Friday Seminar: Event-Based Estimation with Stochastic Triggering

Marcus T. Andrén

Dept. of Automatic Control
Lund Institute of Technology

Outline

Outline of this seminar

- The Remote Estimation Problem
- Current Research: Stochastic Triggering
- Further Work & Summary

Why Event-Based Estimation?

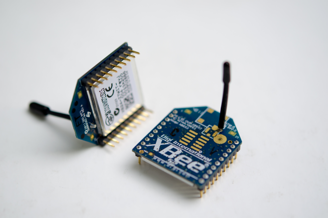
Event-Based Estimation: when sensing and transmitting measurements has a **cost**.

Networked Control Systems

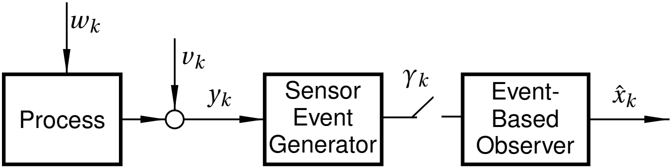
- Increasing use of wireless networks in control
- Shared network bandwidth, energy consumption and computations should be efficiently used

Smart Sensors

- Wireless, programmable transceiver modules
- Opportunity for simple event-generation



The Remote Estimation Problem



$$x_{k+1} = Ax_k + w_k, \quad y_k = Cx_k + v_k$$

$$w_k \sim N(0, Q), \quad v_k \sim N(0, R)$$

$$\gamma_k = \begin{cases} 0 \implies \text{No transmission} \\ 1 \implies \text{Transmission} \end{cases}$$

Goal:

Design a triggering condition, derive the corresponding MMSE-estimator

Estimation with No Measurement?

Simplest possible:

Intermittent Kalman Filter [Sinopoli et.al 2004]

Time Update:

$$\hat{x}_{k+1|k} = A\hat{x}_{k|k}$$
$$\hat{y}_{k+1|k} = C\hat{x}_{k+1|k}$$
$$P_{k+1|k} = AP_{k|k}A^T + Q$$

Measurement Update:

$$\hat{x}_{k+1|k+1} = \hat{x}_{k+1|k} + \gamma_k K_{k+1}[y_{k+1} - \hat{y}_{k+1|k}]$$
$$P_{k+1|k+1} = [I - \gamma_k K_{k+1} C] P_{k+1|k}$$

- $\gamma_k = 0 \implies$ just propagate prediction
- Simple, but disregards information from triggering condition!
- However, for e.g random packet drops, intermittent Kalman is optimal

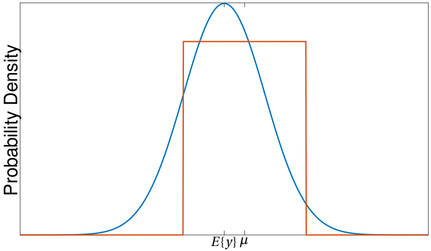
Many event-generators based on the **Send-on-Delta** rule:

Send-on-Delta

$$\gamma_k = \begin{cases} 1, & \text{if } |y_k - \mu_k| \geq \Delta \\ 0, & \text{else} \end{cases}$$

- Sensor compares collected measurement y_k to prediction μ_k. Transmits if difference is larger than Δ
- Commonly, $\mu_k = y_{last}$
- By varying Δ, different average communication rates can be achieved

Probability Density using SoD

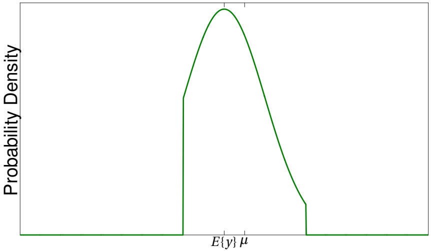


- $\gamma_k = 0 \implies$ no longer Gaussian!

Solution?

- **Particle Filter** - Good performance, but heavy online computations, approximate, and difficult to analyze.
- **Approximate as Gaussian** - Simpler analysis, but approximation might be poor.
- **Other Triggering Conditions** - E.g Stochastic Triggering

Probability Density using SoD

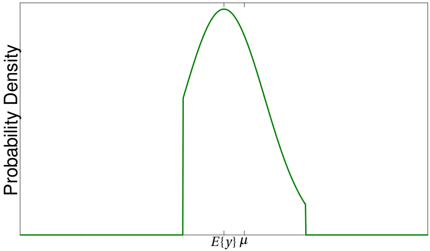


- $\gamma_k = 0 \implies$ no longer Gaussian!

