)}{\partial x}^T$ then we can show

$$\dot{V} \leq -\frac{1}{2}\left(\|u\|_2^2 + \|x\|_2^2 - \gamma^2\|w\|_2^2\right) = -\frac{1}{2}\|z\|_2^2 + \frac{1}{2\gamma^2}\|w\|_2^2$$

which is sufficient to ensure the L2 induced gain is less than γ

State-based Sampled Data System



- Sequence of **"release" times** $\{r_j\}_{j=0}^{\infty}$
- Sequence of **sampled states**

$$\hat{x}_j(t) = x(r_j) \text{ for all } t \in [r_j, r_{j+1})$$

- **Gap** between current state and sampled state

$$e_j(t) = \hat{x}_j(t) - x(t) \text{ for all } t \in [r_j, r_{j+1})$$

- **Process Model** - discretely sampled state feedback

$$\begin{aligned} \dot{x}(t) &= f(x(t), k(\hat{x}_j(t)), w(t)) = f(x(t), k(x(t) + e_j(t)), w(t)) \\ & \quad \text{for all } t \in [r_j, r_{j+1}) \text{ and all } j = 0, \ldots, \infty \end{aligned}$$

- **ISS assumption** - under continuous sampling of state

"Continuously" sampled closed-loop system

$$\dot{x}(t) = f(x(t), k(x(t)), w(t))$$

is input-to-state stable with respect to the input w.

ISS Event Trigger

- ISS assumption implies the existence of an **ISS-Lyapunov function**, V
  There exist class $\mathcal{K}$ functions $\underline{\alpha}$, $\overline{\alpha}$, γ, β_1, and β_2 such that

$$\underline{\alpha}(\|x\|) \leq V(x) \leq \overline{\alpha}(\|x\|)$$
$$\frac{\partial V}{\partial x} f(x, k(x+e), w)) \leq -\gamma(\|x\|) + \beta_1(\|e\|) + \beta_2(\|w\|)$$

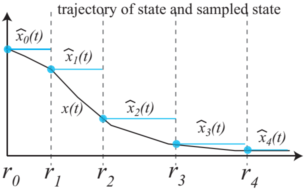
- **ISS Event trigger:** [Tabauda 07]
  If we can guarantee for all $t \geq 0$ and all $j = 0, \ldots, \infty$ that

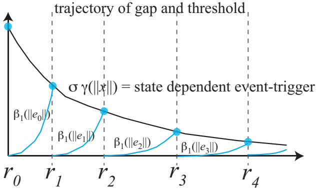
ISS event trigger $\boxed{\beta_1(\|e_j(t)\|) \leq \sigma\gamma(\|x(t)\|)}$

for some $\sigma \in [0, 1]$, then we can ensure that

$$\dot{V} \leq -(1-\sigma)\gamma(\|x(t)\|) + \beta_2(\|w\|)$$

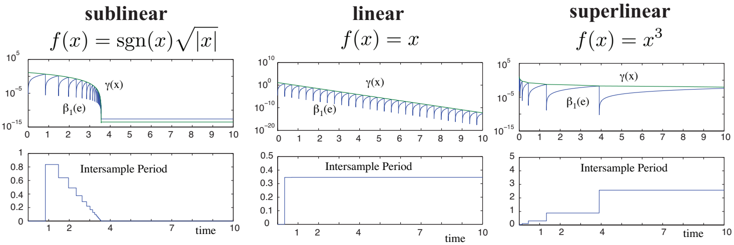
which is sufficient to ensure the sampled system is ISS with respect to w.





Event-triggering Example

- **Process Model:** $$\begin{aligned} \dot{x}(t) &= f(x(t)) + u(t) \\ u(t) &= -2f(\hat{x}_j(t)) \end{aligned}$$
- **Event Trigger:** $\beta_1(\|e_j\|) = e_j^2(t) \geq x^2(t) = \gamma(\|x(t)\|)$



sublinear $f(x) = \text{sgn}(x)\sqrt{|x|}$

linear $f(x) = x$

superlinear $f(x) = x^3$

- **Three cases** in which system dynamic is sublinear, linear, or superlinear
  - sublinear dynamics : exhibit Zeno sampling
  - linear dynamics exhibit periodic sampling
  - linear dynamics exhibit slow sampling around equilibrium point

Lower Bound on Sampling Period

- **Lipschitz continuity of system** $\|f(x, k(x+e))\| \leq L\|x\| + L\|e\|$
- **Lipschitz continuity of event trigger**

$$\beta_1(\|e_j\|) \geq \sigma\gamma(\|x(t)\|) \Rightarrow \gamma^{-1}(\sigma^{-1}\beta_1(\|e_j\|)) \leq \|x(t)\|$$

There exists positive constant P such that

$$\frac{1}{\sigma}\gamma^{-1}(\beta_1(\|e\|)) \leq P\|e\| \leq \|x\| \Rightarrow \frac{\|e\|}{\|x\|} \leq \frac{1}{P}$$

- **Bounds on rate of growth for the "event" quotient** [Tabuada 07]

$$\frac{d}{dt}\frac{\|e\|}{\|x\|} \quad \leq \quad \left(1 + \frac{\|e\|}{\|x\|}\right)\frac{L\|x\| + L\|e\|}{\|x\|} = L\left(1 + \frac{\|e\|}{\|x\|}\right)^2$$

This is a differential inequality that can be used to bound the evolution of the event quotient, $\|e\|/\|x\|$

Lower Bound on Sampling Period

- **Solution of the differential inequality**

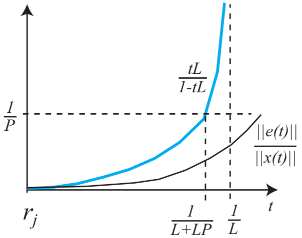
$$\frac{d}{dt}\frac{\|e\|}{\|x\|} \leq L\left(1+\frac{\|e\|}{\|x\|}\right)^2 \implies \frac{\|e(t)\|}{\|x(t)\|} \leq \frac{tL}{1-tL}$$

- **Lower bound on sampling period** [Tabuada 07]

$$\frac{\|e(t)\|}{\|x(t)\|} \leq \frac{tL}{1-tL} \leq \frac{1}{P} \implies r_{j+1}-r_j = T_j \geq \frac{1}{L+LP} > 0$$

- **Non-zeno Behavior**
  This bound is bounded away from zero so sampling interval never equals zero.

