Communication constraints and latency in Networked Control Systems

João P. Hespanha

Center for Control Engineering
and Computation

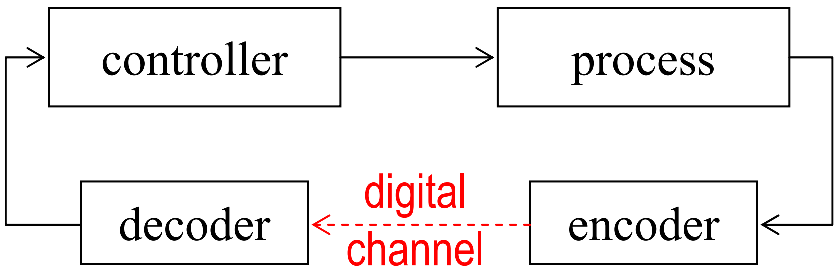
University of California
Santa Barbara

UCSB

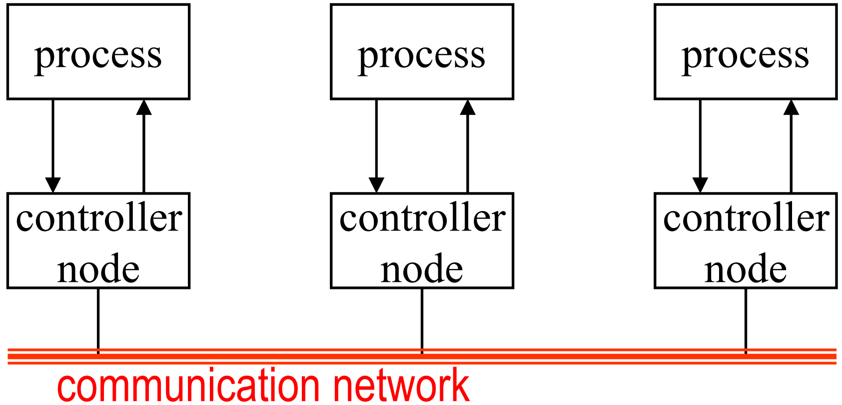
In collaboration with Antonio Ortega (USC) and Yonggang Xu (UCSB)

Outline

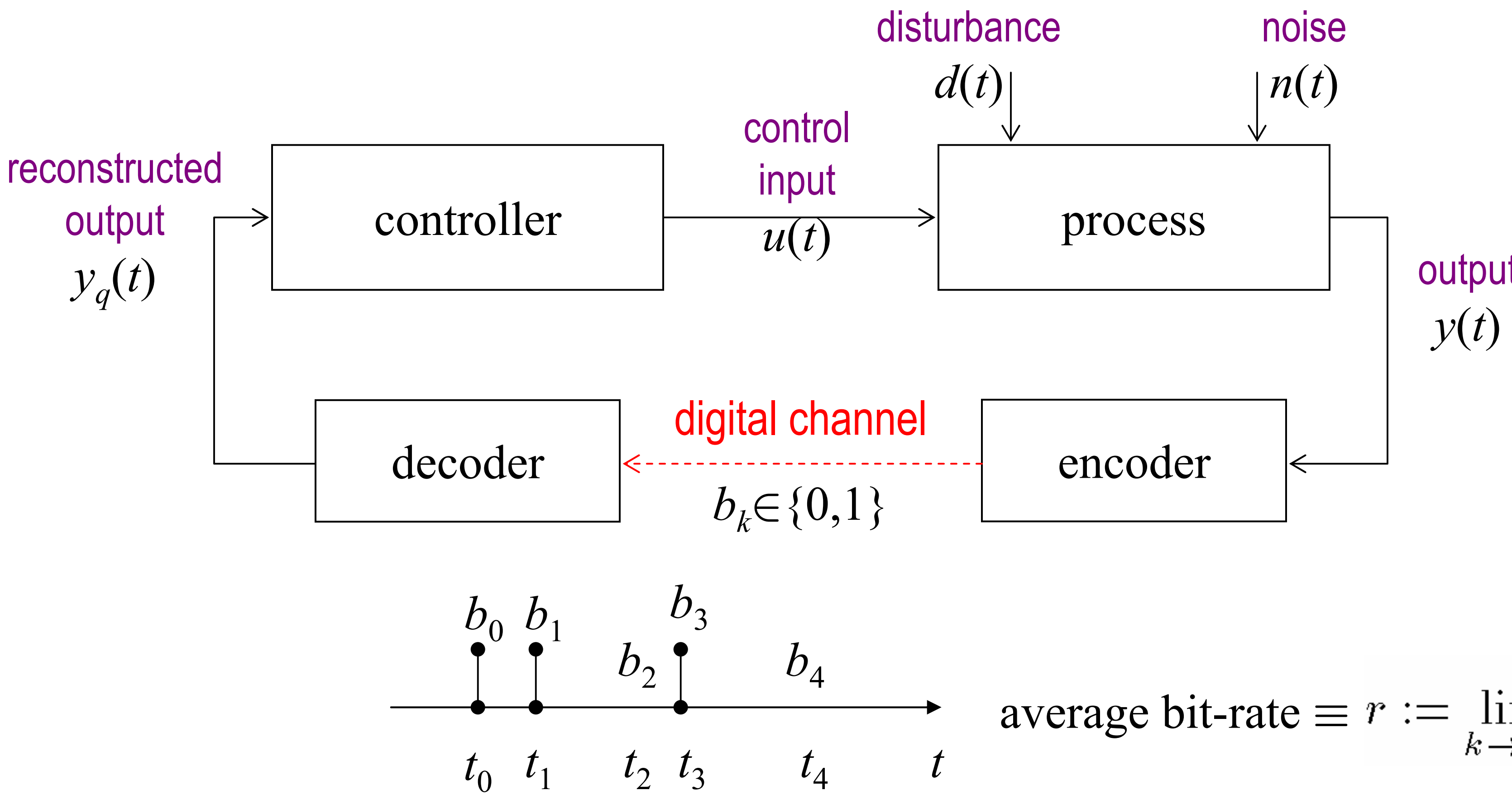
1. Feedback control over a digital channel with limited bit-rate:



2. Decentralized cooperative control with limited message-rate and delays



Control with finite bit-rate

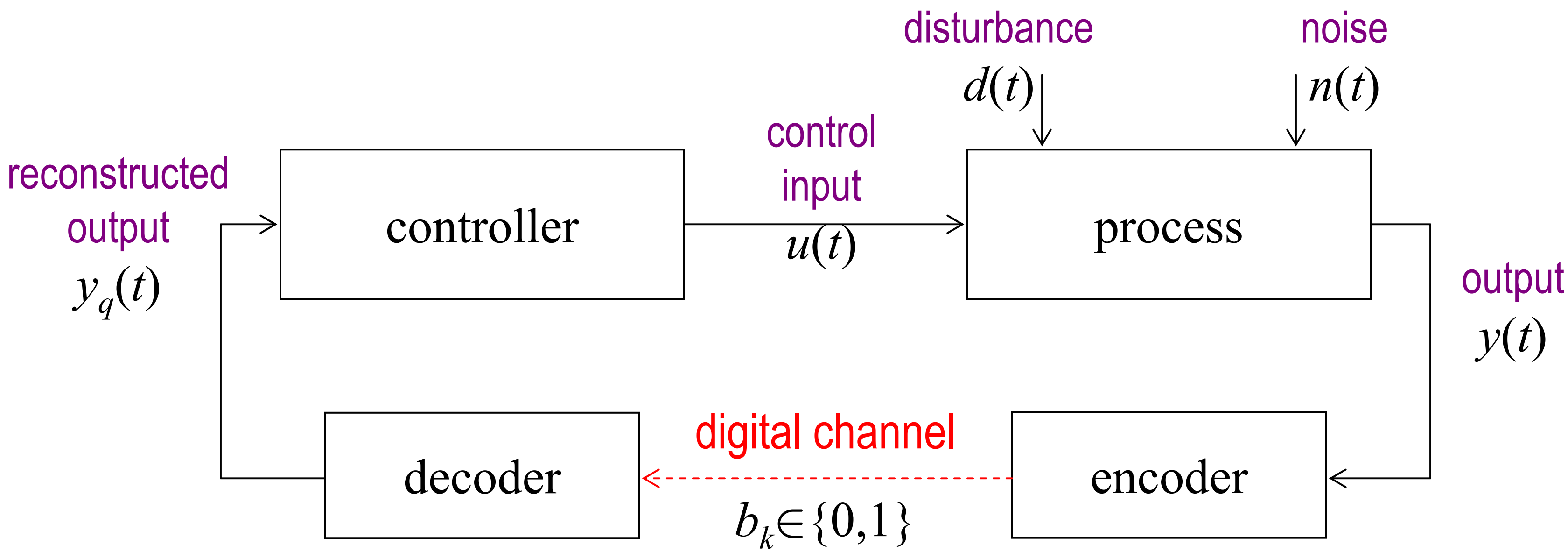


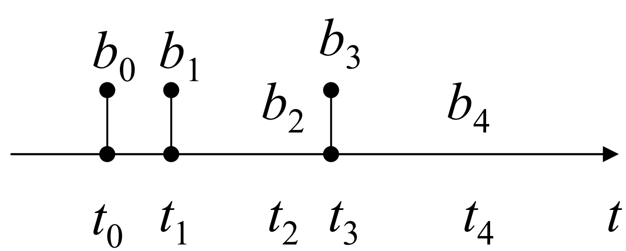
average bit-rate $\equiv r := \lim_{k \to \infty} \frac{k}{t_k}$

Motivation: *Control of systems with sensors and actuators far from each other, connected by a digital network.*

E.g., control of an autonomous flying vehicle using measurements from a camera on the ground.

Control with finite bit-rate





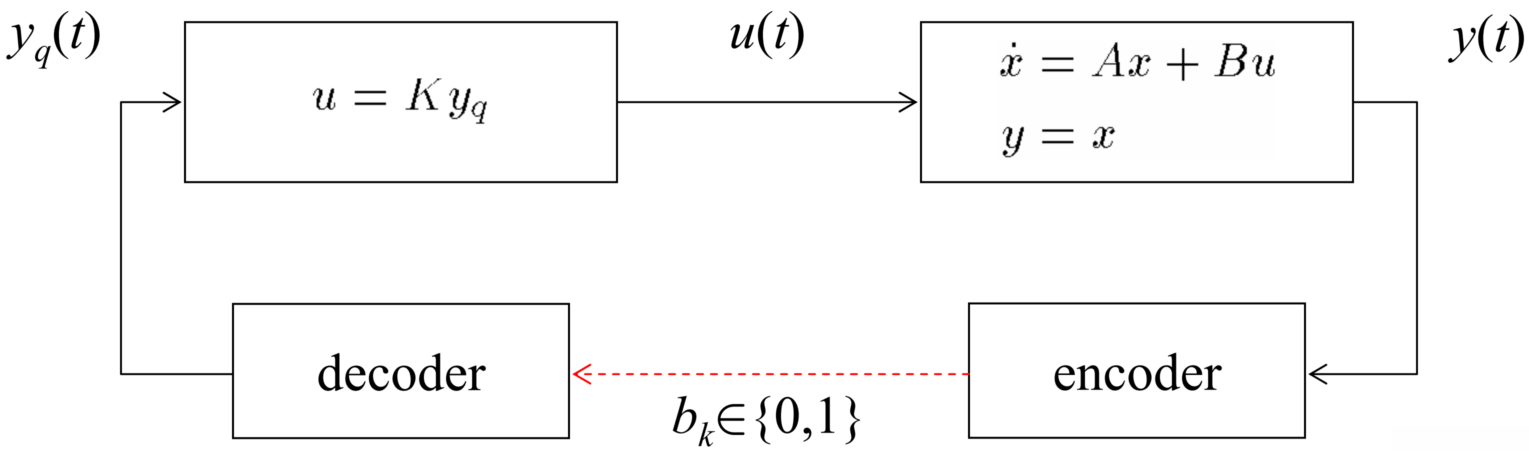
average bit-rate $\equiv r := \lim_{k\to\infty} \frac{k}{t_k}$

Questions:

1. *What is the minimum bit-rate for which stabilization (boundedness) is possible?*
2. *How to divide the bits among the distinct components of the output?*
3. *How to choose quantization intervals?*

Minimum bit-rate

no noise/disturbance



average bit-rate $\equiv r := \lim_{k\to\infty} \frac{k}{t_k}$

Theorem: Stabilization is ***not*** possible with average bit-rate smaller than

$$r_{\min} := \frac{1}{\log 2} \sum_{\Re\lambda_i > 0} \lambda_i$$

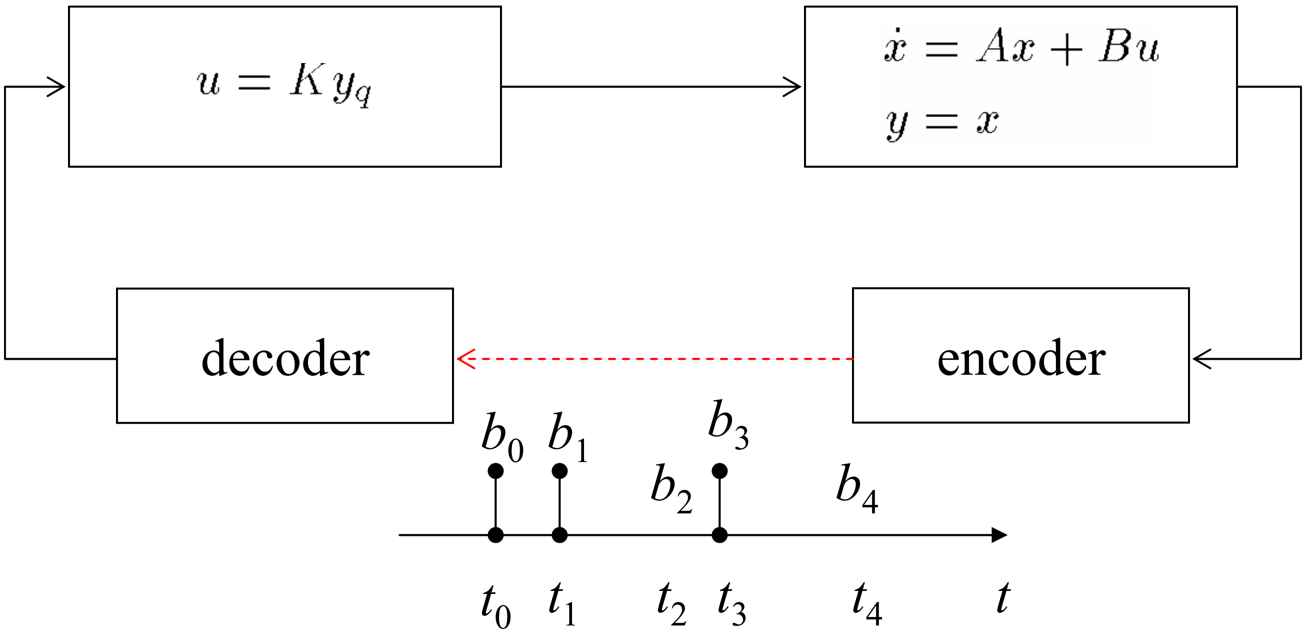
continuous-time process

$$r_{\min} := \frac{1}{\log 2} \sum_{|\lambda_i| > 1} \log \lambda_i$$

discrete-time process
[Tatikonda & Mitter]