Solution?

- **Particle Filter** - Good performance, but heavy online computations, approximate, and difficult to analyze.
- **Approximate as Gaussian** - Simpler analysis, but approximation might be poor.
- **Other Triggering Conditions** - E.g Stochastic Triggering

Probability Density using SoD



- $\gamma_k = 0 \implies$ no longer Gaussian!

Solution?

- **Particle Filter** - Good performance, but heavy online computations, approximate, and difficult to analyze.
- **Approximate as Gaussian** - Simpler analysis, but approximation might be poor.
- **Other Triggering Conditions** - E.g Stochastic Triggering

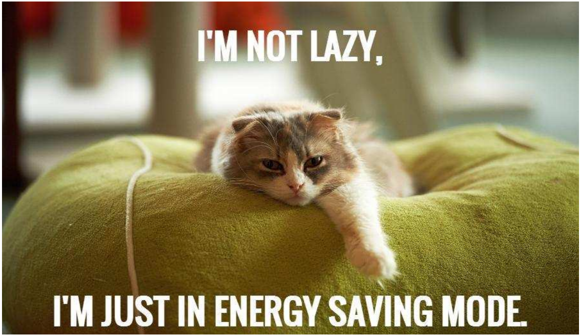
Outline

Outline of this seminar

- *The Remote Estimation Problem*
- *Current Research: Stochastic Triggering*
- *Further Work & Summary*

Stochastic Triggering (aka "Lazy Sampling")

Idea: A "lazy" sensor which chooses to transmit according to certain probability



Stochastic Triggering

Consider another triggering rule:

Stochastic Triggering:

$$\xi_k \sim U(0,1), \quad \Phi(y_k - \mu_k) \in [0,1]$$

$$\gamma_k = \begin{cases} 0 \text{ if } \xi_k \leq \Phi(y_k - \mu_k) \\ 1 \text{ else} \end{cases}$$

$$\implies \Pr(\gamma_k = 0) = \Phi(y_k - \mu_k)$$

Nice properties when Φ is a scaled Gaussian:

Decision function:

$$\Phi(y_k - \mu_k) = \exp[-\tfrac{1}{2}(y_k - \mu_k)^T Y (y_k - \mu_k)]$$

Stochastic Triggering

Consider another triggering rule:

Stochastic Triggering:

$$\xi_k \sim U(0,1), \quad \Phi(y_k - \mu_k) \in [0,1]$$

$$\gamma_k = \begin{cases} 0 \text{ if } \xi_k \leq \Phi(y_k - \mu_k) \\ 1 \text{ else} \end{cases}$$

$$\implies \Pr(\gamma_k = 0) = \Phi(y_k - \mu_k)$$

Nice properties when Φ is a scaled Gaussian:

Decision function:

$$\Phi(y_k - \mu_k) = \exp[-\frac{1}{2}(y_k - \mu_k)^T Y (y_k - \mu_k)]$$

Consider another triggering rule:

Stochastic Triggering:

$$\xi_k \sim U(0,1), \quad \Phi(y_k - \mu_k) \in [0,1]$$

$$\gamma_k = \begin{cases} 0 \text{ if } \xi_k \leq \Phi(y_k - \mu_k) \\ 1 \text{ else} \end{cases}$$

$$\implies \Pr(\gamma_k = 0) = \Phi(y_k - \mu_k)$$

Nice properties when Φ is a scaled Gaussian:

Decision function:

$$\Phi(y_k - \mu_k) = \exp[-\tfrac{1}{2}(y_k - \mu_k)^T Y (y_k - \mu_k)]$$

Stochastic Triggering

Consider another triggering rule:

Stochastic Triggering:

$$\xi_k \sim U(0,1), \quad \Phi(y_k - \mu_k) \in [0,1]$$

$$\gamma_k = \begin{cases} 0 \text{ if } \xi_k \leq \Phi(y_k - \mu_k) \\ 1 \text{ else} \end{cases}$$