  Earlier sublinear example exhibits Zeno behavior because f isn't Lipschitz



$\mathcal{L}_2$ Event Triggers: process model

- **L2 event triggers** are chosen to preserve the induced L2 gain of a previously designed control system [Wang 09]
- **Process Model** - discretely sampled state feedback

$$\begin{aligned}\dot{x}(t) &= A(x) + B_1(x)w + B_2(x)u \\ z(t) &= \begin{bmatrix} x(t) \\ u(t) \end{bmatrix} \\ u(t) &= -B_2^T(\hat{x}_j)\frac{\partial V(\hat{x}_j)}{\partial x}^T = k(\hat{x}_j)\end{aligned}$$

State sampled at release times $\{r_j\}_{j=0}^{\infty}$

$$\hat{x}_j(t) = x(r_j) \text{ for all } t \in [r_j, r_{j+1})$$

Sampling gap

$$e_j(t) = \hat{x}_j(t) - x(t)$$

- **L2 Controller**

  V is a storage function satisfying the HJI for some $\gamma > 0$

$$\frac{\partial V(x)}{\partial x}A(x) - \frac{1}{2}\left\|B_2(x)\frac{\partial V(x)}{\partial x}^T\right\|_2^2 \leq -\frac{1}{2\gamma^2}\left\|B_1(x)\frac{\partial V(x)}{\partial x}^T\right\|_2^2 - \frac{1}{2}\|x\|_2^2$$

  With this V, the continuously-sampled system's has a gain less than γ.

$\mathcal{L}_2$ Event Trigger

- **Assumption:** Lipschitz continuity of L2 controller $\|k(x) - k(\hat{x})\|_2 \leq L\|e\|_2$
- **Storage Function Rate of Change**

$$
\begin{aligned}
& \dot{V} && = \frac{\partial V(x)}{\partial x}\left(A(x) + B_1(x)w - B_2(x)k(\hat{x})\right) \\
\text{(HJI)} \quad & && \leq -\frac{1}{2\gamma^2}\left\| B_1(x)\frac{\partial V(x)}{\partial x}^T \right\|_2^2 - \frac{1}{2}\|x\|_2^2 + \frac{\partial V(x)}{\partial x}B_1(x)w - k^T(x)k(\hat{x}) \\
\text{complete square} \quad & && \leq -\frac{1}{2}\|x\|_2^2 - \frac{1}{2}\left\| \gamma w - \frac{1}{\gamma}B_1^T(x)\frac{\partial V(x)}{\partial x}^T \right\|_2^2 + \frac{\gamma^2}{2}\|w\|_2^2 - k^T(x)k(\hat{x}) \\
\text{Lipschitz} \quad & && \leq -\frac{\beta^2}{2}\|x\|_2^2 + \frac{\gamma^2}{2}\|w\|_2^2 - \boxed{\frac{1-\beta^2}{2}\|x\|_2^2 - \frac{1}{2}\|k(\hat{x})\|_2^2 + \frac{1}{2}L^2\|e\|_2^2}
\end{aligned}
$$

- **Event Trigger** ensures that boxed term is always negative definite which implies the gain is less than γ/β

$$L^2|e\|_2^2 \leq (1-\beta^2)\|x\|_2^2 + \|k(\hat{x})\|_2^2$$

where $(\mathbf{0} < \beta < 1)$

$\mathcal{L}_2$ Event Trigger: example

- **Process Model:**

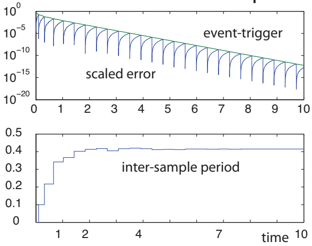
$$\begin{aligned} \dot{x}(t) &= x^3 + u + w \\ u &= -\alpha\hat{x}^3 - \hat{x} \\ w(t) &= (e^{-2t} + \eta)\nu(t) \end{aligned}$$

- **Event Trigger:**

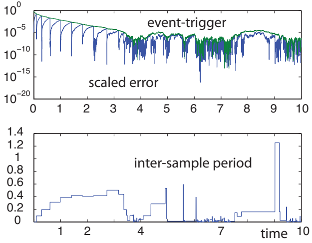
$$L^2\|e\|_2^2 \geq (1-\beta^2)\|x\|_2^2 + \|k(\hat{x})\|_2^2$$

where $L = 1.e$, $\beta = 0.5$, $\gamma = 2$, and $\alpha > \frac{2\gamma^2}{\gamma^2 - 1}$

No Wideband noise: η=0



Wideband noise: η=0.1



- **Example** shows event-triggered systems sensitive to wideband noise

Delays and Periods of $\mathcal{L}_2$ Event Triggers

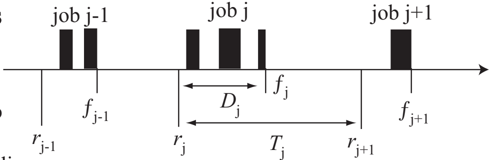
- **Sampling with delays**

  r_j = release time of jth job (sample taken)

  f_j = finishing time of jth job (control applied)

  D_j = delay of jth job < Deadline

  T_j = Period of jth job



- **LTI Process Model:** for all t $\in [f_j, f_{j+1})$

  Noise model: $\|w(t)\| \leq W\|x(t)\|$

$$\boxed{\begin{array}{rcl} \dot{x}(t) & = & Ax + B_1 w + B_2 u \\ z(t) & = & \left[\begin{array}{c} x(t) \\ u(t) \end{array}\right] \\ u(t) & = & -B_2^T P \hat{x}_j = k(\hat{x}_j) \end{array}}$$

- **HJI becomes Riccati Inequality**

  P is symmetric P.D. Matrix satisfying the Riccati inequality for some $\gamma > 0$

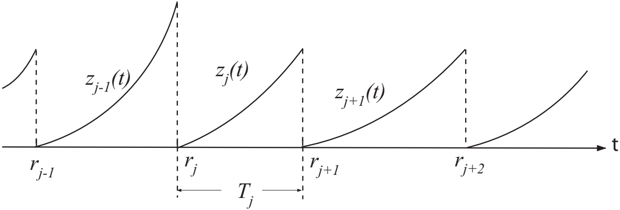
$$A^T P + PA - P\left(B_2 B_2^T - \gamma^{-2} B_1 B_1^T\right) P + I \leq 0$$

- **L2 Event Trigger** $\boxed{e_j^T(t) M e_j(t) \geq \delta x^T(r_j) N x(r_j)}$ [Wang 09]

  where $M = (1-\beta^2)I + PB_2B_2^TP, \quad N = \frac{1}{2}(1-\beta^2)I + PB_2B_2^TP$

Lower Bound on Sampling Period: no delay

- **Normalized gap function** $z_j(t) = \sqrt{M} e_j(t)$
- **L2 Event Trigger:** $\|z_j(t)\| \geq \delta \sqrt{x^T(r_j) N x(r_j)} = \delta \rho(x(r_j))$



- **Sample Period Bound: no delay** [Wang 09]

The normalized gap satisfies $$\frac{d}{dt}\|z_j(t)\|_2 \leq \alpha \|z_j(t)\|_2 + \mu_0(x(r_j))$$

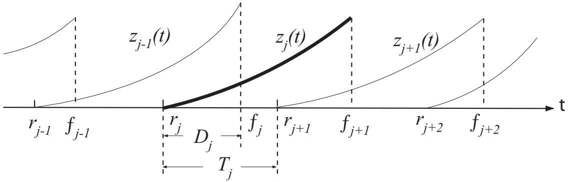
The solution to the differential inequality: $$\|z_j(t)\|_2 \leq \frac{\mu_0(x(r_j))}{\alpha}\left(e^{\alpha(t-r_j)} - 1\right)$$

which bounds the sampling period as $$\boxed{\text{Period} = T_j \geq \frac{1}{\alpha} \ln\left(1 + \alpha \frac{\delta \rho(x(r_j))}{\mu_0(x(r_j))}\right)}$$