$\lambda_i \equiv$ eigenvalues of A

Proof outline

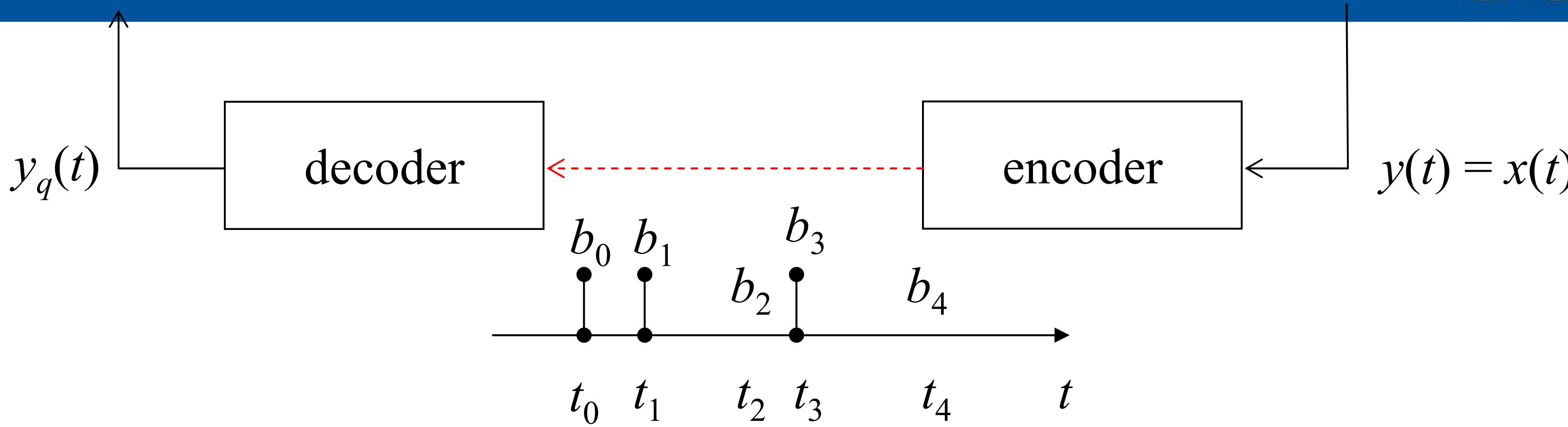


A stable ⇒ x decays to zero without control ⇒ minimum bit-rate is zero (just set $u := 0$)

A has asympt. stable inv. subspace $\mathcal{S}^-$ ⇒ component of x in $\mathcal{S}^-$ goes to zero without control ⇒ can reduce A to its unstable inv. subspace

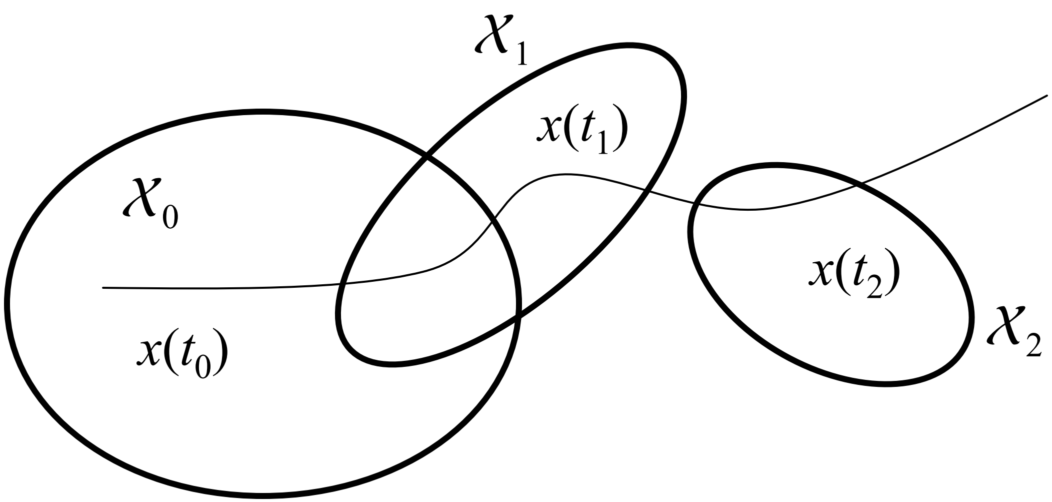
∴ we will assume that all eigenvalues of A have real part ≥ 0

Proof outline

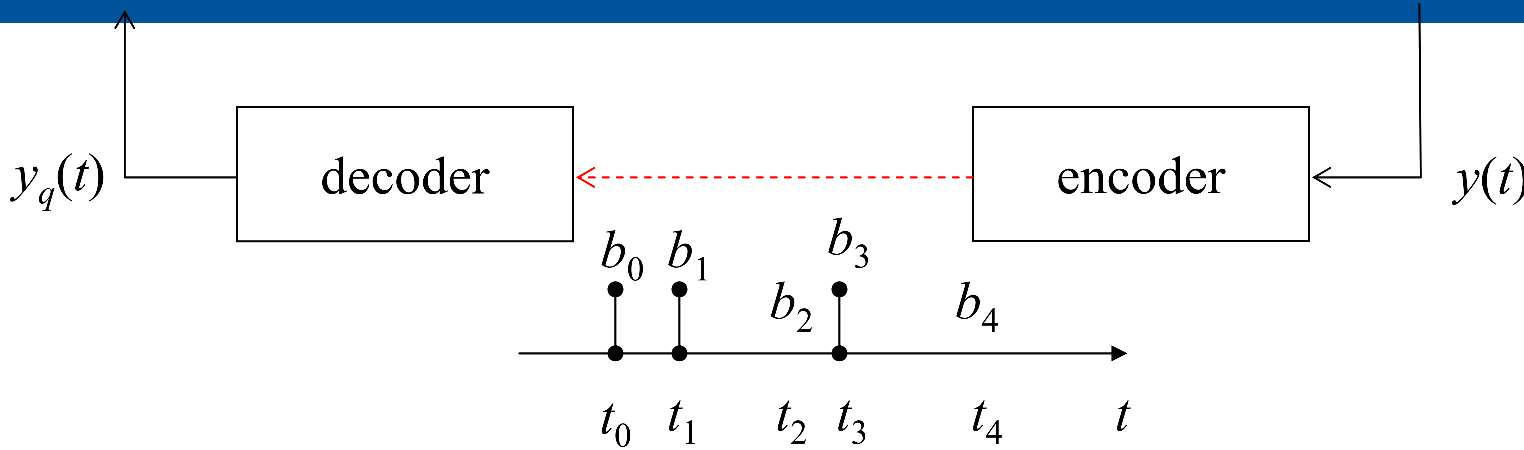


$\mathcal{X}_0 \equiv$ set to which $x(t_0)$ is known to belong after bit b_0 is received

$\mathcal{X}_1 \equiv$ set to which $x(t_1)$ is known to belong after bit b_1 is received …



Proof outline



$\mathcal{X}_0 \equiv$ set to which $x(t_0)$ is known to belong after bit b_0 is received

$\mathcal{X}_1 \equiv$ set to which $x(t_1)$ is known to belong after bit b_1 is received …

Case 1: For some k, $\mathcal{X}_k$ has a single element

$\Rightarrow$ could take x to zero (even in finite time) after t_k

diameter of $\mathcal{X}_k$

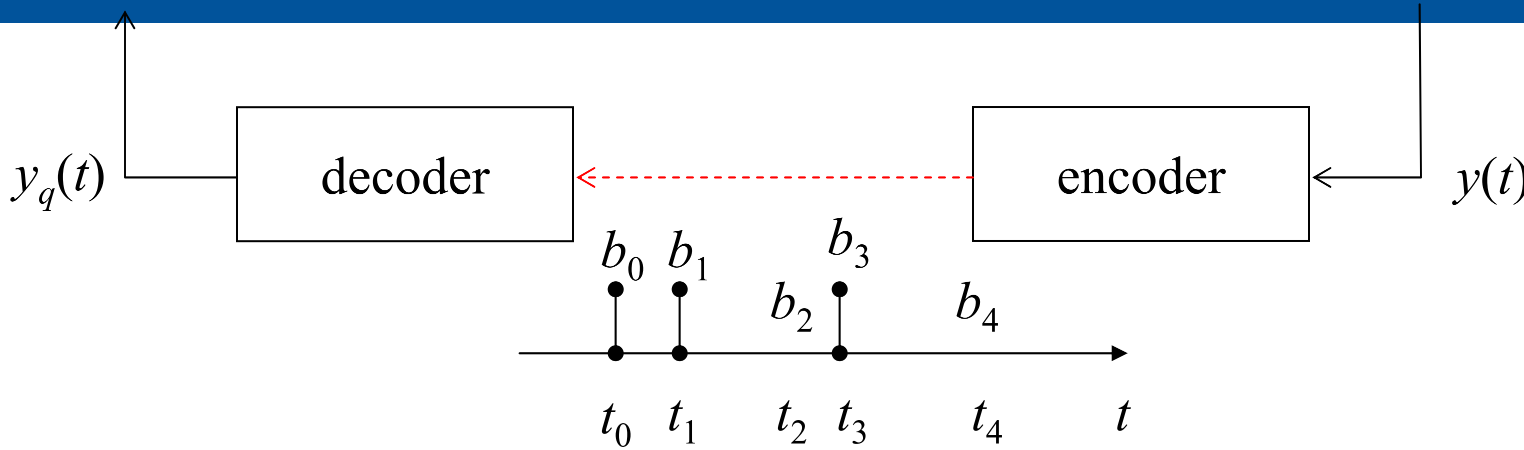
Case 2: As $k \to \infty$, $\rho(\mathcal{X}_k) \to 0$

$\Rightarrow$ could take x to zero as $k \to \infty$

Case 3: As $k \to \infty$, $\rho(\mathcal{X}_k)$ unbounded

$\Rightarrow$ no matter what control we use, $x(t_k)$ is unbounded

Proof outline



$\mathcal{X}_0 \equiv$ set to which $x(t_0)$ is known to belong after bit b_0 is received
$\mathcal{X}_1 \equiv$ set to which $x(t_1)$ is known to belong after bit b_1 is received …

Case 1: For some k, $\mathcal{X}_k$ has a single element

$\Rightarrow$ could take x to zero (even in finite time) after t_k

Case 2: As $k \to \infty$, $\rho(\mathcal{X}_k) \to 0$ (diameter of $\mathcal{X}_k$)

$\Rightarrow$ could take x to zero as $k \to \infty$

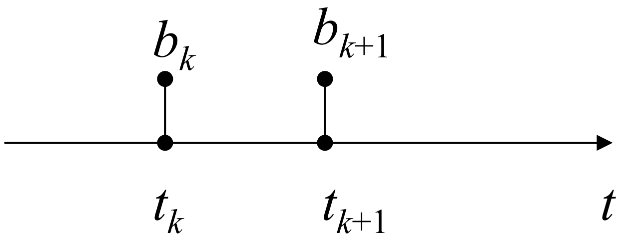
Case 3: As $k \to \infty$, $\rho(\mathcal{X}_k)$ unbounded

$\Rightarrow$ no matter what control we use, $x(t_k)$ is unbounded

Main idea:

bit rate $\leq r_{min}$ $\Rightarrow$ $\mu(\mathcal{X}_k)$ unbounded (volume of $\mathcal{X}_k$) $\Rightarrow$ $\rho(\mathcal{X}_k)$ unbounded (diameter of $\mathcal{X}_k$) $\Rightarrow$ *Stabilization not possible*

Proof outline



$$x(t_{k+1}) = e^{A(t_{k+1}-t_k)}\underbrace{x(t_k)}_{\in \mathcal{X}_k} + \underbrace{\int_{t_k}^{t_{k+1}} e^{A(\tau - t_k)} u(\tau) d\tau}_{u_k}$$

before b_{k+1} is received, it is only known that

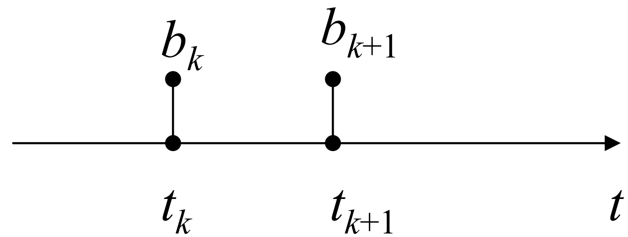
$$x(t_{k+1}) \in \mathcal{X}_{k+1}^{-} := e^{A(t_{k+1}-t_k)}\mathcal{X}_k + u_k$$

$$\mu(\mathcal{X}_{k+1}^{-}) = |\det e^{A(t_{k+1}-t_k)}|\, \mu(\mathcal{X}_k)$$

volume of $\mathcal{X}^{-}_{k+1}$

volume of $\mathcal{X}_k$

Proof outline



$$x(t_{k+1}) = e^{A(t_{k+1}-t_k)} \underbrace{x(t_k)}_{\in \mathcal{X}_k} + \underbrace{\int_{t_k}^{t_{k+1}} e^{A(\tau - t_k)} u(\tau) d\tau}_{u_k}$$

before b_{k+1} is received, it is only known that

$$x(t_{k+1}) \in \underbrace{\mathcal{X}_{k+1}^{-} := e^{A(t_{k+1}-t_k)} \mathcal{X}_k + u_k}$$

$$\mu(\mathcal{X}_{k+1}^{-}) = |\det e^{A(t_{k+1}-t_k)}| \, \mu(\mathcal{X}_k)$$

volume of $\mathcal{X}^{-}_{k+1}$ volume of $\mathcal{X}_k$

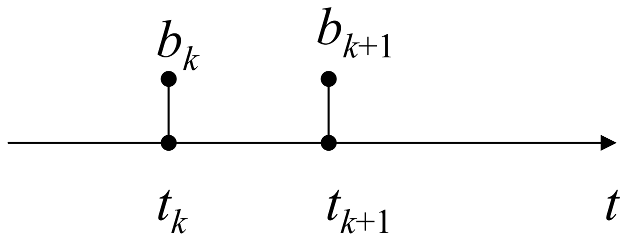
Q: Which coding would make $\mu(\mathcal{X}_{k+1})$ as small as possible?