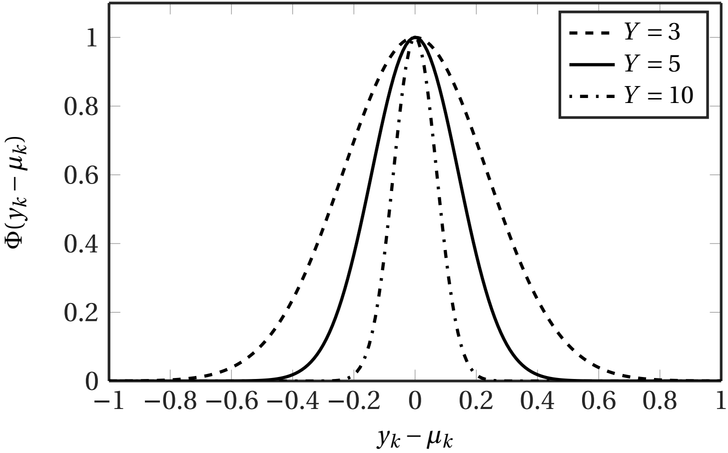
$$\implies \Pr(\gamma_k = 0) = \Phi(y_k - \mu_k)$$

Nice properties when Φ is a scaled Gaussian:

Decision function:

$$\Phi(y_k - \mu_k) = \exp[-\tfrac{1}{2}(y_k - \mu_k)^T Y (y_k - \mu_k)]$$

Scalar case:



- Design variable Y used in same way as Δ

The MMSE Estimator

$\Pr(\gamma_k = 0) = \Phi(y_k - \mu_k)$ has Gaussian shape

$\implies$ closed-form MMSE estimator derivable with Bayes' theorem

Assume sensor transmitted l steps ago. Let $\mu_k = y_{last} = y_{k-l}$

Stochastic Send-on-Delta MMSE Estimator:

Time Update:

$$\hat{x}_{k|k-1} = A\hat{x}_{k-1|k-1}$$

$$\hat{y}_{k|k-1} = C\hat{x}_{k-1|k-1}$$

$$P_{k|k-1} = AP_{k-1|k-1}A^T + Q$$

Measurement Update:

$$\hat{x}_{k|k} = \hat{x}_{k|k-1} + K_k[\gamma_k y_k + (1-\gamma_k)y_{k-l} - \hat{y}_{k|k-1}]$$

$$P_{k|k} = [I - K_k C]P_{k|k-1}$$

$$K_k = P_{k|k-1}C^T[CP_{k|k-1}C^T + R + (1-\gamma_k)Y^{-1}]^{-1}$$

The MMSE Estimator

$\Pr(\gamma_k = 0) = \Phi(y_k - \mu_k)$ has Gaussian shape
$\implies$ closed-form MMSE estimator derivable with Bayes' theorem

Assume sensor transmitted l steps ago. Let $\mu_k = y_{last} = y_{k-l}$

Stochastic Send-on-Delta MMSE Estimator:

Time Update:

$$\hat{x}_{k|k-1} = A\hat{x}_{k-1|k-1}$$
$$\hat{y}_{k|k-1} = C\hat{x}_{k-1|k-1}$$
$$P_{k|k-1} = AP_{k-1|k-1}A^T + Q$$

Measurement Update:

$$\hat{x}_{k|k} = \hat{x}_{k|k-1} + K_k[\gamma_k y_k + (1-\gamma_k) y_{k-l} - \hat{y}_{k|k-1}]$$
$$P_{k|k} = [I - K_k C]P_{k|k-1}$$
$$K_k = P_{k|k-1}C^T[CP_{k|k-1}C^T + R + (1-\gamma_k)Y^{-1}]^{-1}$$

The MMSE Estimator

$\Pr(\gamma_k = 0) = \Phi(y_k - \mu_k)$ has Gaussian shape
$\Longrightarrow$ closed-form MMSE estimator derivable with Bayes' theorem

Assume sensor transmitted l steps ago. Let $\mu_k = y_{last} = y_{k-l}$

Stochastic Send-on-Delta MMSE Estimator:

Time Update:

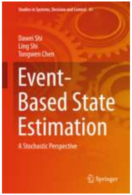
$$\hat{x}_{k|k-1} = A\hat{x}_{k-1|k-1}$$
$$\hat{y}_{k|k-1} = C\hat{x}_{k-1|k-1}$$
$$P_{k|k-1} = AP_{k-1|k-1}A^T + Q$$

Measurement Update:

$$\hat{x}_{k|k} = \hat{x}_{k|k-1} + K_k[\gamma_k y_k + (1-\gamma_k) y_{k-l} - \hat{y}_{k|k-1}]$$
$$P_{k|k} = [I - K_k C]P_{k|k-1}$$
$$K_k = P_{k|k-1}C^T[CP_{k|k-1}C^T + R + (1-\gamma_k)Y^{-1}]^{-1}$$

The Choice of μ_k

Others were having the same idea...