Bounds with Nonzero Delay

- With non-zero delay, the evolution of $z_j(t)$ is governed by

$$\begin{array}{rcll} \dot{x}(t) & = & Ax(t) - B_2 B_2^T P x(r_{j-1}) + B_1 w(t) & \text{for } t \in [r_j, f_j) \\ \dot{x}(t) & = & Ax(t) - B_2 B_2^T P x(r_j) + B_1 w(t) & \text{for } t \in [f_j, f_{j+1}) \end{array}$$



- Use differential inequalities to show that if **next release time** is

$$r_{j+1} \quad = \quad f_j + \frac{1}{\alpha} \ln \left(1 + \alpha \frac{\delta \rho(x(r_j)) - \phi(x(r_j), x(r_{j-1}); D_j)}{\mu_0(x(r_j)) + \alpha \phi(x(r_j), x(r_{j-1}); D_j)} \right)$$

and if the delay satisfies the following **deadline**

$$D_j < \frac{1}{\alpha} \ln \left(1 + \alpha \frac{(1-\delta)\rho(x(r_j))}{\alpha \delta \rho(x(r_{j-1})) + \mu_0(x(r_{j-1}))} \right) = \text{Deadline}$$

then the sampled-data system is L2-stable [Wang 09]

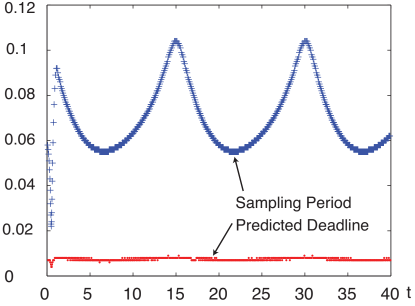
Self-Triggered Feedback

- Self-triggering is a software implementation of event-triggering.
- Use estimates of next release to trigger next control job [Velasco 03, Lemmon 07]
- Use predicted bound on delay as deadline for next control job
- Predicted periods and deadlines serve as task constraints that the real-time scheduler needs to enforce.

Simulation of Self-Triggered System

- Self-triggered inverted-pendulum example [Wang 09]

$$\dot{x}(t) = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & -\frac{mg}{M} & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & \frac{g}{\ell} & 0 \end{bmatrix} x(t) + \begin{bmatrix} 0 \\ \frac{1}{M} \\ 0 \\ -\frac{1}{M\ell} \end{bmatrix} + \begin{bmatrix} 1 \\ 1 \\ 1 \\ 1 \end{bmatrix} w(t)$$



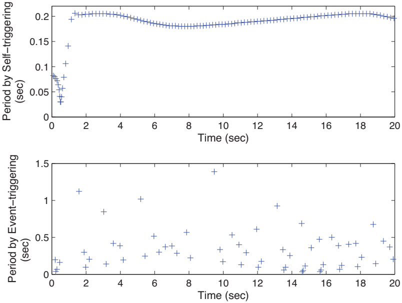
Self-triggered periods tend to exhibit periodic oscillations in sample period

Periodic oscillations in period breakdown in presence of wideband noise

Average Period Comparisons

Self-triggered Period = **0.1782**
Event-triggered Periods = **0.3375** [Wang 09]

Periodically Triggered (MATI) = 0.0092 [Nesic 04]



Event-Triggering in Networked Control Systems

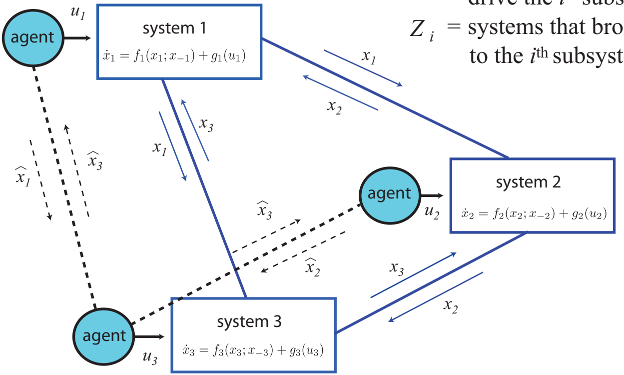
- Model of Networked Control Systems
- ISS Event-Trigger
- Network Artifacts due to Broadcast Protocol
- Impact of Network Artifacts on Event-triggered NCS

Model of Networked Control System

Agent's Dynamics

$$\begin{aligned}
\dot{x}_i(t) &= f_i(x_{D_i}(t), u_i(t), w_i(t)) \\
\hat{x}_i(t) &= x_i(r^i_j) \quad \text{for all } t \in [r^i_j, r^i_{j+1}) \\
u_i(t) &= k_i(\hat{x}_{\overline{Z}_i}(t)) \quad \text{for all } t > 0
\end{aligned}$$

x_i = local state
$\widehat{x}_i$ = broadcast state
r^i_j = broadcast time
D_i = systems that physically drive the i^{th} subsystem
Z_i = systems that broadcast to the i^{th} subsystem



ISS Event-Triggers in NCS

Let $e_i = \hat{x}_i - x_i$ denote the **gap** between agent i's current state and its last broadcast state. Let $V : \Re^{nN} \rightarrow \Re$ be a positive definite function so that there exist class $\mathcal{K}$ functions γ_i, ψ_i, and β_i such that

$$\dot{V} = \sum_{i=1}^{N} L_{f_i} V \leq \sum_{i=1}^{N} (-\gamma_i(\|x_i\|_2) + \psi_i(\|e_i\|_2) + \beta_i(\|w_i\|_2))$$

where

$$L_{f_i} V = \frac{\partial V}{\partial x_i} f_i(x_{D_i}, k(x_{Z_i} + e_{Z_i}), w_i)$$

This means that V is an ISS-Lyapunov function when the gap $e_i = 0$. In other words, k is an ISS controller for the "continuously" sampled networked system.

ISS Event-Triggers in NCS

- Assume for $\rho_i \in (0,1)$, the state and gap trajectories satisfy

  $$-\rho_i \gamma_i(\|x_i(t)\|_2) + \psi_i(\|e_i(t)\|_2) \leq 0$$

  for all $t \in \Re$ and all $i = 1, \ldots, N$. Then clearly

  $$\sum_{i=1}^{N} L_{f_i} V \leq \sum_{i=1}^{N} (-(1-\rho_i)\gamma_i(\|x_i\|_2) + \beta_i(\|w_i\|_2))$$

  which implies the NCS is input-to-state stable.