A: Divide $\mathcal{X}^{-}_{k+1}$ into two sets of equal volume & use bit b_{k+1} to locate $x(t_{k+1})$ in one of them

coding that minimizes volume

$$\mu(\mathcal{X}_{k+1}) = \frac{\mu(\mathcal{X}_{k+1}^{-})}{2}$$

Proof outline



At best…

$$\mu(\mathcal{X}_{k+1}) \geq \frac{\mu(\mathcal{X}_{k+1}^{-})}{2} = \frac{|\det e^{A(t_{k+1}-t_k)}|}{2}\,\mu(\mathcal{X}_k) = e^{(t_{k+1}-t_k)\sum \lambda_i}\,\mu(\mathcal{X}_k)$$

with = only for coding that minimizes volume

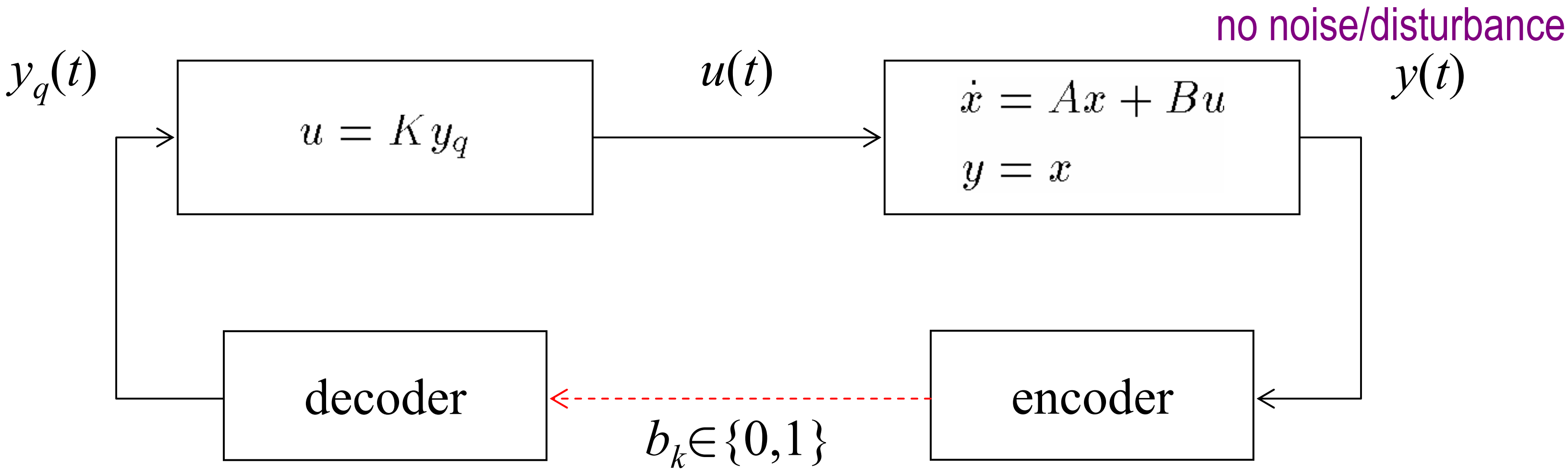
iterating…

$$\mu(\mathcal{X}_{k+1}) \geq e^{t_k(\sum \lambda_i)-k\log 2}\,\mu(\mathcal{X}_k)$$

explodes if average bit-rate is smaller than

$$r_{\min} := \frac{1}{\log 2}\sum \lambda_i$$

Minimum bit-rate



average bit-rate $\equiv r := \lim_{k \to \infty} \frac{k}{t_k}$

Theorem: Stabilization is ***not*** possible with average bit-rate smaller than

$$r_{\min} := \frac{1}{\log 2} \sum_{\Re \lambda_i > 0} \lambda_i$$

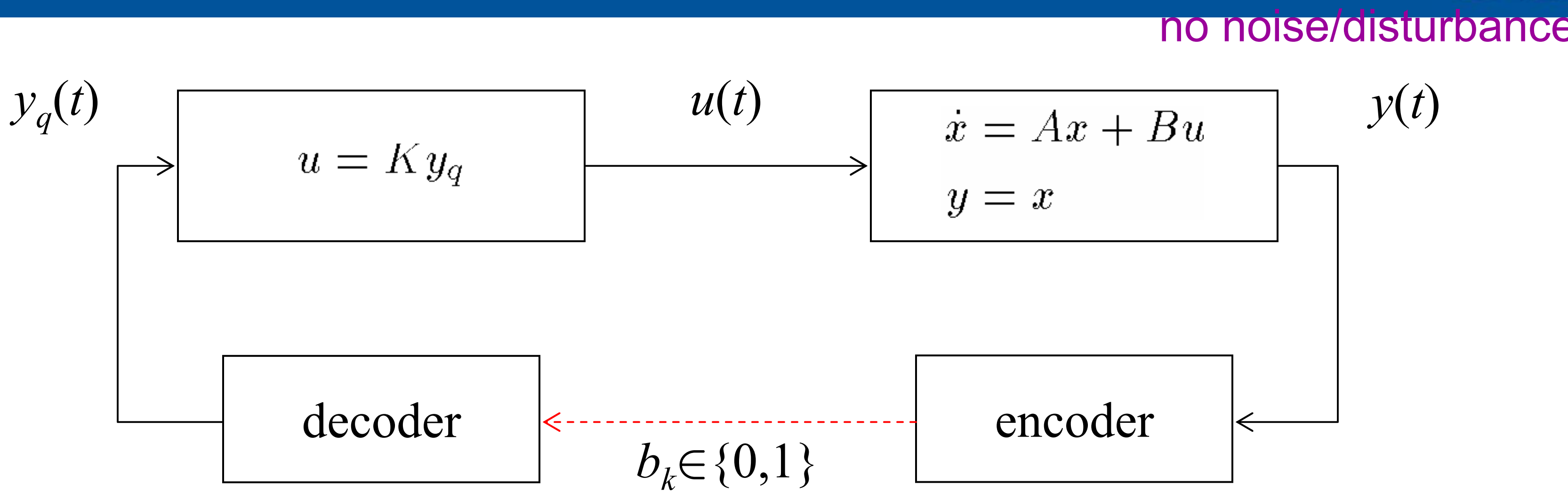
continuous-time process

$$r_{\min} := \frac{1}{\log 2} \sum_{|\lambda_i| > 1} \log \lambda_i$$

discrete-time process
[Tatikonda & Mitter]

$\lambda_i \equiv$ eigenvalues of A

Minimum bit-rate



average bit-rate $\equiv r := \lim_{k \to \infty} \frac{k}{t_k}$

Theorem: Assume A is diagonalizable

1. It is possible to keep the state bounded with any average bit rate larger or equal to $r_{\min}$
2. It is possible to make the state converge to zero with any average bit rate strictly larger than $r_{\min}$

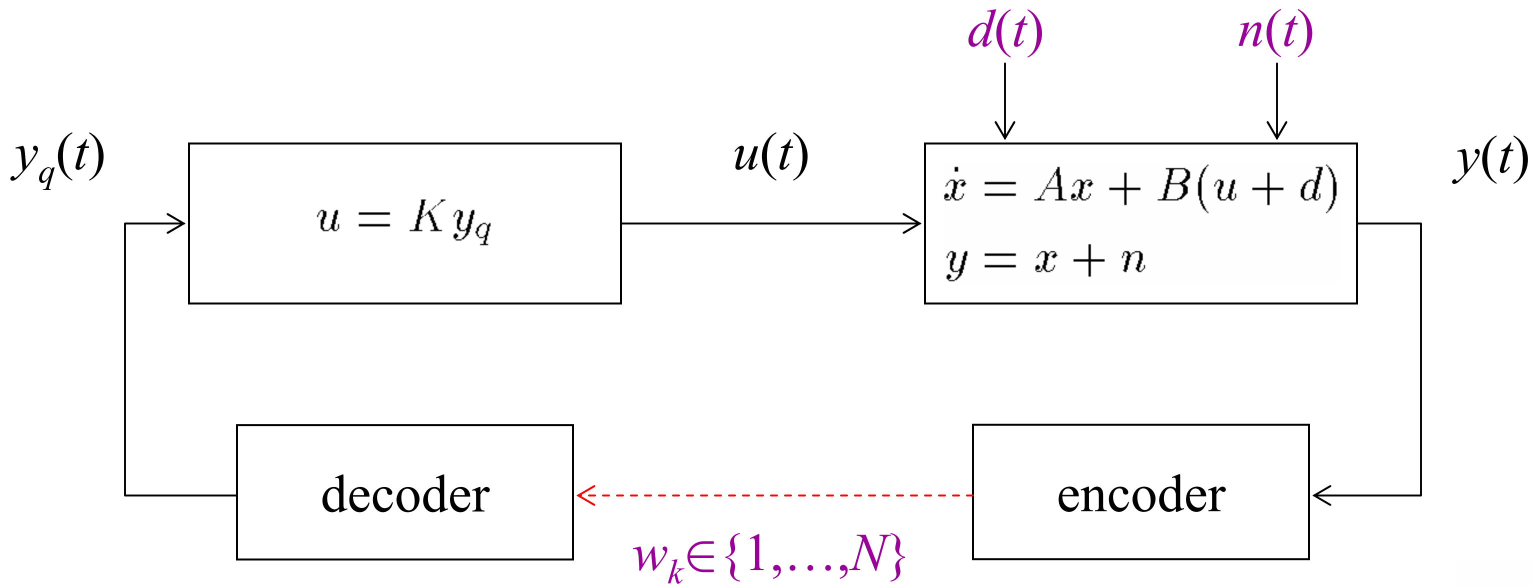
} previous bound is tight

But … minimum volume coding may not work because

unbounded volume $\Rightarrow$ unbounded diameter

bounded volume $\not\Rightarrow$ bounded diameter

Encoding/decoding schemes



$T_s \equiv$ (fixed) sampling interval

bit-rate $\equiv r := \dfrac{\log_2 N}{T_s}$

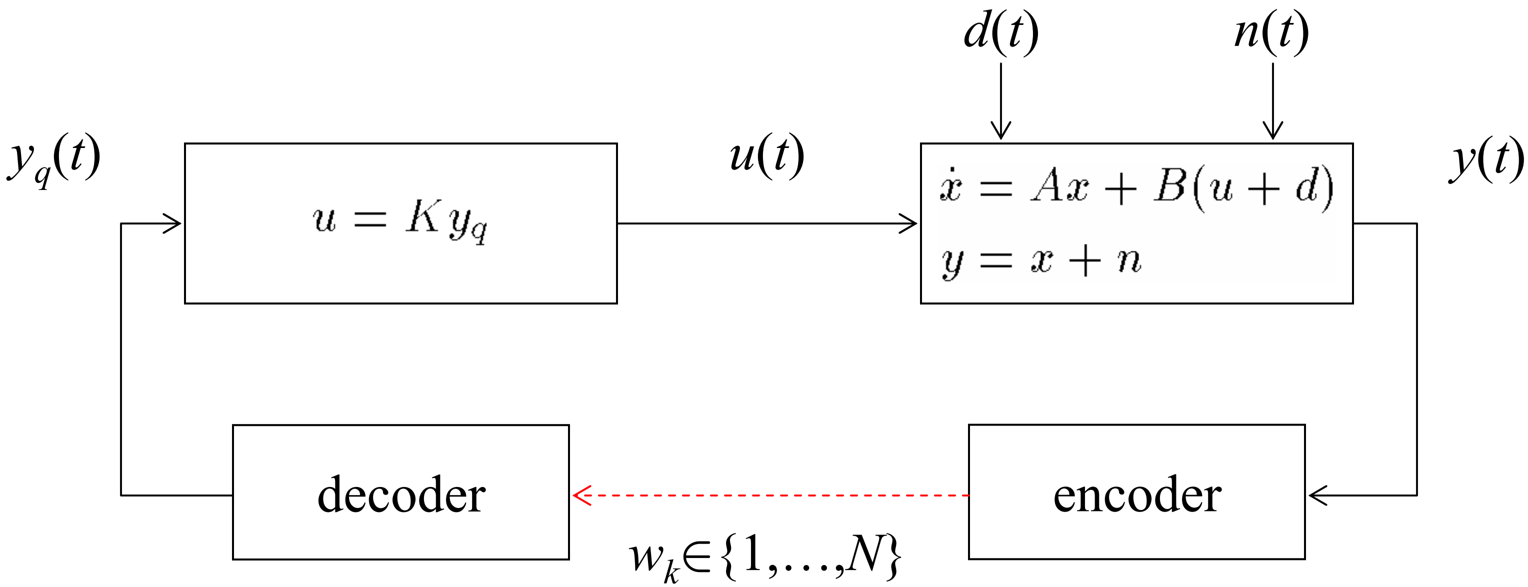
Assume: closed-loop is stable for "transparent" encoding/decoding, i.e.,

$A + B\,K$ asymptotically stable

Questions:

1. *How to design the encoder/decoder pair to make the closed-loop stable?*
2. *How much larger than r_{min} does the bit-rate need to be for stability?*

State-prediction coding



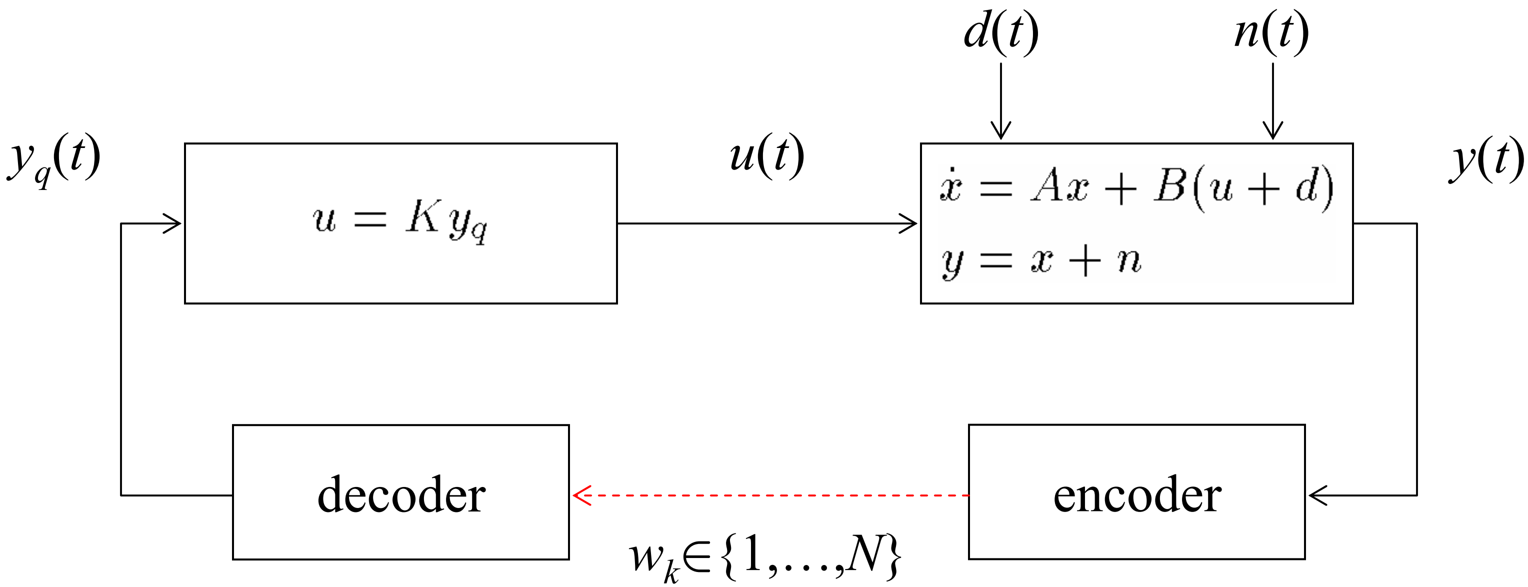
Inspired by *Differential Pulse Code Modulation* (DPCM)…

1. Encoder and decoder maintain consistent estimates of the state, based only on the quantized information sent to the decoder.