In [Shi et.al 2016] two other choices of μ_k are considered:

Open-Loop Scheme:

$$\mu_k = 0$$

- Stable systems will have zero-mean y_k in stationarity

Closed-Loop Scheme:

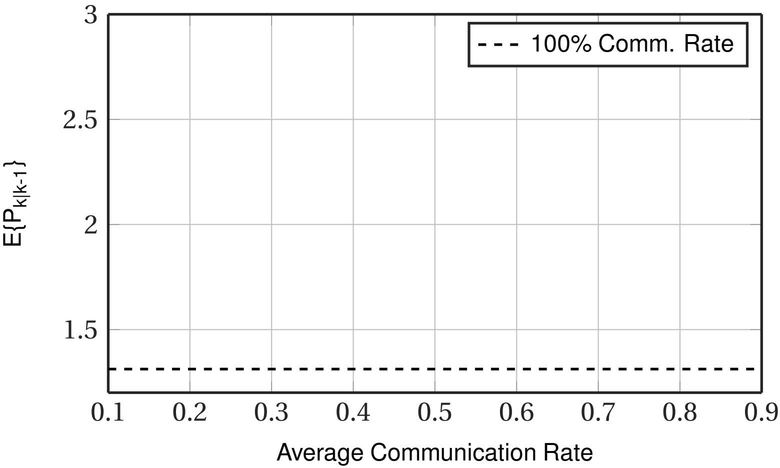
$$\mu_k = \hat{y}_{k|k-1}$$

- Works both for stable and unstable systems
- **Note:** Requires feedback from observer!

Performance Comparison: First-Order

First-order example from [Shi et.al, 2016]:

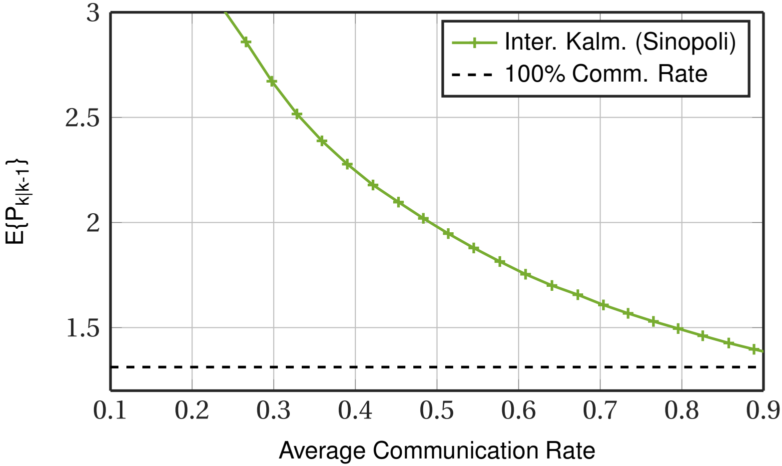
$A = 0.95,\ C = 1,\ Q = 0.8,\ R = 1$



Performance Comparison: First-Order

First-order example from [Shi et.al, 2016]:

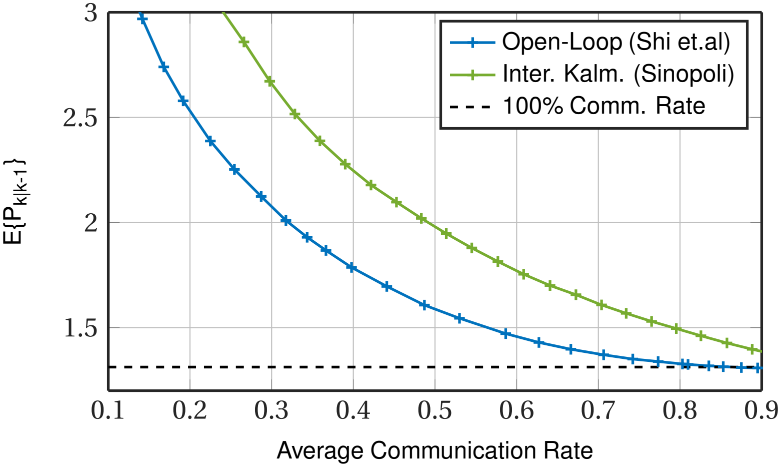
$$A = 0.95,\ C = 1,\ Q = 0.8,\ R = 1$$



Performance Comparison: First-Order

First-order example from [Shi et.al, 2016]:

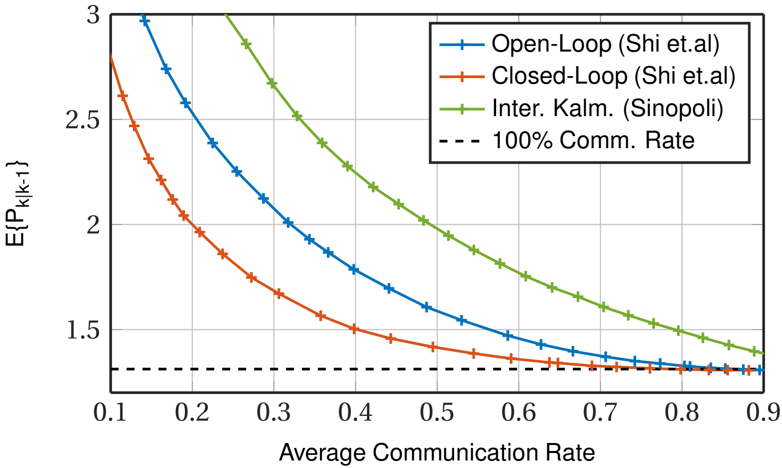
$$A = 0.95,\ C = 1,\ Q = 0.8,\ R = 1$$



Performance Comparison: First-Order

First-order example from [Shi et.al, 2016]:

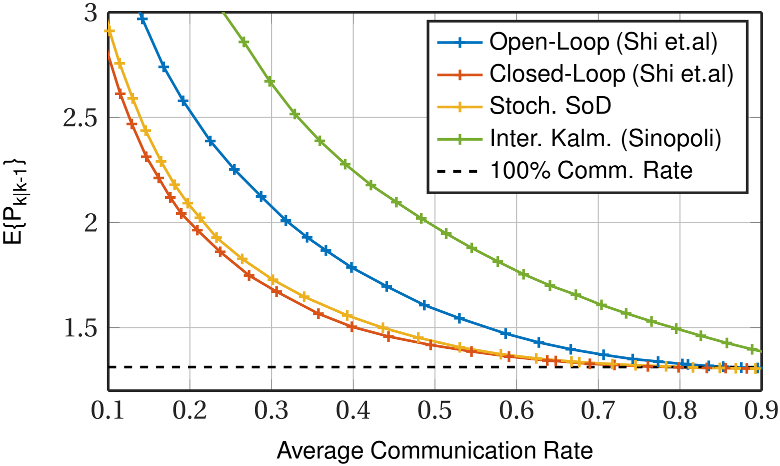
$A = 0.95,\ C = 1,\ Q = 0.8,\ R = 1$



Performance Comparison: First-Order

First-order example from [Shi et.al, 2016]:

$$A = 0.95,\ C = 1,\ Q = 0.8,\ R = 1$$



Performance Comparison: First-Order

First-Order Example

- A lot to gain by using triggering information
- Stoch. SoD better than Open-Loop, closer to Closed-Loop. **No feedback required**
- Is this always the case?

Try again with poorly damped ($\zeta \approx 0.05$) and noisy 2nd order system:

$$A = \begin{bmatrix} 0.8 & -0.55 \\ 0.55 & 0.8 \end{bmatrix}, C = \begin{bmatrix} 1 & 0 \end{bmatrix}$$

$$Q = \begin{bmatrix} 10^{-2} & 0 \\ 0 & 10^{-2} \end{bmatrix}, R = 2$$

Performance Comparison: First-Order

First-Order Example

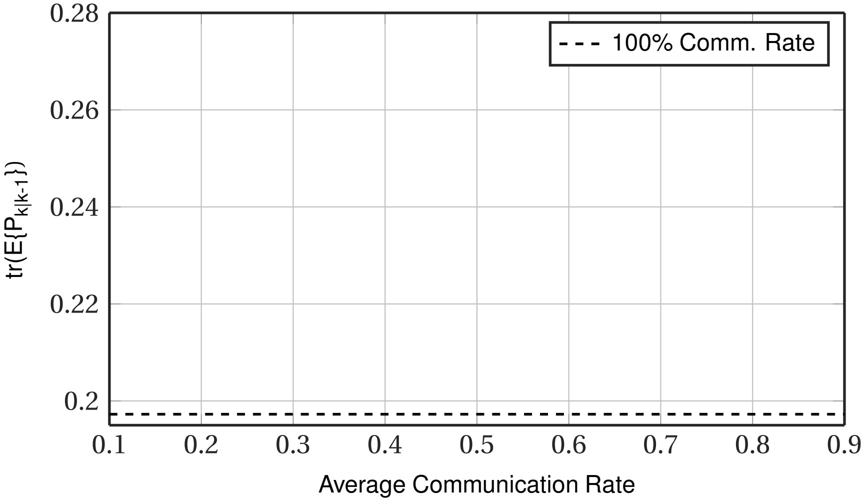
- A lot to gain by using triggering information
- Stoch. SoD better than Open-Loop, closer to Closed-Loop. **No feedback required**
- Is this always the case?