- To enforce the previous "event-triggering" condition, we require the ith agent **broadcast** its state information to all of its neighbors in U_i whenever the triggering condition

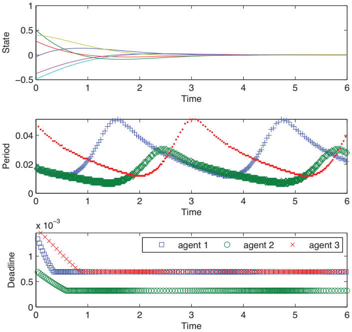
  $$\boxed{\psi_i(\|e_i(t)\|_2) \leq \rho_i \gamma_i(\|x_i(t)\|_2)}$$

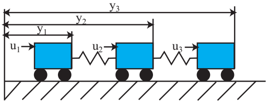
  is violated.

Event-Triggers in NCS: example

- **Process Model** [Wang 09a]

$$\begin{aligned} \dot{x}_i &= \begin{bmatrix} \dot{y}_i \\ u_i + k_i^1 \tanh(y_{i+1} - y_i) + k_i^2 \tanh(y_{i-1} - y_i) + w_i \end{bmatrix} \\ u_i &= K_i \hat{x}_i - k_i^1 \tanh(\hat{y}_{i+1} - \hat{y}_i) - k_i^2 \tanh(\hat{y}_{i-1} - \hat{y}_i) \end{aligned}$$



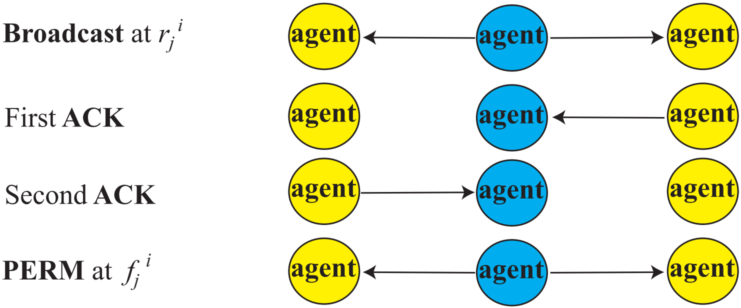


- **Event Trigger**

$$\begin{aligned} 0 &= -0.2\|x_i(b_j^i)\|_2 + 5.9\|e_i^j(t)\|_2 \\ &\quad \text{for the endpoint agents} \\ 0 &= -0.2\|x_i(b_j^i)\|_2 + 10.3\|e_i^j(t)\|_2 \\ &\quad \text{for interior agents} \end{aligned}$$

Broadcast Protocol in Wireless NCS

- Prior analysis assumes that
  - There is no delay between broadcast and reception.
  - All neighbors in Z_i receive and use the broadcast data.
- A **broadcast protocol** must be used to ensure all neighbors receive the broadcast. This protocol will always introduce delay



Broadcast at r_j^i

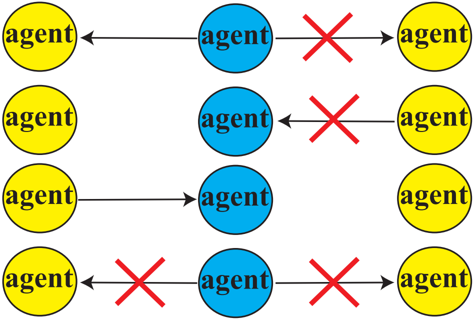
First **ACK**

Second **ACK**

PERM at f_j^i

Dropout Mechanism in Wireless NCS

If broadcast is not “heard” by one agent in Z_i, then Broadcasting agent will not get ACKS from all of its neighbors and therefore will not transmit the PERM message



Broadcast at r_j^i

First **NACK**

Second **ACK**

PERM not sent

If the neighbors don’t receive the PERM message, then they DROP the broadcast state information.

Timing Relations under Broadcast Protocol

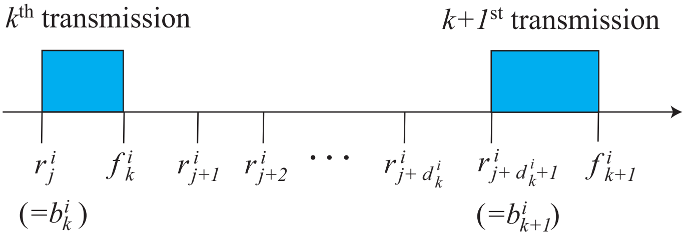
Timing Relations

r_j^i = jth consecutive **broadcast** time

b_k^i = kth **successful broadcast** time

f_k^i = kth **successful finishing** time

d_k^i = number of **dropped** broadcasts between kth and $k+1$st broadcasts

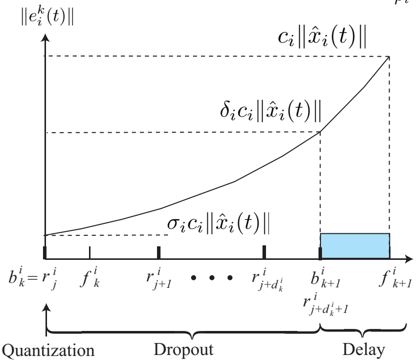


Signal Definitions

$$\begin{aligned}
\hat{x}_i(t) = x_i(b_k^i) &= \text{ith agent's "received" state} \\
e_i(t) = x_i(t) - \hat{x}_i(t) &= \text{gap between received and actual state} \\
\epsilon_i(t) = x_i(t) - x_i(r_j^i) &= \text{gap between transmitted and actual}
\end{aligned}$$

Quantization, Dropout, and Delay Budgets

Event-Trigger for this analysis $\|\epsilon_i(t)\| \leq \frac{\rho_i}{\rho_i + a_i}\|\hat{x}_i(t)\| = c_i\|\hat{x}_i(t)\|$



- 3 Sources for the gap
  - **Quantization** at b^i_k
  - **Dropouts** between b^i_k and b^i_{k+1}
  - **Delay** between b^i_{k+1} and f^i_{k+1}
- Create an error budget for each source

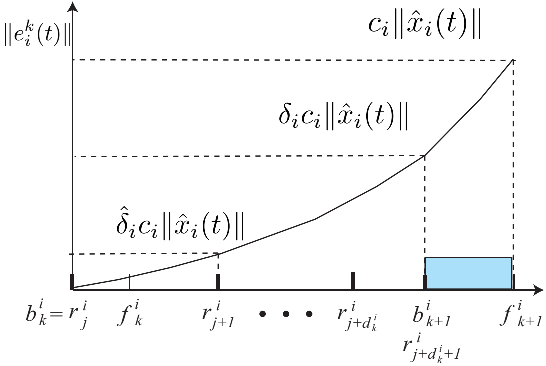
  σ_i = budget for quant. error

  δ_i = budget for quant. and dropout
- Use these budgets to bound the acceptable broadcast quantization error, maximum admissible number of successive dropouts (MANSD), and maximum acceptable broadcast delay (DEADLINE).

Making Event-trigger Robust to Single Dropout

Create Local Event $\left\| x_i(t) - x_i(r_j^i) \right\| = \hat{\delta}_i c_i \| x_i(r_j^i) \|$

Choosing $\hat{\delta}_i \in (0, \delta_i]$, ensures that the broadcast event is triggered before the violation of error budget $\left(\|e_i^k(t)\| \leq \delta_i c_i \|x_i(b_k^i)\| \right)$ allocated for dropouts.