2. At sampling times, the difference between the measured state and its estimate (based on previously transmitted data) is quantized and transmitted digitally. (hopefully state estimation error has smaller dynamic range than state itself)

3. Upon transmission, the state-estimates are corrected using the quantized error.

State-prediction coding



1. Encoder and decoder maintain consistent estimates of the state, based only on the quantized information sent to the decoder.

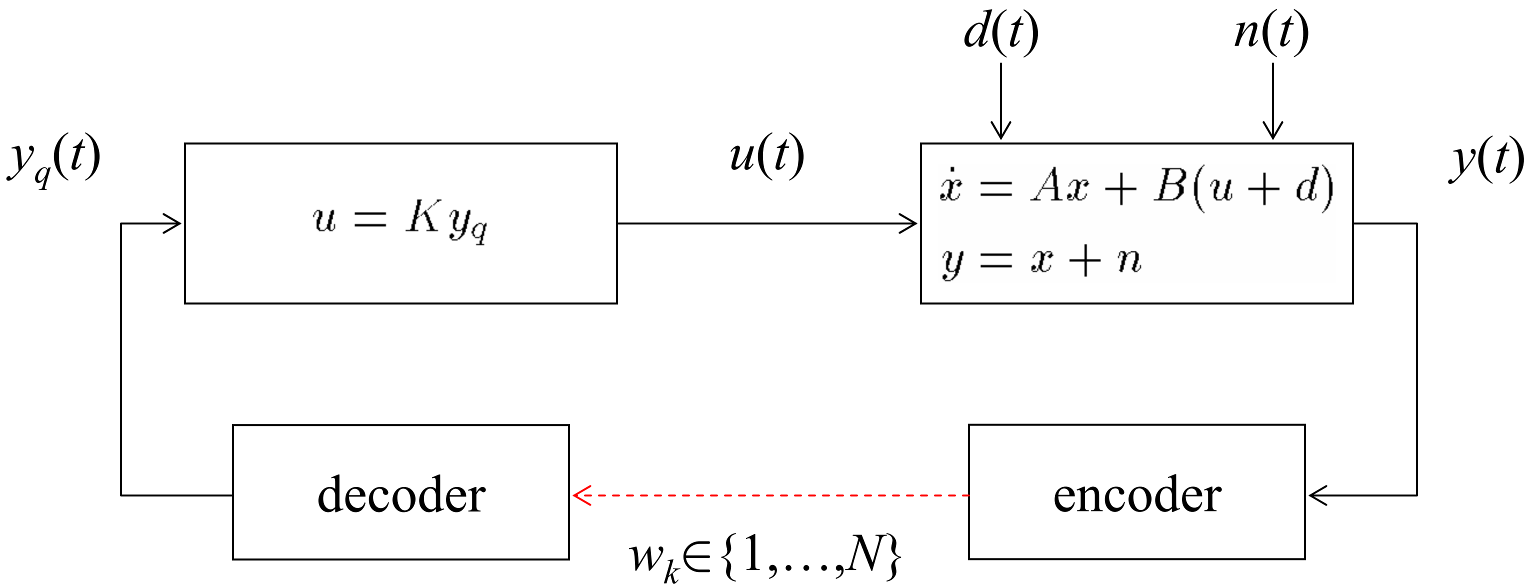
process

$$\dot{x} = Ax + B(u + d)$$
$$y = x + n$$

estimate

$$\dot{\hat{x}} = A\hat{x} + Bu$$
$$\hat{y} = \hat{x}$$

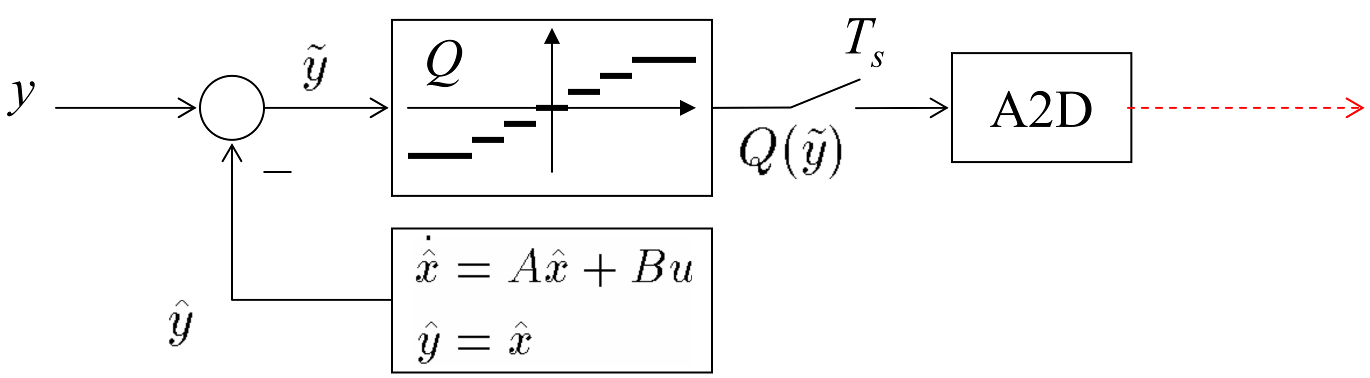
State-prediction coding



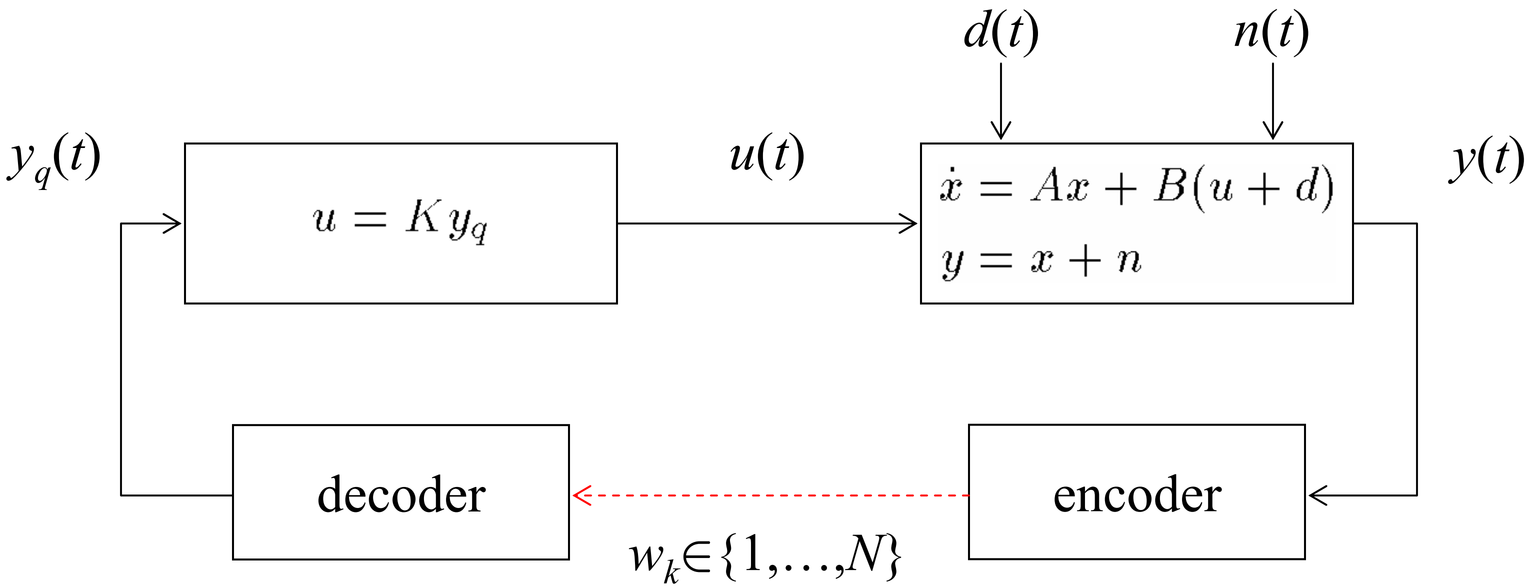
1. Encoder and decoder maintain consistent estimates of the state, based only on the quantized information sent to the decoder.

smaller dynamic range

2. At sampling times, the difference between the measured state and its estimate is quantized and transmitted digitally.



State-prediction coding



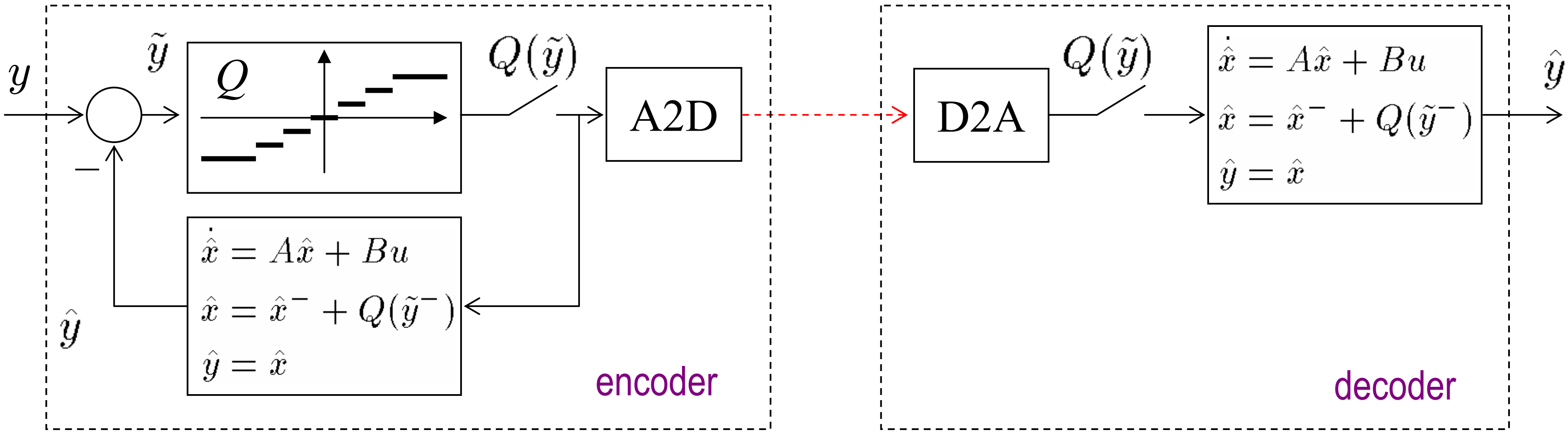
1. Encoder and decoder maintain consistent estimates of the state, based only on the quantized information sent to the decoder.
2. At sampling times, the difference between the measured state and its estimate is quantized and transmitted digitally.
3. Upon transmission, the state-estimates are corrected using the quantized error:

$$\hat{x}(t_k) = \hat{x}(t_k^-) + Q\big(\tilde{y}(t_k^-)\big) = \hat{x}(t_k^-) + Q\Big(y(t_k) - \hat{x}(t_k^-)\Big)$$

without quantization (Q = identity)
and noise, we would have

$$\hat{x}(t_k) = y(t_k) = x(t_k)$$

Encoder/decoder pair



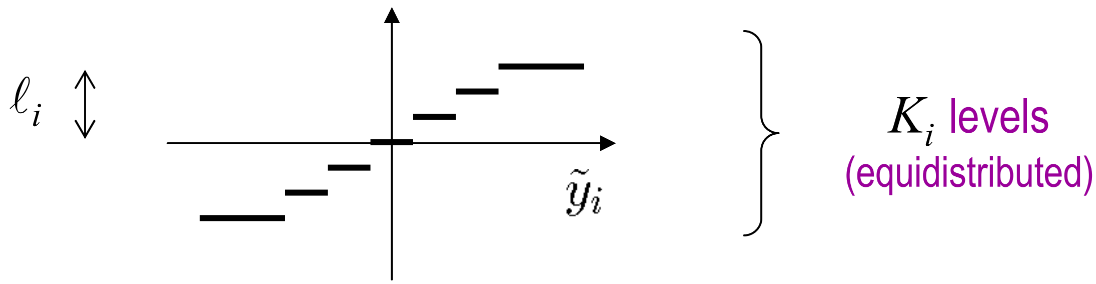
1. Encoder and decoder maintain consistent estimates of the state, based only on the quantized information sent to the decoder.
2. At sampling times, the difference between the measured state and its estimate is quantized and transmitted digitally
3. Upon transmission, the state-estimates are corrected using the quantized error.