Try again with poorly damped ($\zeta \approx 0.05$) and noisy 2nd order system:

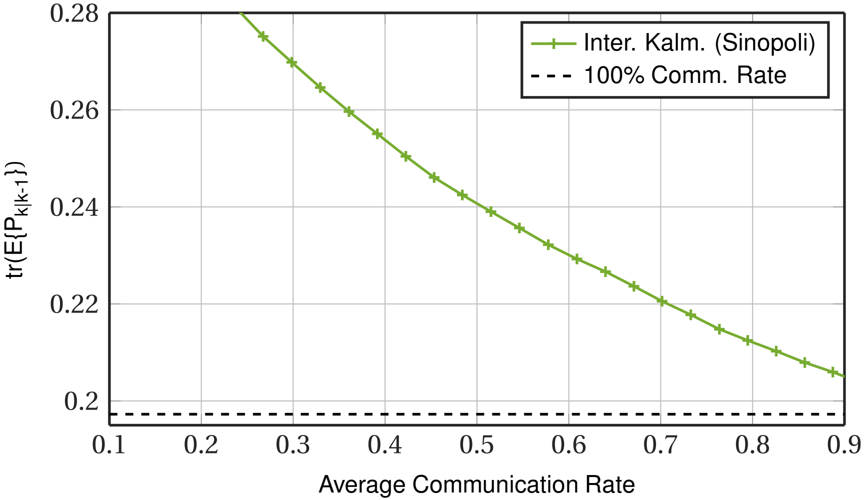
$$A = \begin{bmatrix} 0.8 & -0.55 \\ 0.55 & 0.8 \end{bmatrix}, C = \begin{bmatrix} 1 & 0 \end{bmatrix}$$