Maximum Admissible Number of Dropouts (MANSD)

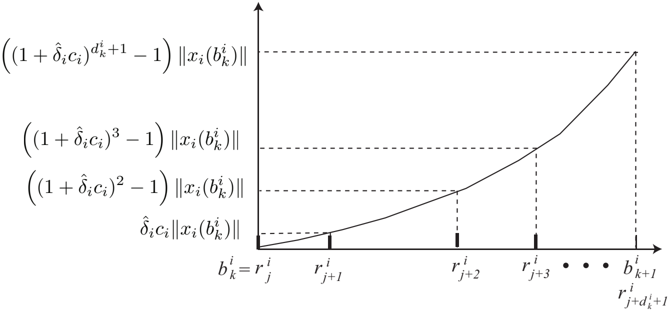
Using the prior event-trigger we can show that the gap satisfies

$$\|e_i^k(t)\| \leq \left((1+\hat{\delta}_i c_i)^{d_k^i+1} - 1\right) \|x_i(b_k^i)\| \leq \delta_i c_i \|x_i(b_k^i)\|$$

We obtain an upper bound on the admissible number of dropouts

$$\boxed{d_k^i \leq \text{MANSD} = \left\lfloor \log_{1+\hat{\delta}_i c_i} (1+\delta_i c_i) \right\rfloor - 1}$$

[Wang 09a]

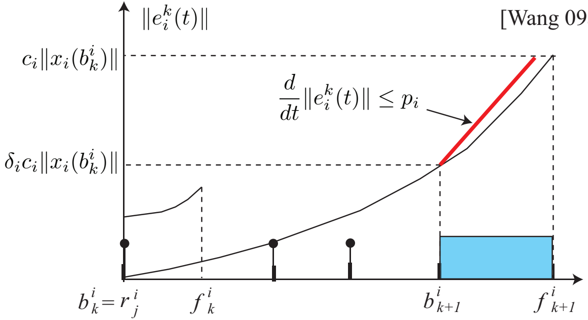


Upper Bound on Admissible Delay

If we can bound the Gap's growth rate $\frac{d}{dt}\|e_i^k(t)\| \leq p_i$

Then to ensure that the total gap is still below the budgeted allowance we require that the "delay" between broadcast and reception satisfy

$$f_{k+1}^i - b_{k+1}^i \leq \max\left\{\frac{(1-\delta_i)c_i}{\rho_i}\|x_i(b_k^i)\|, D_{\min}\right\} = \text{Deadline}$$



[Wang 09a]

Event-Triggered Estimation in Wireless Sensor Networks

- Single Transmission Finite Horizon Problem
- Transmission Decision and Optimal Stopping
- Dynamic Programming
- Optimal Event-trigger

Canonical Finite-Horizon Problem

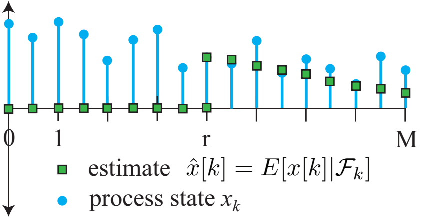
Consider a “simple” canonical problem: [Rabi 06, Rabi 08]

- Finite Horizon, $[0,M]$
- Discrete-time Scalar LTI System

  $$x[k] = ax[k-1] + w[k]$$

  $w[k]$ = zero mean white Gaussian noise with covariance q

  x_0 = initial state is normal random variable $N(\mu_0,\pi_0)$



🟩 estimate $\hat{x}[k] = E[x[k]|\mathcal{F}_k]$

● process state x_k

- Sensor transmits at time instant, $r \in [0,M]$
- Least Square Estimate

$$\hat{x}[k] = E[x[k]|\mathcal{F}_k] = \begin{cases} \mu_0 a^k & k = 0,\dots,r-1 \\ x[r]a^{k-r} & k = r,\dots,M \end{cases}$$

Transmission as Optimal Stopping

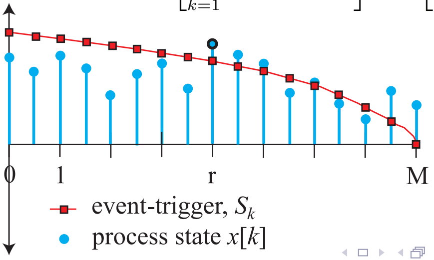
- The transmission time r is a random variable that is a **stopping time** of the random process $\{x[k]\}$. In particular this means

$$r = \min\{k : x[k] \notin S_k\}$$

were S_k is a set of **event sets**

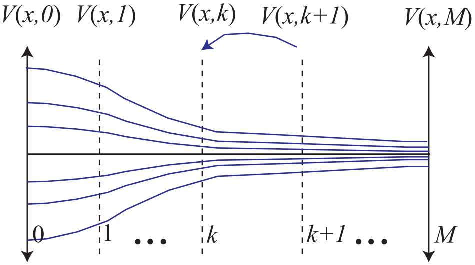
- The problem is to determine these event sets such that the estimator's means square error is minimized

$$J(S_1, \cdots, S_M) = E\left[\sum_{k=1}^{M}(x[k] - \hat{x}[k])^2\right] = E\left[\sum_{k=1}^{M}\tilde{x}[k]^2\right]$$



Dynamic Programming

- This problem may viewed as an optimal “control” problem in which the stoppng events $\{S_k\}$ are the “controls”. We can therefore use dynamic programming to solve the problem.
- Value Function, $V(x,n)$, is the optimal value obtained if the process starts at state x at time $n < M$.
- **Bellman’s Principle of Optimality** says that if $\{u^*[n]\}_{n=k}^{M}$ is an optimal control generating state trajectory $\{x^*[n]\}_{n=k}^{M}$ from initial state $(x^*[k], k)$ to M, then $\{u^*[n]\}_{n=k+1}^{M}$ is the optimal control from initial state $(x^*[k+1], k+1)$ to M.



contours of constant value

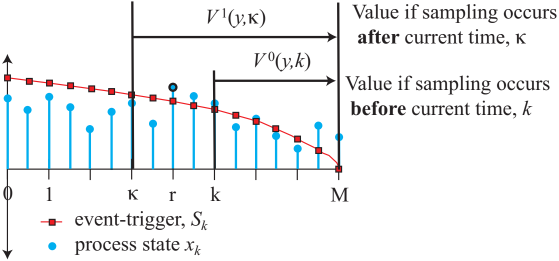
Problem's Value Functions

$$V^1(y,k) = \min_{S_k,\cdots,S_M} E_{r,\tilde{X}_k^M}\left[\sum_{j=k}^{M} \tilde{x}^2[j] \,|\, \tilde{x}^-[k] = y, k \leq r \leq M\right]$$

$$=$$ minimum cost from k to M when transmission occurs **after** $k-1$ and a priori est. error is y.

$$V^0(y,k) = \min_{S_k,\cdots,S_M} E_{r,\tilde{X}_k^M}\left[\sum_{j=k}^{M} \tilde{x}^2[j] \,|\, \tilde{x}^-[k] = y, k > r\right]$$

$$=$$ minimum cost from k to M when transmission occurs **before** k and a priori estimation error is y