Fixed-step quantization

For simplicity…

- assume A diagonal with real eigenvalues
- quantize the vector by using a scalar quantizer on each of its components

If *A* was not diagonal one would precede the component-wise quantization by a diagonalizing linear transformation
(see [JH, Ortega, Vasudevan, 02]
for case of *A* with complex eigenvalues)



$\ell_i \equiv$ saturation level for the ith component quantizer

$K_i \equiv$ # of quantization levels used for the ith component of $\tilde{y}$

# of words needed $\equiv \quad N = \prod_{i=1}^{n} K_i$ $\qquad$ bit-rate $\equiv \dfrac{\log_2 N}{T_s} = \dfrac{\sum_i \log K_i}{T_s \log 2}$

Fixed-step quantization

quantizer saturation level

Theorem: The state of the closed-loop system will remain bounded provided that

$$K_i \geq \frac{\ell_i e^{\lambda_i T_s}}{\ell_i - \eta_i - \delta_i} \qquad \& \qquad |x_i(0)| \leq \ell_i - \eta_i$$

\# of quant. levels

η_i, $\delta_i \equiv$ constants that depend on *upper bounds* on the noise/disturbance

bit allocation…

λ_i large $\Rightarrow e^{\lambda_i T_s}$ large $\Rightarrow K_i$ large $\Rightarrow$ many bits needed for ith eigenspace

Fixed-step quantization

UCSB

quantizer saturation level

Theorem: The state of the closed-loop system will remain bounded provided that

$$K_i \geq \frac{\ell_i e^{\lambda_i T_s}}{\ell_i - \eta_i - \delta_i} \qquad \& \qquad |x_i(0)| \leq \ell_i - \eta_i$$

# of quant. levels

η_i, $\delta_i \equiv$ constants that depend on *upper bounds* on the noise/disturbance

bit allocation…

λ_i large $\Rightarrow e^{\lambda_i T_s}$ large $\Rightarrow K_i$ large $\Rightarrow$ many bits needed for ith eigenspace

required bit-rate…

# of symbols

$$\text{bit-rate} \equiv \frac{\log_2 N}{T_s} \qquad N = \prod_{i=1}^{n} K_i \geq \prod_{i=1}^{n} \max\left\{1, \frac{\ell_i e^{\lambda_i T_s}}{\ell_i - \eta_i - \delta_i}\right\}$$

$$\text{bit-rate} \geq \sum_{K_i > 1} \left(\frac{\lambda_i}{\log 2} + \frac{1}{T_s} \log_2 \frac{\ell_i}{\ell_i - \eta_i - \delta_i} \right)$$

approximately $r_{\min}$ when $\ell_i \gg \eta_i + \delta_i$

(but course quantization leads to large x even without noise/disturbance)

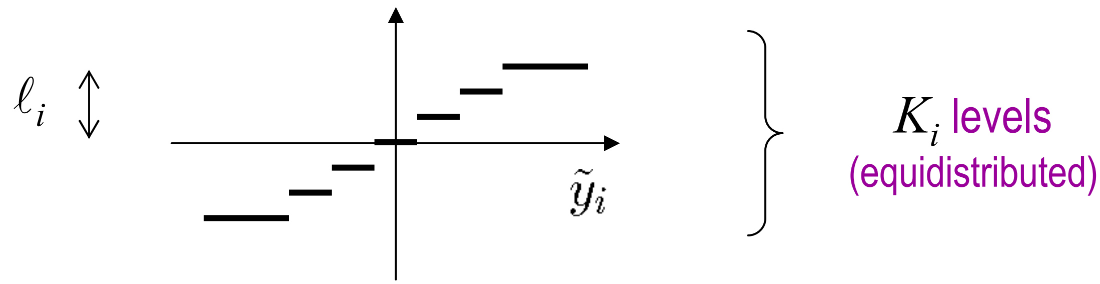
Variable-step quantization

UCSB

For simplicity…

- assume A diagonal with real eigenvalues
- quantize the vector by using a scalar quantizer on each of its components

If A was not diagonal one would precede the component-wise quantization by a diagonalizing linear transformation
(see paper for case of A with complex eigenvalues)



$\ell_i(k) \equiv$ saturation level for the ith component quantizer at sampling time t_k

$K_i \equiv$ # of quantization levels used for the ith component of $\tilde{y}$

# of words needed $\equiv$ $N = \prod_{i=1}^{n} K_i$ bit-rate $\equiv$ $\dfrac{\log_2 N}{T_s} = \dfrac{\sum_i \log K_i}{T_s \log 2}$

Variable-step quantization

Theorem: The state of the closed-loop system will remain bounded provided that

$$K_i \geq e^{\lambda_i T_s}$$

$$\ell_i(k+1) = \frac{e^{\lambda_i T_s}}{K_i}\ell_i(k) + \eta_i + \delta_i$$

$$|x_i(0)| \leq \ell_i(0) - \eta_i$$

η_i, $\delta_i \equiv$ constants that depend on *upper bounds* on the noise/disturbance

bit allocation…

λ_i large $\Rightarrow e^{\lambda_i T_s}$ large $\Rightarrow K_i$ large $\Rightarrow$ many bits needed for ith eigenspace

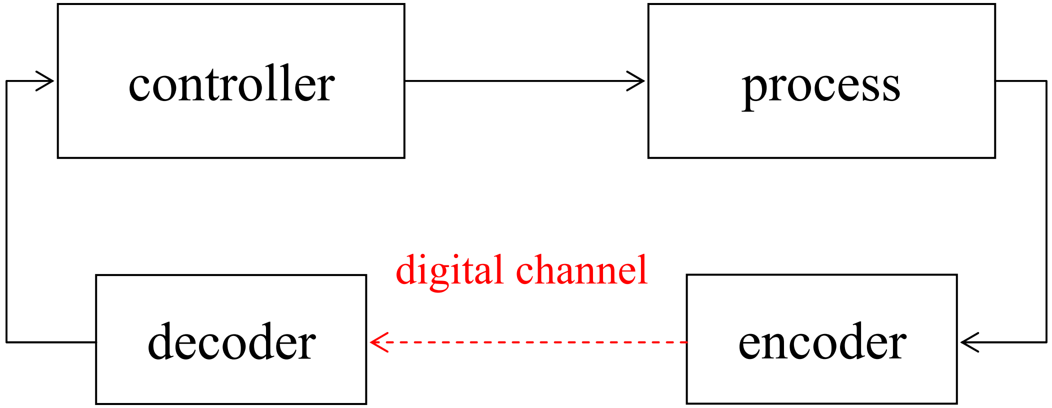
required bit-rate…

$$\text{bit-rate} \geq \frac{1}{\log 2}\sum_{\lambda_i>0} =: r_{\min}$$

} minimum rate can be achieved !!!

Conclusions

UCSB



There exists a minimum rate below which stabilization is not possible

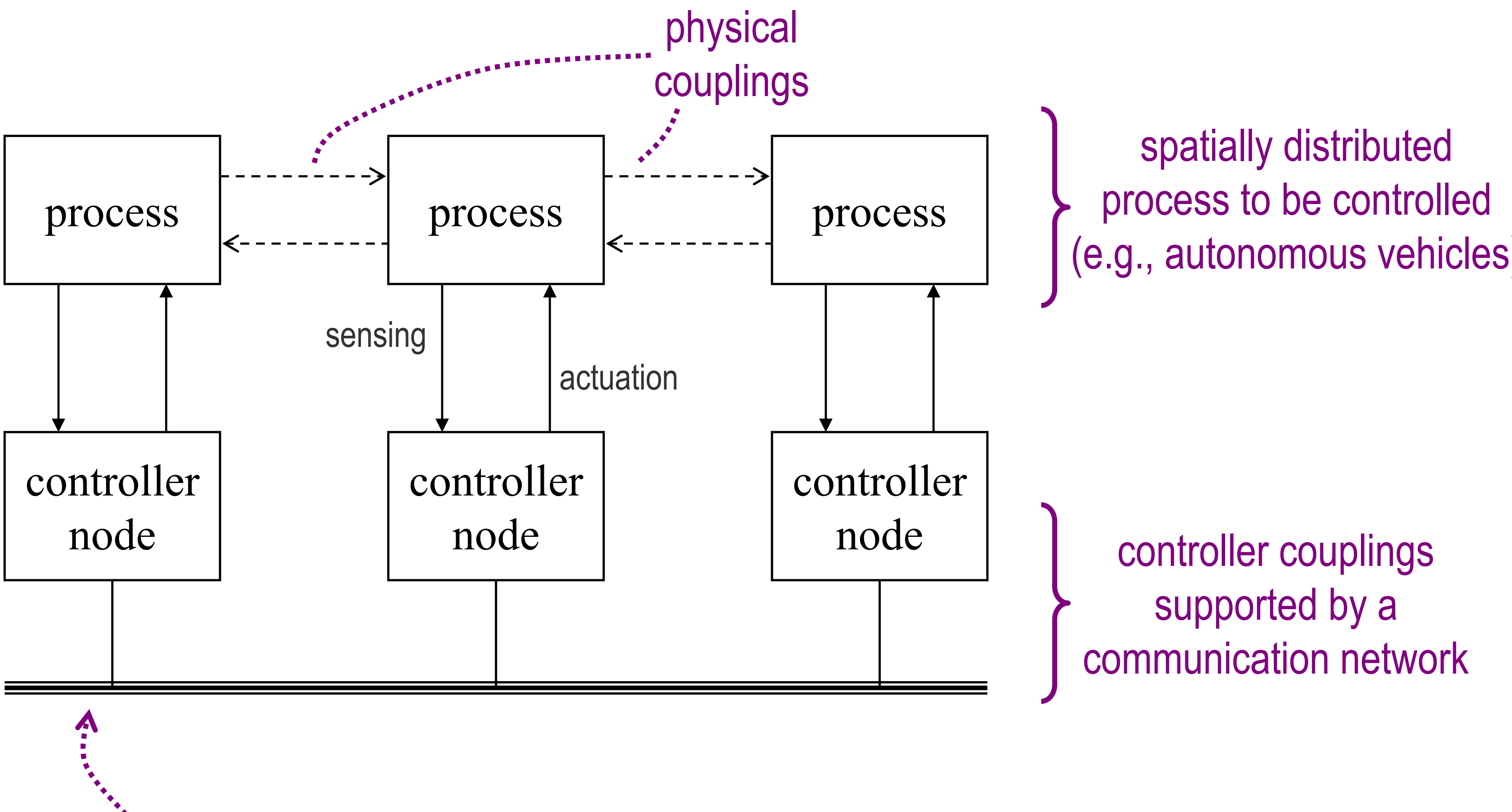
We proposed encoder/decoder pairs (inspired by DPCM) that can achieve rates arbitrarily close to the minimal

Variable-step quantization allows one to achieve the minimum bit rate

Performance/robustness vs. bit-rate tradeoffs are still poorly understood

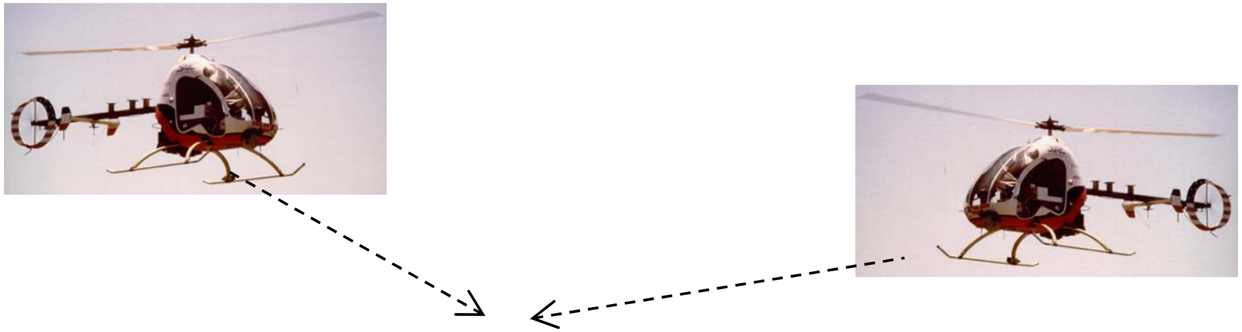
Need to investigate problem in stochastic setting (Will entropy-like coding work with lower rates?)

Distributed Control

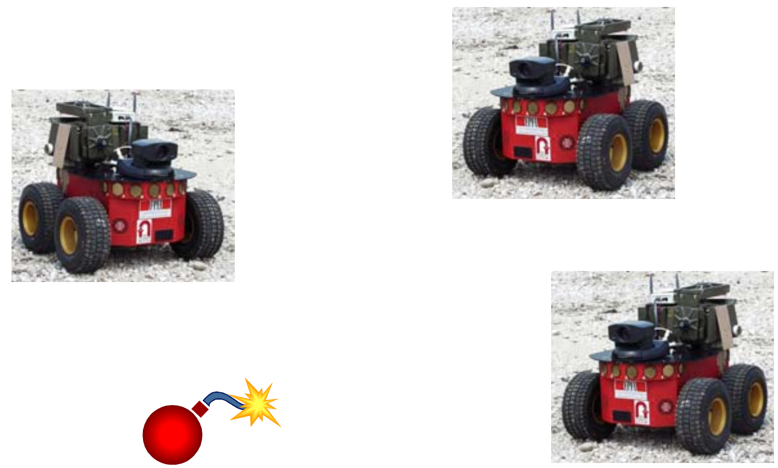


- minimize controller communication (stealth, bandwidth)
- study the effect of nonideal communication (delays, drops, blackouts)

Scenarios

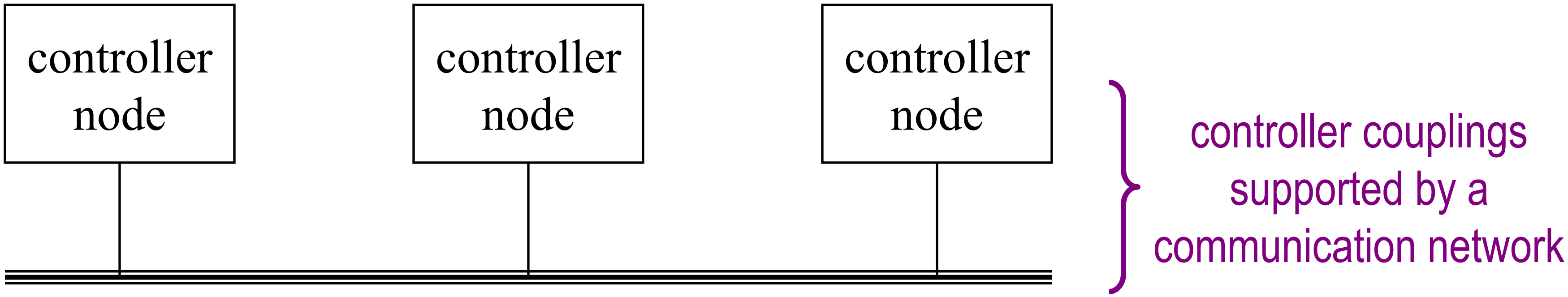


Rendezvous in minimum-time
or using minimum-energy
(in spite of disturbances)



Group of autonomous agents
cooperate in searching for a target
(perhaps mobile—search & pursuit)

Communication minimization



The "every bit-counts" paradigm…

previous case

Goal: Design each controller to minimize the number of *bits/second* that need to be exchanged between nodes (quantization, compression, …)

Domain: Media with little capacity and low-overhead protocols (bit at-a-time)
E.g., underwater acoustic comm. between a small number of nodes.