$$Q = \begin{bmatrix} 10^{-2} & 0 \\ 0 & 10^{-2} \end{bmatrix}, R = 2$$

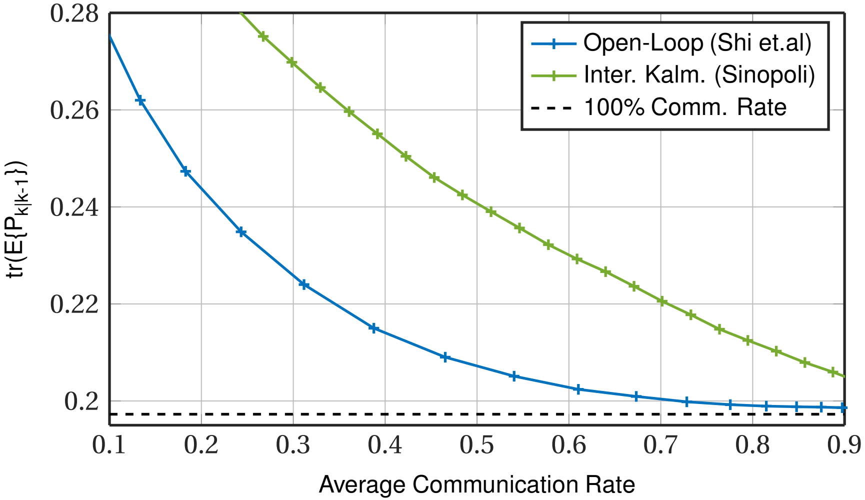
Performance Comparison: Second-Order



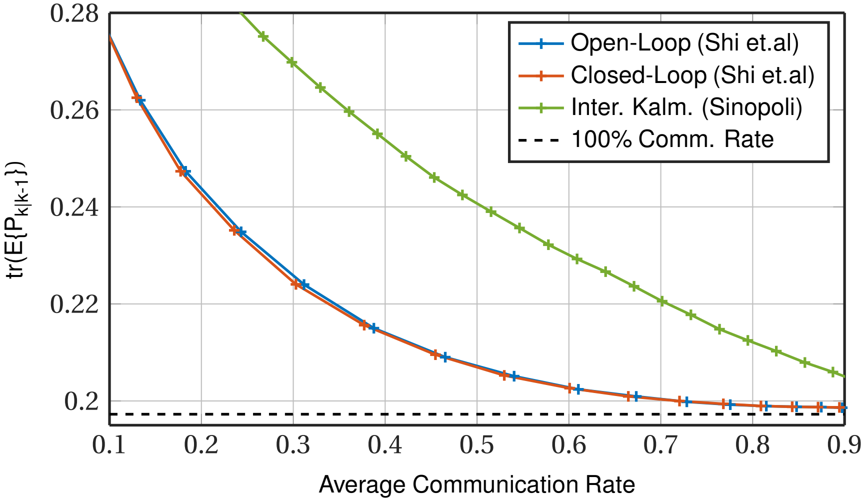
Performance Comparison: Second-Order



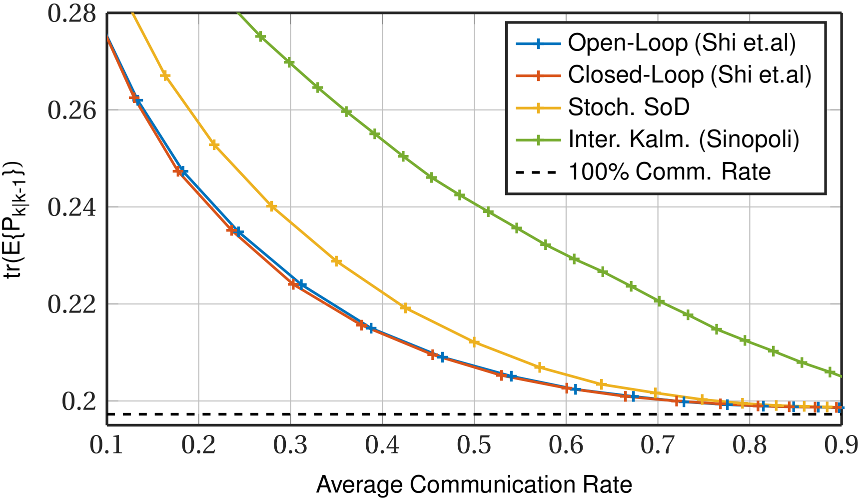
Performance Comparison: Second-Order



Performance Comparison: Second-Order



Performance Comparison: Second-Order



Performance Comparison: Second-Order

For fast and/or noisy systems, $\mu_k = y_{k-l}$ is a poor prediction.
Stoch. SoD worse than Open-Loop!

Q: **Can we improve while keeping μ_k simple?**

Idea: A Simple Prediction

For stable systems, in stationarity:

Prediction

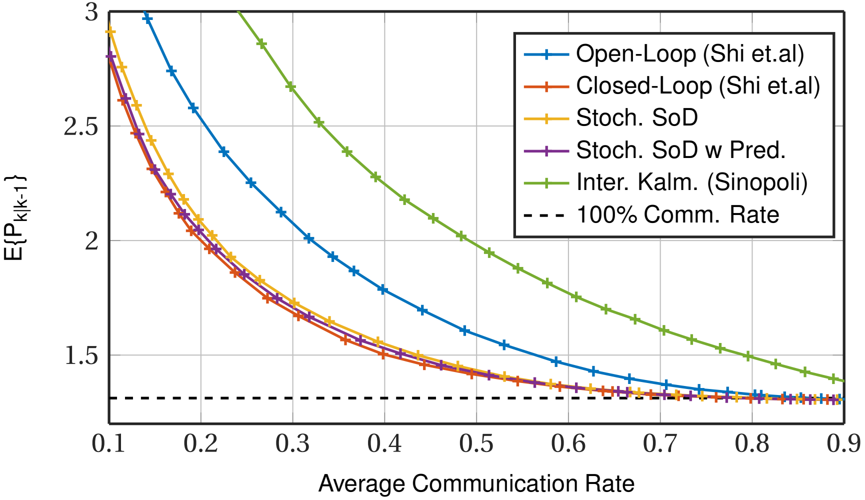
$$\mu_k = E\{y_k | y_{k-l}\} = S_l y_{k-l}, \quad S_l = CA^l \Sigma C^T [C\Sigma C^T + R]^{-1}$$

since $E\{x_k\} = 0$ and $E\{x_k x_k^T\} = \Sigma$ where $\Sigma = A\Sigma A^T + Q$

- A simple scaling, dependent on l
- Pre-compute S_l and store in look-up table
- MMSE Estimator same as for regular Stoch. SoD. Just change y_{k-l} to $S_l y_{k-l}$.