Optimal Event-Triggered Transmission Thresholds

- The minimum cost is achieved by $J^* = E\left[V^1(\tilde{x}_0, 0)\right]$. We can use Bellman's principle to compute this in a backward recursion.

$$V^1(y,k) = \min\left\{F_k(y), G_k(q)\right\}$$

where

$$\begin{aligned}
G_k(q) &= E_{\tilde{x}[k+1]}\left[V^0(k+1, \tilde{x}^-[k+1]) \,|\, \tilde{x}[k] = 0\right] \\
&= \text{Cost if transmission occurs at time } k \\
F_k(y) &= y^2 + E_{\tilde{x}[k+1]}\left[V^1(k+1, \tilde{x}^-[k+1]) \,|\, \tilde{x}[k] = y\right] \\
&= y^2 + G_{k+1}(q) - \int_{S_{k+1}} \left[G_{k+1}(q) - F_{r+1}(\tilde{x}[r+1])\right] dP_{\tilde{x}[k+1]} \\
&= \text{Cost if transmission occurs after time } k \text{ given error is } y
\end{aligned}$$

- The optimal control (stopping event) is easily obtained from the choice implied by the backward recursion

$$u^*[k] = \begin{cases} 0 \text{ don't sample} & F_k(\tilde{x}[k]) < G_k(q) \\ \quad 1 \text{ sample} & F_k(\tilde{x}[k]) \geq G_k(q) \end{cases}$$

This is an event-triggered threshold logic.

Suboptimal Transmission Thresholds

- The threshold function F_k can be computed numerically off-line.
- We can also adopt a sub-optimal approach by noting that

$$\begin{aligned} F_k(y) &= y^2 + G_{k+1}(q) - \int_{S_{k+1}} \left[G_{k+1}(q) - F_{k=1}(\tilde{x}[k+1]) \right] dP_{\tilde{x}[k+1]} \\ &\leq y^2 + G_{k+1}(q) \end{aligned}$$

- which leads to a stopping event that is only dependent on the process noise covariance q.

$$S_k^+ = \left[-\sqrt{\frac{1-a^{2(M-k)}}{1-a^2}q}, \sqrt{\frac{1-a^{2(M-k)}}{1-a^2}q} \right]$$

which is an inverse quadratic threshold that goes to zero as k goes to M.

Simulation Results

- Simulation of optimal, suboptimal, and periodically sampled schemes where $M = 10$, $q = 1$, and $w_k \in N(0, 1)$.

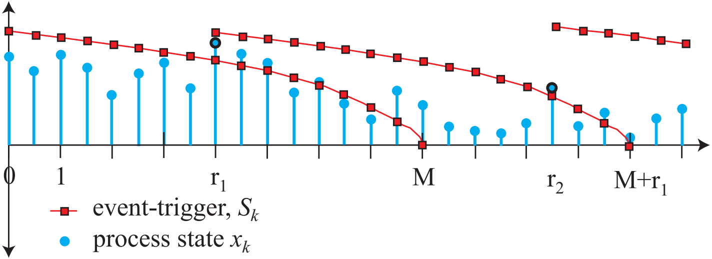
$a = 0$ (stable)	periodic	suboptimal	optimal
J (cost)	9.98	8.50	7.68
r (sampling time)	5	3.13	7.1

$a = 1$ (unstable)	periodic	suboptimal	optimal
J (cost)	30.15	21.70	21.68
r (sampling time)	5	7.01	7.56

- Simulation results show that
  - $J^* < J^+ < J_p$, so that optimal policy is indeed optimal
  - $J^+ \approx J^*$ when process is unstable (forgets initial condition quickly)
  - optimal policy always delays sampling longer than periodic policy
    - but this is not necessarily true for suboptimal sampling policy.

Moving Beyond Finite-Horizon Problem

- Finite-horizon problem can be used as the basis of a receding-horizon estimation scheme



- Trigger broadcast based on thresholds for finite-horizon problem. Treat broadcast as starting point for next finite-horizon problem

Event-triggered Distributed Optimization

- Network Utility Maximization Problem
- Dual-Decomposition Algorithm
- Event-triggered Augmented Lagrangian Algorithm
- Scalability Results

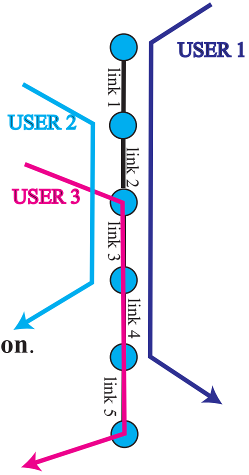
Network Utility Maximization (NUM) Problem

- Maximize utilities of N users transmitting over M shared communication links.

$$\begin{array}{ll} \text{maximize:} & U(x) = \sum_{i=1}^{N} U_i(x_i) \\ \text{subject to:} & Ax \leq c, \quad x \geq 0 \end{array}$$

where A is an incidence matrix mapping network nodes to network links, c is a vector of limits on link throughput, and x is the vector of user transmission rates.

- $U_i(x_i) = \log x_i$ is the i^{th} user's **utility function**.
- The Network Utility Maximization (NUM) problem is found in numerous applications that optimize overall networked system performance subject to a shared resources.



[Kelly 98]

Dual Version of NUM Problem

- Recast as a dual min-max problem

  minimize: $\max_{x \geq 0} \left(\sum_{i=1}^{N} U_i(x_i) - p^T(Ax - c) \right)$

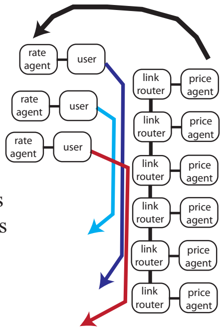
  subject to: $p \geq 0$

  where the vector p is a set of **shadow prices** that each link charges its users.

- Solve this problem in an “alternating” manner where
  - each “link” selects a price based on observed user rates
  - each user selects its rate based on the transmitted prices

- An algorithm known as “dual-decomposition” allows us to implement this “alternating” recursion in a highly distributed manner [Low 99]

prices transmitted back over communication network



user rates observed by each link router

Dual-Decomposition Algorithm

- **Dual-decomposition** [Low 99] is a “distributed” algorithm commonly used to solve the dual form of the NUM problem

$$\begin{aligned} x_i[k+1] &= \arg\max_{x \geq 0} \left(U_i(x_i[k]) - x_i[k] \sum_{j \in L_i} p_j[k] \right) = \left(\sum_{j \in L_i} p_j[k] \right)^{-1} \\ p_j[k+1] &= \max \left\{ 0, p_j[k] + \gamma \left(\sum_{i \in S_j} x_i[k] - c_j \right) \right\} \end{aligned}$$

where $U_i(x_i) = \log(x_i)$ where is the user’s utility function

- γ is a step size chosen to ensure convergence [Low 99]

$$0 < \gamma < \frac{-2\max_{(i,x_i)} \nabla^2 U_i(x_i)}{\bar{L}\,\bar{S}}$$

where $\bar{L}$ is maximum number of links that any route uses and $\bar{S}$ is the maximum number of users on any link.