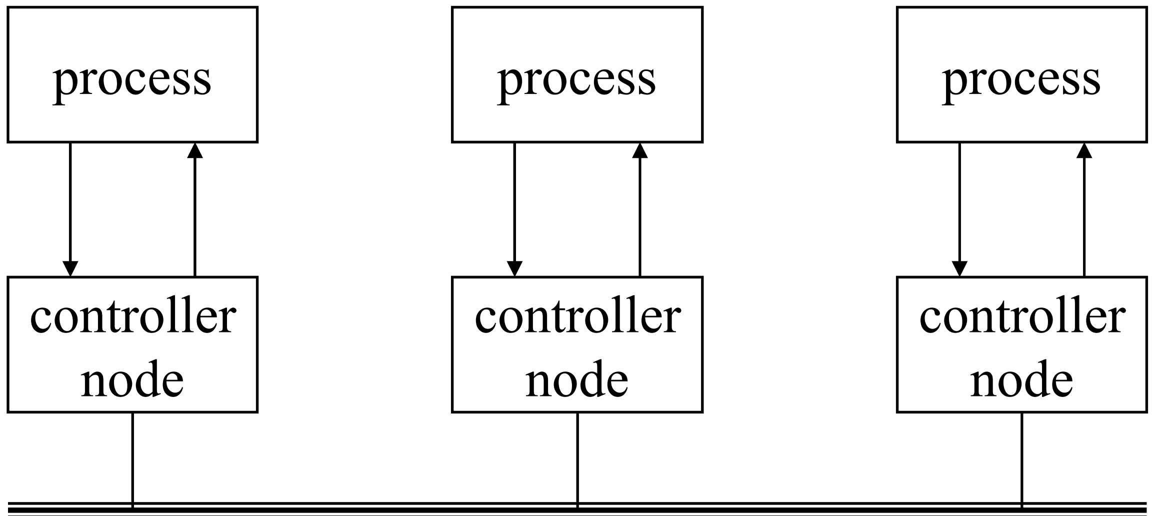
The "cost-per-message" paradigm…

current focus

Goal: Design each controller to minimize the number of *message exchanges* between nodes (scheduling, estimation, …)

Domain: Media shared by a large number of nodes with nontrivial media access control (MAC) protocol (packet at-a-time)
E.g., 802.11 wireless comm. between a large number of nodes.

Prototype problem



In this talk:

★ *decoupled* linear processes (with stochastic disturbance *d*)

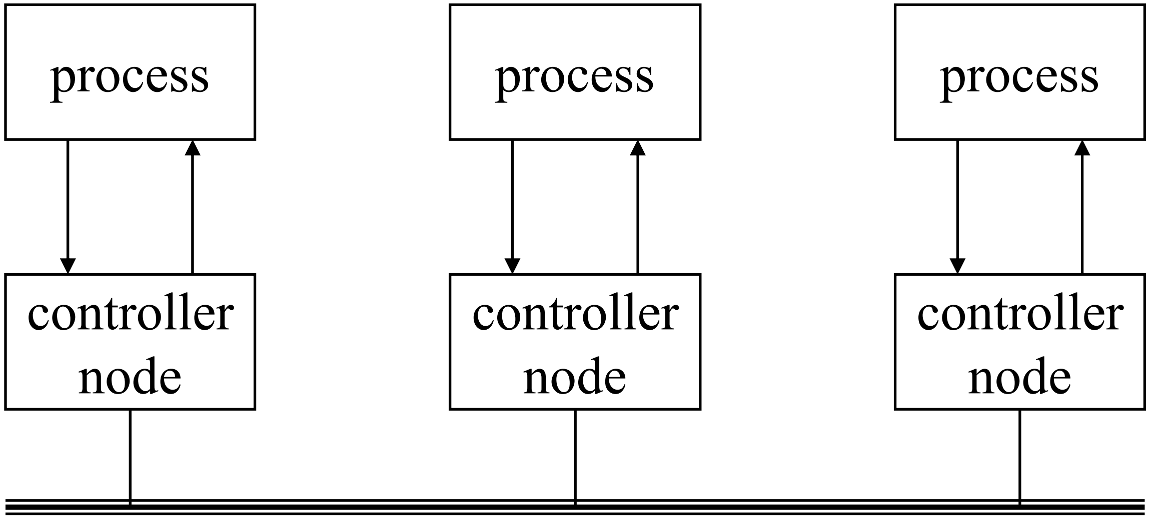
$$x^{+} = \begin{bmatrix} \star & 0 & 0 \\ 0 & \star & 0 \\ 0 & 0 & \star \end{bmatrix} x + \begin{bmatrix} \star & 0 & 0 \\ 0 & \star & 0 \\ 0 & 0 & \star \end{bmatrix} u + d$$

★ *coupled* quadratic control objective

$$\sum_{k=0}^{\infty} x' \begin{bmatrix} \star & * & 0 \\ * & \star & * \\ 0 & * & \star \end{bmatrix} x + \|u\|^2$$

notation: $x^{+} \equiv x(k{+}1)$

Prototype problem



In this talk:

★ *decoupled* linear processes (with stochastic disturbance d)

★ *coupled* quadratic control objective

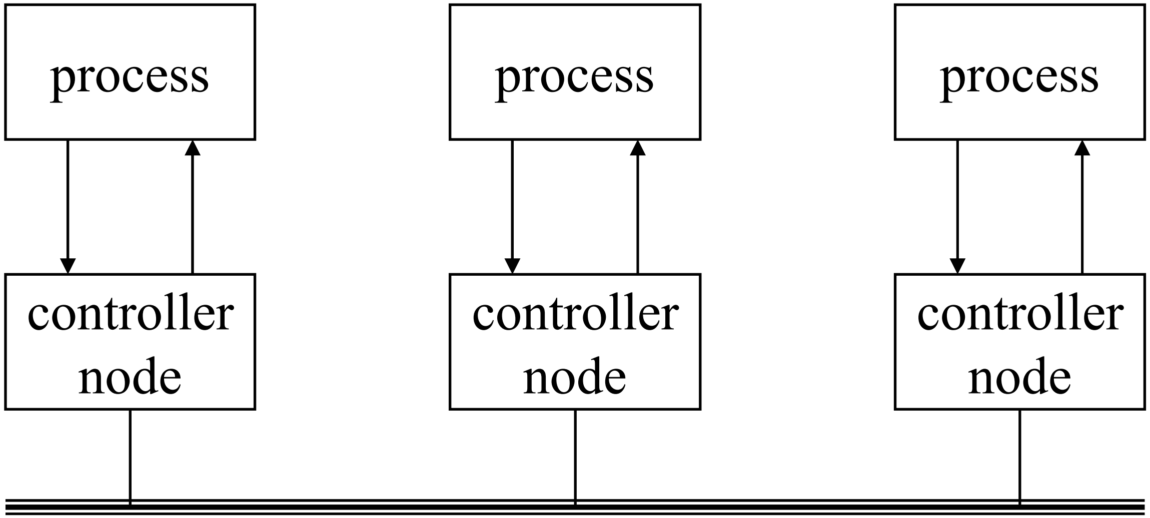
$$x^+ = \begin{bmatrix} A_1 & 0 \\ 0 & A_2 \end{bmatrix} x + \begin{bmatrix} B_1 & 0 \\ 0 & B_2 \end{bmatrix} u + d$$

$$\sum_{k=0}^{\infty} x' \begin{bmatrix} C_1'C_1 & -C_1'C_2 \\ -C_2'C_1 & C_2'C_2 \end{bmatrix} x + \|u\|^2$$

E.g., rendez-vous of two vehicles

notation: $x^+ \equiv x(k+1)$

Prototype problem



In this talk:

★ *decoupled* linear processes
(with stochastic disturbance *d*)

$$x^+ = \begin{bmatrix} \star & 0 & 0 \\ 0 & \star & 0 \\ 0 & 0 & \star \end{bmatrix} x + \begin{bmatrix} \star & 0 & 0 \\ 0 & \star & 0 \\ 0 & 0 & \star \end{bmatrix} u + d$$

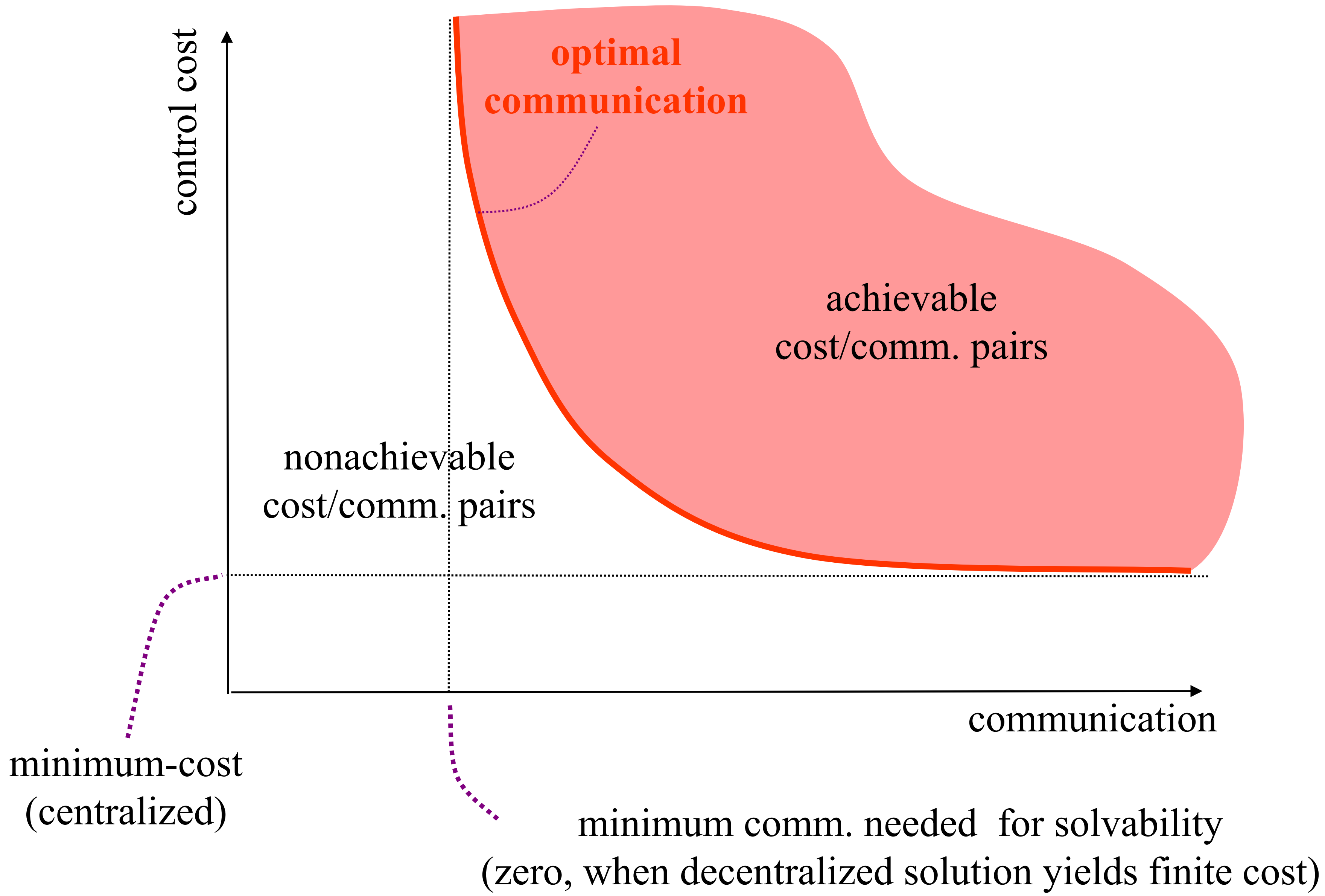
Minimum-cost solution (centralized)

$$u = \begin{bmatrix} \star & * & * \\ * & \star & * \\ * & * & \star \end{bmatrix} x$$