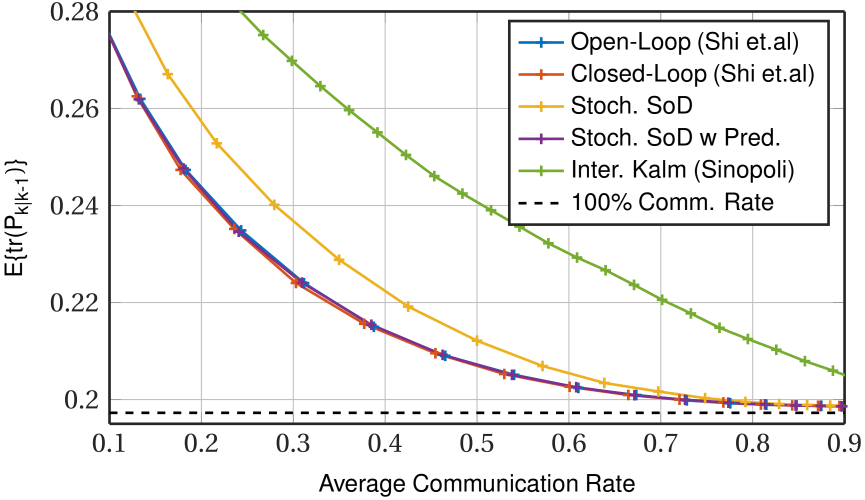
Performance Comparison Revisited

Modification performs well in first example...



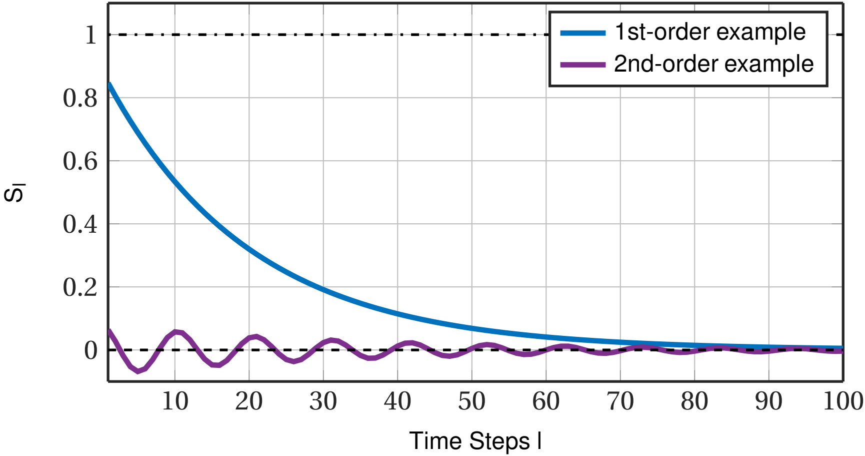
Performance Comparison Revisited

... and also in second example



Open-Loop or Stoch. SoD?

Open-Loop: "$S_l = 0$"   Stoch. SoD: "$S_l = 1$"



Outline

Outline of this seminar

- The Remote Estimation Problem
- Current Research: Stochastic Triggering
- Further Work & Summary

Ongoing Work: Average Communication Rate

Desirable to know communication rate for given Y

- **Open-Loop:** Closed-form expression exists
- **Closed-Loop:** (Loose) Bounds exists
- **Stoch. SoD:** Pr($\gamma_k = 0$| sensor info) not time invariant, depends on explicit value of y_{k-l}
- **Modified Stoch. SoD:** Pr($\gamma_k = 0$|sensor info) independent of explicit value of y_{k-l}. Current work.

Further Work: Other Properties

- Covariance bounds exists. **Tighter bounds** on covariance?
- **Controller co-design** when using Stochastic Triggering?

Summary

- **When sensing and transmitting has a cost**, consider event-based estimation
- Potential for greater resource efficiency in e.g **wireless network applications**
- Current research: Event-Based Estimation using **Stochastic Triggering**
- Stoch. SoD with Simple Prediction outperforms regular Stoch. SoD and Open-Loop **without feedback**
- Properties such as average communication rate still open.

Questions?