- $\bar{L}$ and $\bar{S}$ are measures of network complexity. So the number of messages passed by dual-decomposition increases with longer routes and neighborhoods.

Augmented Lagrangian Algorithm

- Event-triggered NUM algorithm is best implemented using an “interior-point” algorithm based on an Augmented Lagrangian.
- **Augmented Lagrangian** of the NUM problem

$$L(x;\lambda,w) = -\sum_{I=1}^{N} U_i(x_i) + \sum_{j=1}^{M} \psi_j(x;\lambda,w)$$

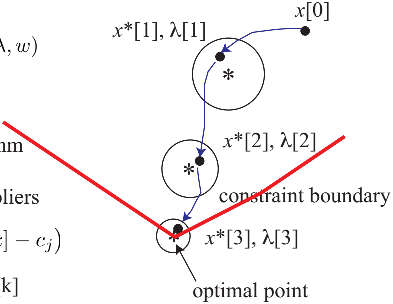
where $\psi_j(x;\lambda,w)$ is a constraint penalty

- Primal **Augmented-Lagrangian** algorithm
  - Sequence $\{\lambda_j[k]\}$ of Lagrange Multipliers

$$\lambda_j[k+1] \quad = \quad \lambda_j[k] + \frac{1}{w_j}\left(a_j^T x[k] - c_j\right)$$

  - Compute approximate minimizer $x^*[k]$

$$x^*[k+1] = \arg\max L(x;\lambda[k],w)$$



[Wan 09, Wan 09a]

Distributed Augmented Lagrangian Algorithm - continuous access

- Users compute the approximate minimizer using a gradient following algorithm.

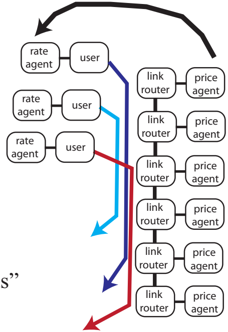
$$x_i(t) \quad = \quad \int_0^t \left(\frac{\partial U_i(x_i(\tau))}{\partial x_i} - \sum_{j=1}^{M} \mu_j(\tau) A_{ji} \right)^+ d\tau$$

where the links update the variable

$$\mu_j(t) = \left(\lambda_j + \frac{1}{w_j} \left(\sum_{i=1}^{N} A_{ij} x_i(t) - c_j \right) \right)^+$$

- Note that these two equations define a feedback loop between users and links in the players have “continuous” access to each others’ states.
- In practice, link updates are carried over packet switched network, so that users only have “discrete” access to link states.

link states transmitted over communication network



user rates observed by each link router

Broadcasting Link States

- With discrete access to "link" states, let $\{T_j^L[\ell]\}_{\ell=0}^{\infty}$ be the time instants when link j transmits its link state to its users. The link state received by the users is

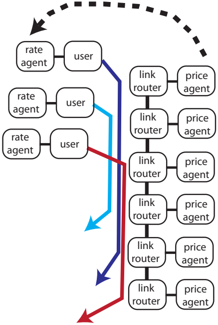
$$\hat{\mu}_j(t) = \mu_j(T_j^L[\ell])$$

- With this sampled link state, the user's gradient algorithm becomes

$$x_i(t) = \int_0^t \left(\frac{\partial U_i(x_i(\tau))}{\partial x_i} - \sum_{j=1}^{M} \hat{\mu}_j(\tau) A_{ji} \right)^+ d\tau$$

- If we treat $L(x;\lambda,w)$ as a candidate Lyapunov function then we need to examine $\dot{L}(x;\lambda,w)$

prices transmitted at discrete instants over comm. network



user rates observed by each link router

Convergence Analysis

- Let z_i be the *i*th user state's rate of change

$$z_i(t) = \dot{x}_i(t) = \left(\frac{\partial U_i(x_i(t))}{\partial x_i} - \sum_{j=1}^{M} \hat{\mu}_j(t) A_{ji} \right)^{+}$$

- Let $L(x;\lambda,w)$ be a candidate Lyapunov function

$$\begin{aligned}
\dot{L}(x;\lambda,w) &= \sum_{i=1}^{N} \frac{\partial L}{\partial x_i}\frac{dx_i}{dt} = -\sum_{i=1}^{N} z_i \left(\frac{\partial U_i(x_i(t))}{\partial x_i} - \sum_{j=1}^{M} \mu_j A_{ji} \right) \\
&\le -\sum_{i=1}^{N} \left[\frac{1}{2} z_i^2 - \frac{1}{2} \left(\sum_{j=1}^{M} (\mu_j - \hat{\mu}_j) A_{ji} \right)^2 \right] \\
&\le -\frac{1}{2} \sum_{i=1}^{N} z_i^2 + \frac{1}{2} \sum_{j=1}^{M} \overline{L}\,\overline{S} (\mu_j - \hat{\mu}_j)^2
\end{aligned}$$

- We can't use this to directly set up an event trigger for the link.

Convergence Analysis and Event Triggers

- We need "local" conditions that can ensure $\dot{L}(x;\lambda,w)$ is negative definite. So introduce a sequence $\{T_i^S[\ell]\}_{\ell=0}^{\infty}$ of time instants when the ith user transmits its modified state, z_i, to its links. The transmitted user state is

$$\hat{z}_i(t) = z_i(T_i^S[\ell])$$

- We can now rewrite our bound on $\dot{L}$ as

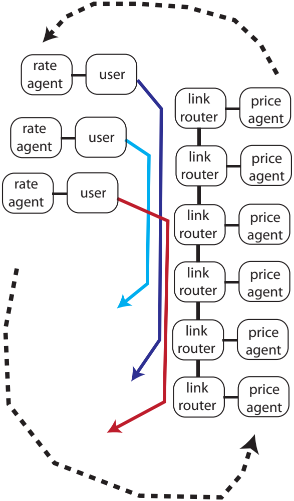
$$\dot{L}(x;\lambda,w) \quad \leq \quad -\frac{1}{2}\sum_{i=1}^{N}\boxed{\left[z_i^2 - \rho\hat{z}_i^2\right]} - \frac{1}{2}\sum_{j=1}^{M}\left[\boxed{\rho\sum_{i\in S_j}\frac{1}{\overline{L}}\hat{z}_i^2 - \overline{L}\,\overline{S}(\mu_j - \hat{\mu}_j)^2}\right]$$

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

Event Trigger for User

Event Trigger for Link

Event-Triggered Distributed Optimization



- jth link broadcast its state, μ_j, at times $\{T_j^L[\ell]\}_{\ell=0}^{\infty}$ when

$$\overline{L}\,\overline{S}(\mu_j(t)-\hat{\mu}_j(t))^2 \geq \rho \sum_{i\in S_j} \frac{1}{\overline{L}} \hat{z}_i^2(t)$$

- jth link can continuously monitor its local state

$$\mu_j(t) = \left(\lambda_j + \frac{1}{w_j}\left(\sum_{i\in S_j} x_i(t) - c_j\right)\right)^+$$