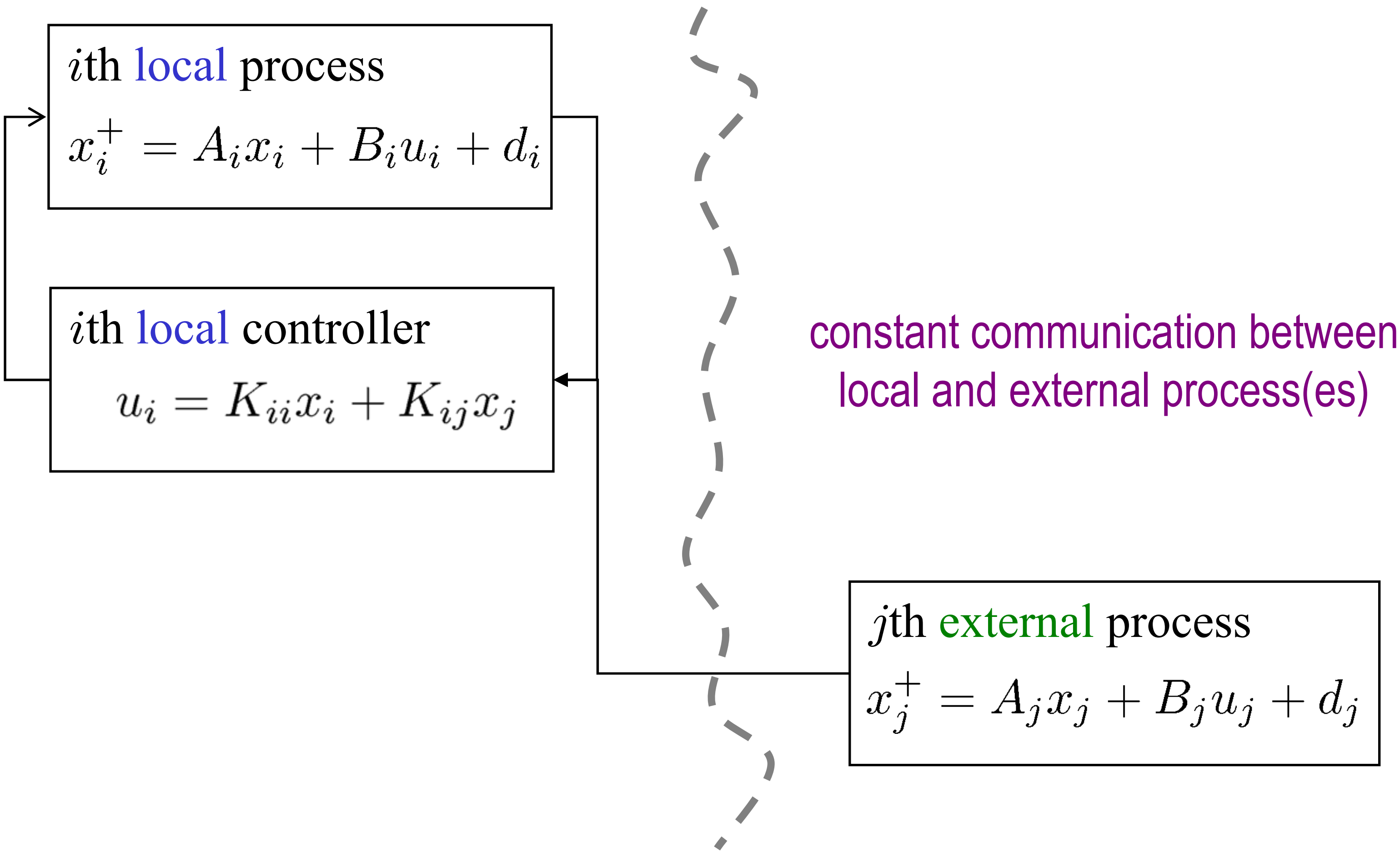
Completely decentralized solution

$$u = \begin{bmatrix} \star & 0 & 0 \\ 0 & \star & 0 \\ 0 & 0 & \star \end{bmatrix} x$$

Communication performance trade-off



Centralized architecture



Closed-loop system

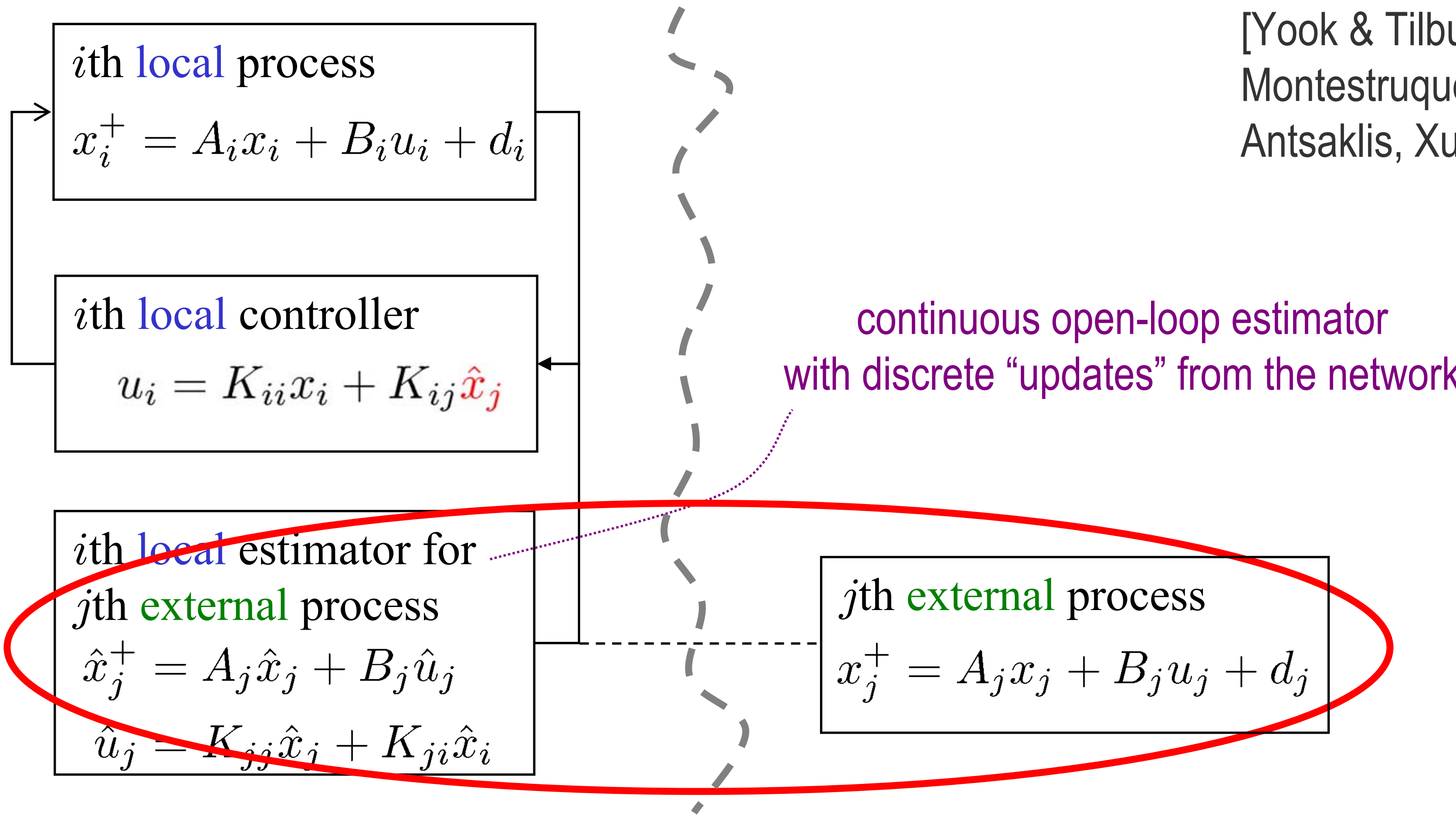
$$x_i^+ = (A_i + B_i K_{ii}) x_i + B_i K_{ij} x_j + d_i$$

$$x_j^+ = (A_j + B_j K_{jj}) x_j + B_j K_{ji} x_i + d_j$$

for simplicity here we assume only *two* processes

Estimator-based distributed architecture

[Yook & Tilbury, Montestruque & Antsaklis, Xu & JH]



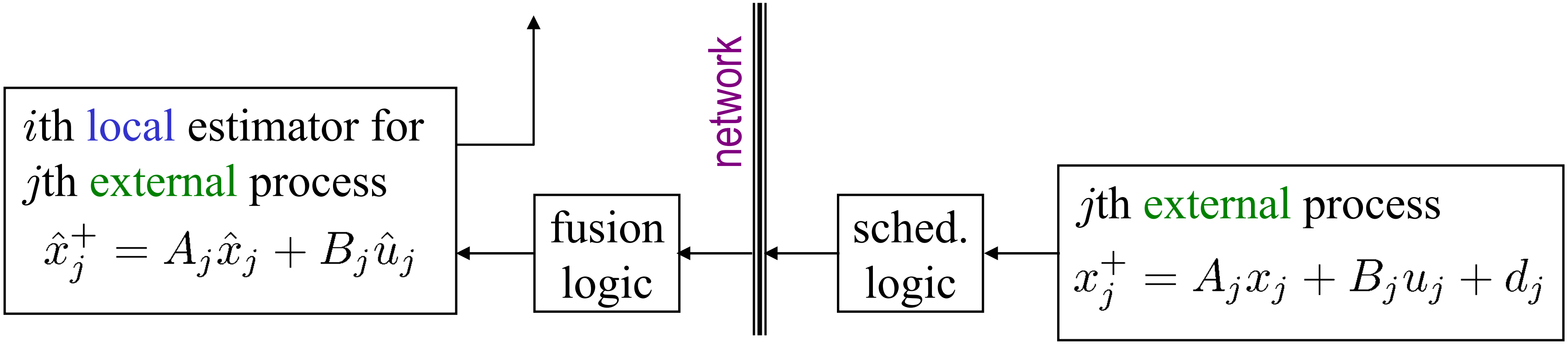
Closed-loop system

$$x_i^+ = (A_i + B_i K_{ii}) x_i + B_i K_{ij} x_j + d_i - B_i K_{ij} e_j \qquad e_i := x_i - \hat{x}_i$$

$$x_j^+ = (A_j + B_j K_{jj}) x_j + B_j K_{ji} x_i + d_j - B_j K_{ji} e_i \qquad e_j := x_j - \hat{x}_j$$

additive perturbation w.r.t centralized equations

Communication logic



How to fuse data? *When to send data?*

With no noise & delay (for now…)

$x_j(k)$ received from network at time k

$$\Downarrow$$

$$\hat{x}_j(k+1) = A_j x_j(k) + B_j u_j(k)$$

"best-estimate" based on data received

$$\Downarrow$$

$$e_j(k+1) = -d_j(k)$$

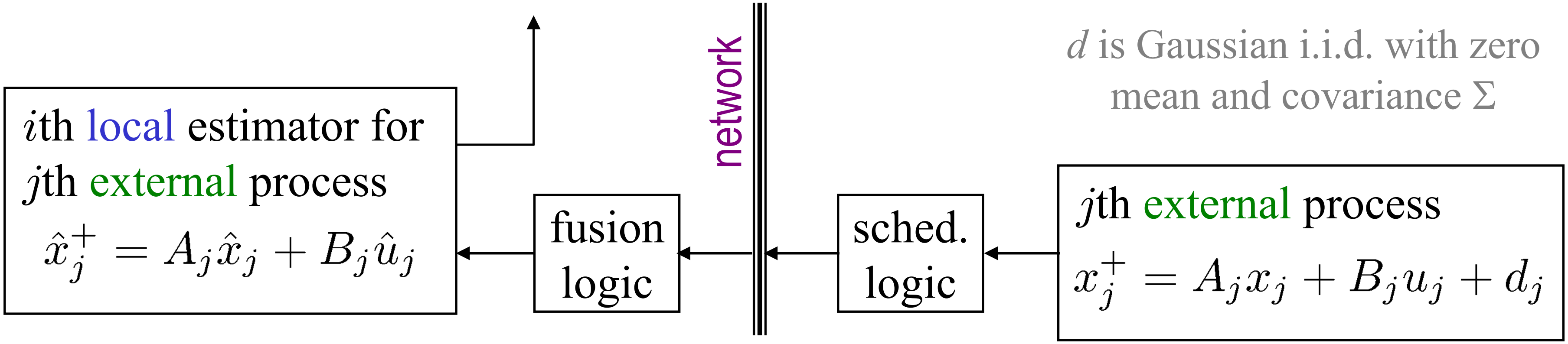
Scheduling logic *action*

$$a_j(k) = \begin{cases} 1 & x_j(k) \text{ sent at time } k \\ 0 & \text{no transmission at } k \end{cases}$$

Options…

- periodically
  $a_j(k) = \{k \text{ divisible by } T?\}, \quad T \in \mathbb{N}$
- feedback policy
  $a_j(k) = F(x_j(k), \ldots)$
- "optimal" …

Optimal Scheduling Logic



Goals:

minimize the estimation error $\Rightarrow$ minimize cost-penalty w.r.t. centralized

minimize the number of transitions $\Rightarrow$ minimize communication bandwidth

$$\limsup_{T\to\infty} \mathrm{E}\left[\frac{1}{T}\sum_{k=0}^{T-1} e_j(k)' Q e_j(k)\right]$$

average L-2 norm

$$\limsup_{T\to\infty} \mathrm{E}\left[\frac{1}{T}\sum_{k=0}^{T-1} a_j(k)\right]$$

average transmission rate

$$\min_{a(k)} \limsup_{T\to\infty} \mathrm{E}\left[\frac{1}{T}\sum_{k=0}^{T-1} e_j(k)' Q e_j(k) + \lambda\, a_j(k)\right]$$

relative weight of two criteria (will lead to Pareto-optimal solution)

Dynamic Programming solution

$$\min_{a(k)} \limsup_{T\to\infty} \mathrm{E}\left[\frac{1}{T}\sum_{k=0}^{T-1} e(k)'Qe(k) + \lambda\, a(k)\right]$$

undiscounted
average-cost problem

Theorem

$$(TV)(e) := \min_{a} \mathrm{E}\left[{e^+}'Qe^+ + \lambda\, a + V(e^+) \mid e\right]$$

dynamic programming
(DP) operator

1. There exists $J^* \in \mathbb{R}$ and bounded $h^* : \mathbb{R}^n \to \mathbb{R}$ such that

$$h^*(0) = 0, \qquad h^* + J = Th^*$$

2. J^* is the optimal cost and is achieved by the (*deterministic) static policy*

$$a(k) = \pi^*(e(k)), \quad \pi^*(e) := \begin{cases} 1 & \mathrm{E}[h^*(Ae+d)] + e'A'QAe \geq \mathrm{E}[h^*(d)] + \lambda \\ 0 & \text{otherwise} \end{cases}$$

3. h can be found by *value iteration*

$$h_{i+1} = Th_i - (Th_i)(0) \xrightarrow[i\to\infty]{exp.} h^*$$

Dynamic Programming solution

UCSB

Proof outline:

1. *$e(k)$ is Markov and its transition distribution satisfies an Ergodic property* (requires a mild restriction on the set of admissible policies omitted here)
2. T is a span-contraction [Hernandez-Lerma 96]
3. Result follows using standard arguments based on Banach's Fixed-Point Theorem for semi-norms.

Theorem

$$(TV)(e) := \min_a \mathrm{E}\left[e^{+\prime} Q e^{+} + \lambda\, a + V(e^{+}) \;\middle|\; e\right]$$

dynamic programming (DP) operator

1. There exists $J^* \in \mathbb{R}$ and bounded $h^* : \mathbb{R}^n \to \mathbb{R}$ such that

$$h^*(0) = 0, \qquad h^* + J = Th^*$$

2. J^* is the optimal cost and is achieved by the *(deterministic) static policy*

$$a(k) = \pi^*\big(e(k)\big), \quad \pi^*(e) := \begin{cases} 1 & \mathrm{E}[h^*(Ae+d)] + e'A'QAe \geq \mathrm{E}[h^*(d)] + \lambda \\ 0 & \text{otherwise} \end{cases}$$

3. h can be found by *value iteration*

$$h_{i+1} = Th_i - (Th_i)(0) \xrightarrow[i\to\infty]{exp.} h^*$$

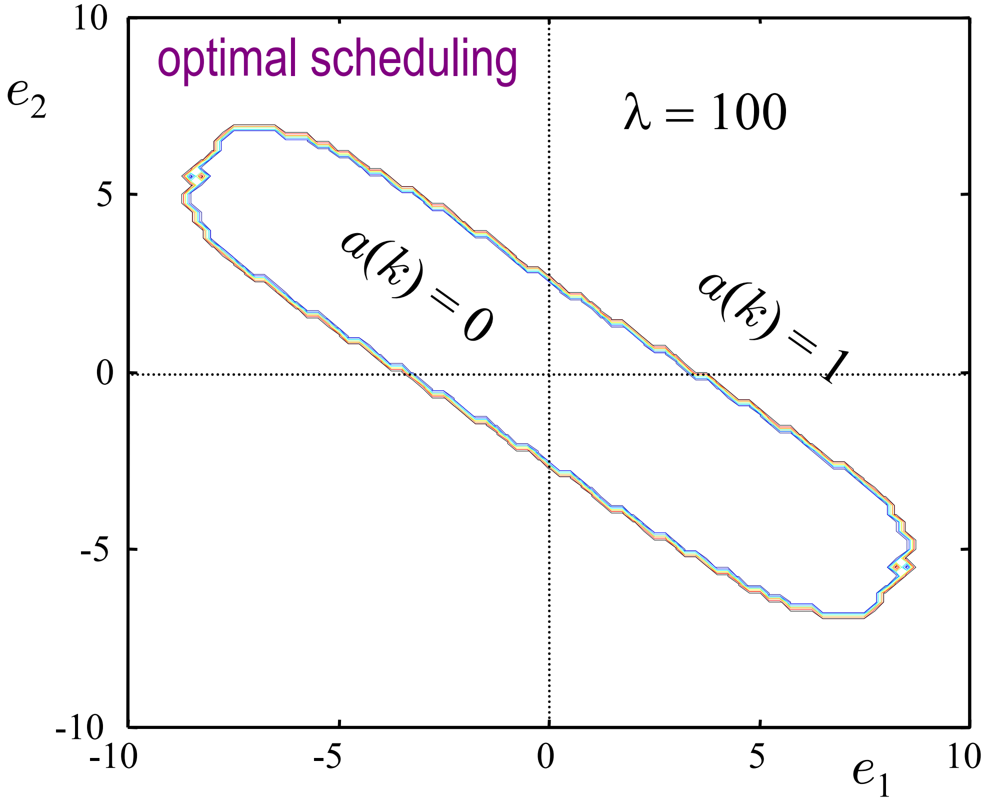
Example (2-dim)

$$\limsup_{T\to\infty} \mathrm{E}\left[\frac{1}{T}\sum_{k=0}^{T-1} e(k)' \begin{bmatrix} 4 & 0 \\ 0 & 1 \end{bmatrix} e(k) + \lambda\, a(k)\right]$$

local process

$$A := A_j + B_j K_j = \begin{bmatrix} 1 & 1 \\ .1 & .9 \end{bmatrix}$$

$$\mathrm{E}\left[d(k)d(k)'\right] = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$$



not ellipses!

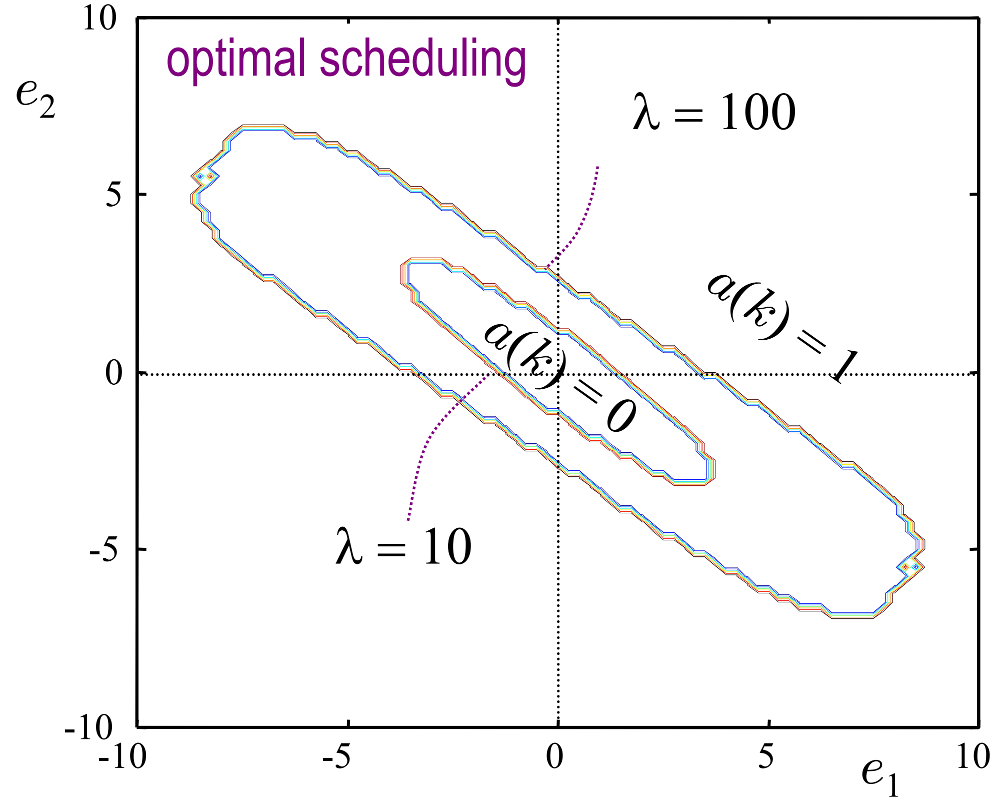
Example (2-dim)

$$\limsup_{T\to\infty} \mathrm{E}\left[\frac{1}{T}\sum_{k=0}^{T-1} e(k)' \begin{bmatrix} 4 & 0 \\ 0 & 1 \end{bmatrix} e(k) + \lambda\, a(k)\right]$$

local process

$$A := A_j + B_j K_j = \begin{bmatrix} 1 & 1 \\ .1 & .9 \end{bmatrix}$$

$$\mathrm{E}\left[d(k)d(k)'\right] = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$$



large weight in comm. cost

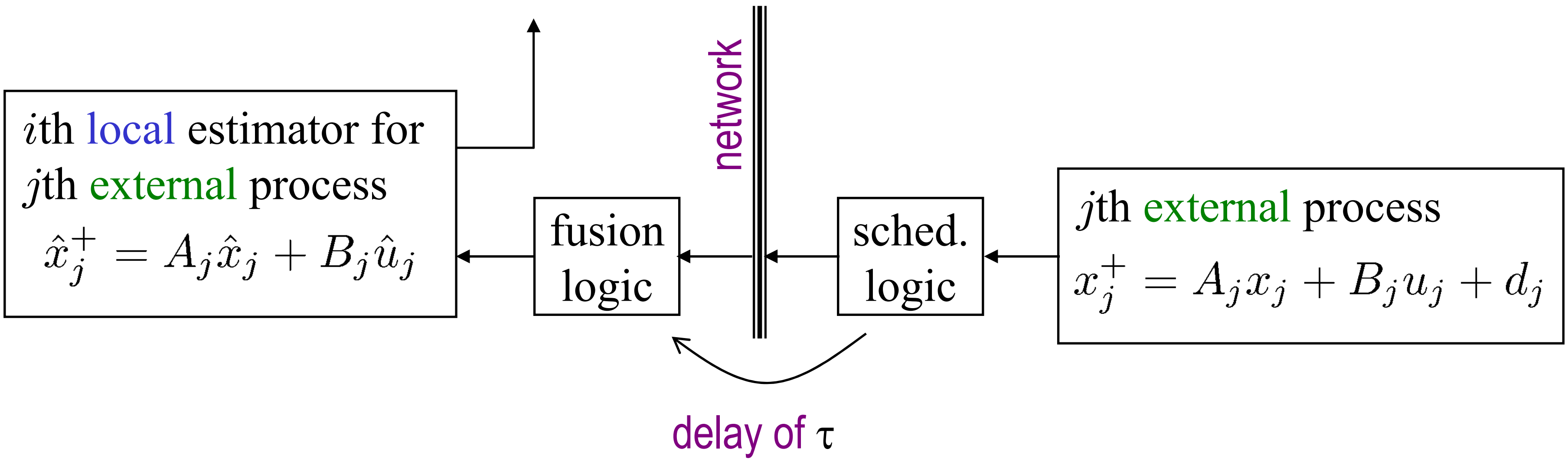
⇓

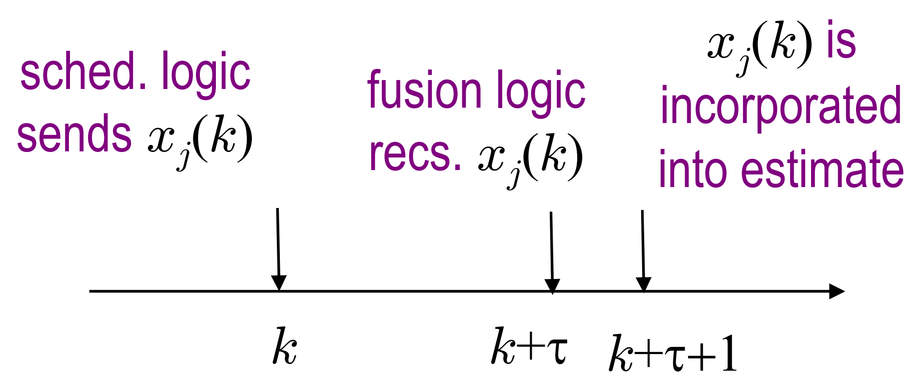
large error threshold

⇓

only communicate
when error is very large

Communication with latency





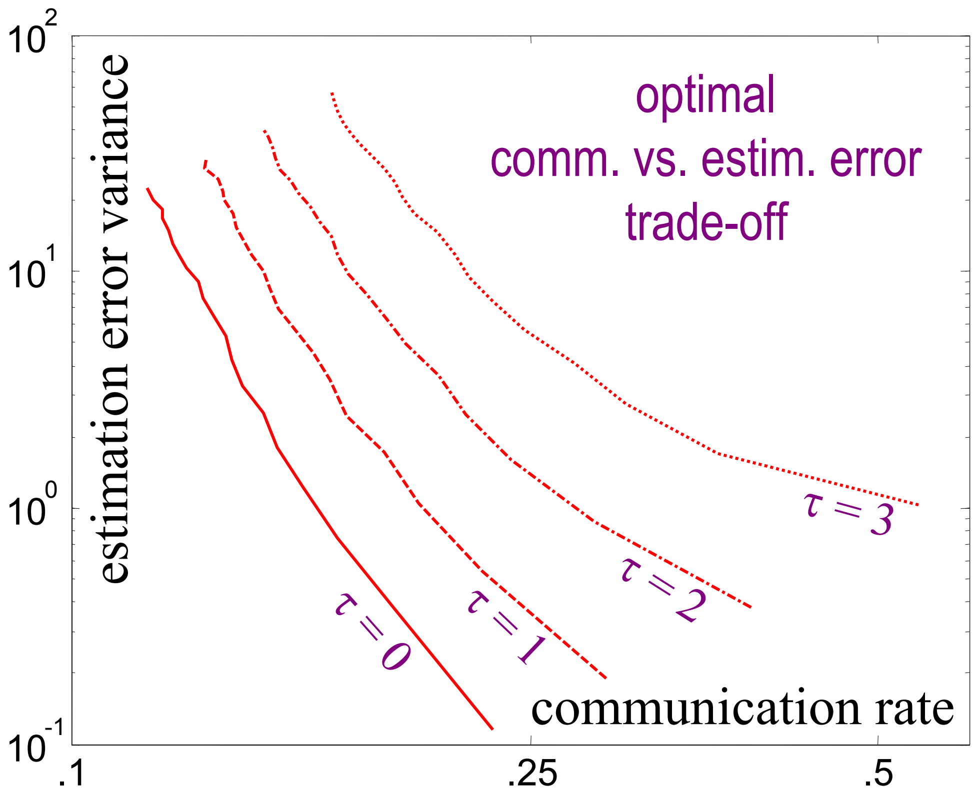
$\tau = 0$ in previous case

Example (1-dim)

$$\limsup_{T\to\infty} \mathrm{E}\left[\frac{1}{T}\sum_{k=0}^{T-1} e(k)^2 + \lambda\, a(k)\right]$$

$$A := A_j + B_j K_j = 2$$

$$\mathrm{E}[d(k)^2] = .01, \ \forall k$$



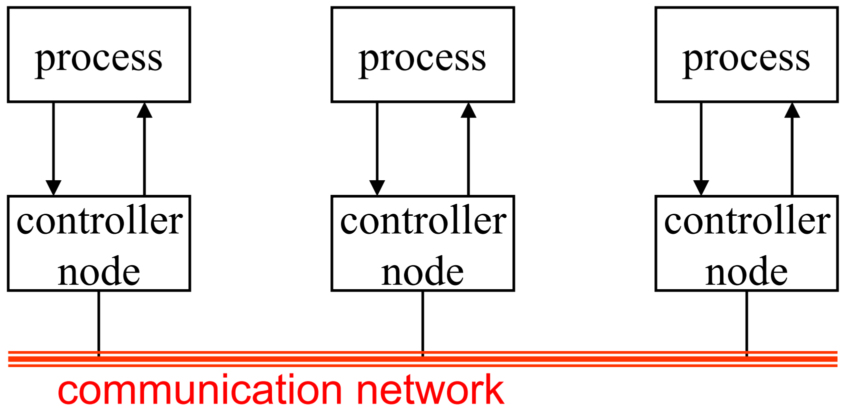
optimal scheduling

$$a(k) = \begin{cases} 1 & |\bar{e}(k)| \geq e_{\text{threshold}} \\ 0 & \text{otherwise} \end{cases}$$

estimation error, given all information enroute

with network latency
same error variance
requires more bandwidth

Conclusions



We constructed communications logics that minimize communication (measured in messages sending rate)

We considered networks with (fixed) latency

Study the effect of *packet losses* (especially important in wireless networks)

Coupled control/communication-logic design

Nonlinear processes

Papers available at http://www.ece.ucsb.edu/~hespanha