- ith user can continuously monitor its local modified state

$$z_i(t) = \left(\frac{\partial U_i(x_i(t))}{\partial x_i} - \sum_{j\in L_i} \hat{\mu}_j(t)\right)^+$$

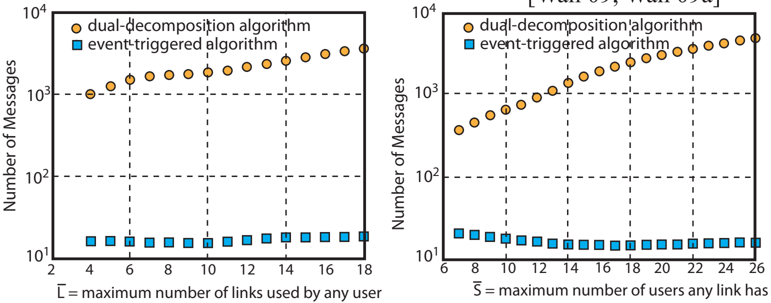
- ith user broadcasts its modified state, z_i, at times $\{T_i^S[\ell]\}_{\ell=0}^{\infty}$ when

$$z_i^2(t) - \rho\hat{z}_i^2(t) \leq 0$$

Simulation Results and Scaling

- Simulated on Randomly Generated Network - 150 users
- Event-triggering reduced message complexity by two orders of magnitude
- Event-triggering message complexity was nearly scale-free

[Wan 09, Wan 09a]



Research Issues

- Real-life applications and testbeds
- Fault Tolerance and Resiliency
- Safety-critical Applications
- Extending Single Sample Finite Horizon Estimation Problem
- Moving from State-Feedback to Output Feedback
- Observability, controllability, and certainty equivalence of event-triggered systems
- Event-triggers based on stochastic stability concepts
- Event-triggered consensus filtering and flocking

References

- [Anta 08] A.Anta and P. Tabuada, Self-triggered stabilization of homogeneous control systms< American Control Conference, 2008
- [Arzen 99] K.E. Arzen, A simple event-based PID controller, IFAC Congress, 1999.
- [Arzen 00] K.E. Arzen, A. Cervin, J. Eker, and L. Sha, An introduction to control and scheduling co-design, CDC, 2000.
- [Astrom 02] K.J. Astrom and B.M. Bernhardsson, Comparison of Riemann and Lebesgue Sampling for first order stochastic systems, CDC, 2002.
- [Bamieh 03], B. Bamieh, Intersample and finite wordlength effects in sampled-data problems, IEEE-TAC, Vol 48(4):639-643 2003.
- [Carnevale 07] D. Carnevale, A. Teel and D. Nesic A Lyapunov proof of improved maximum allowable transfer interval for networked control systems, IEEE-TAC, Vol 52:892-897, 2007
- [Cervin 08] A. Cervin and T. Henningsson, Scheduling of Event-triggered Controllers on a Shared Network, CDC 2008.
- [Cogill 07] R. Cogill, S. Lall, and J.P. Hespanha, A constant factor approximation algorithm for event-based sampling, ACC 2007.
- [Heemels 09] W. Heemels, A.Teel, N. Van de Wouw, and D. Nesic, Networked Control Systems with Communication Constraints: tradeoffs between sampling intervals, delays and performance, ECC 2009.
- [Heemels 08] W.P.M.H. Heemels, J.H. Sandee, P.P.J. van den Bosch, Analysis of event-driven controllers for linear systems, Int. J. of Control, 81(4), pp. 571-590 (2008)
- [Heemels 99] W.P.M.H. Heemels, R.J.A. Gorter, A. van Zijl, P.P.J. v.d. Bosch, S. Weiland, W.H.A. Hendrix, M.R. Vonder, Asynchronous measurement and control: a case study on motor synchronisation, Control Engineering Practice, 7(12), 1467-1482, (1999).
- [Kelly 98] F.P. Kelly, A.K. Maulloo and DKH Tan, Rate control for communicatio networks: shadow prices, proportional fairness and stability, Journal of the Operation Research Society, Vol 49(3):237-252, 1998.
- [Lemmon 07] M. Lemmon, T. Chantem, X. Hu, and M. Zyskowski, On self-triggered full information H-infinity controllers, HSCC 2007.
- S.H. Low and D.E. Lapsley, Optimization flow control I: basic algorithm and convergence, IEEE/ACM Transactions on Networking, Vol 7(6):861-874, 1999.
- [Mazo 08] M. Mazo and P. Tabuada, On event triggered and self-triggered control over sensor actuator networks, CDC 2008.
- [Nesic 04] D. Nesic and A.R. Teel, Input-output stability properties of networked control systems, IEEE-TAC, Vol 49(10):1650-1667, 2004.
- [Rabi 08] M. Rabi, G. Moutstakides, J.S. Baras, Adaptive sampling for linear state estimation, submitted to SIAM journal on Control and Optimziation, December 2008.
- [Rabi 06] M. Rabi, Packet based inference and control, Ph.D. thesis, University of Maryland, 2006.
- [Sandee] J.H. Sandee, Event-driven contorl in in theory and practice: tradeoffs in software and control performance, Ph.D. Thesis, Technische Universeteit Endhoven, 2006.
- [Seto 96], D. Seto, J. Lehoczky, L. Sha and K.G. Shin, On task schedulability in real-time control systems, RTAS, 1996.
- [Tabuada 07] P. Tabuada, Event-triggered real-time scheduling of stabilizing control tasks IEEE-TAC, Vol 52(9):1680-1684, 2007
- [Velasco 03] M. Velasco, P. Marti and J.M. Fuertes, The self-triggered task model for real-time control ssytems, WIP track, RTSS, 2003.
- [Van der Schaft 00], A.J. Van der Schaft, L2-gain and passitivity techniques in nonlinear control, Springer, 2000.
- [Walsh 2002] G. Walsh, H. Ye and L. Bushnell, Stability analysis of networked control systems, IEEE Transactions on Control Systems Technolgy, Vol 10(3):438-445, 2002.
- [Wan 09] P. Wan and M.D. Lemmon, Distributed network utility maximization using event-triggered barrier methods, ECC 2009.
- [Wan 09a] P. Wan and M.D. Lemmon, Event triggered distributed optimization in sensor networks, Information Processing in Sensor Networks (IPSN), 2009.
- [Wang 08] X. Wang and M.D. Lemmon, Event triggered broadcasting across distributed networked control systems, ACC 2008.
- [Wang 09] X. Wang and M.D. Lemmon, Self triggered feedback control systems with finite gain L2 stability, IEEE-TAC, Vol 54(3):452-467, 2009.
- [Wang 09a] X. Wang and M.D. Lemmon, Event triggering in distributed networked systems with data dropouts and delays, Hybrid Systems: computation and control (HSCC), 2009.
- [Xu 04] Y. Xu and J.P. Hespanha, Optimal communication logics in networked control systems, CDC 2004.
- [Zhang 01] W. Zhang, M.S. Branicky, and S.M. Phillips, Stability of Networked Control Systems, IEEE Control Systems Magazine, Vol 21(1):84-99, 2001. `