CDS + CMS + Networks

A CDS+CMS perspective on recent results in distributed control

Nikolai Matni (nmatni@caltech.edu)
http://www.cds.caltech.edu/~nmatni

CDS@20 2014, Pasadena

August 4, 2014

What makes a problem "easy"?

In optimization and control, we strive for

Computational Tractability

and

Scalability

What makes a problem “easy”?

In optimization and control, we look for

Convexity

and

Reasonable (Sub) Problem Sizes

What makes a problem “easy”?

In optimization and control, we look for

Convexity

and

Reasonable (Sub) Problem Sizes

Reasonably Sized Implementations

What makes a problem "easy"?

Different Flavors of Convexity

- Linear Programs (LPs)
- Second Order Cone Programs (SOCPs)
- Semi-definite Programs (SDPs)

Different Reasonable Problem Sizes

- LPs: Millions of variables
- SOCPs: Thousands of variables
- SDPs: Hundreds of variables

What makes a problem "easy"?

Different Flavors of Convexity

- Linear Programs (LPs)
- Second Order Cone Programs (SOCPs)
- Semi-definite Programs (SDPs)

↓ Expressivity

Different Reasonable Problem Sizes

- LPs: Millions of variables
- SOCPs: Thousands of variables
- SDPs: Hundreds of variables

What makes a problem "easy"?

Different Flavors of Convexity

- Linear Programs (LPs)
- Second Order Cone Programs (SOCPs)
- Semi-definite Programs (SDPs)

↓ Expressivity

Different Reasonable Problem Sizes

- LPs: Millions of variables
- SOCPs: Thousands of variables
- SDPs: Hundreds of variables

↑ Scalability

Application Areas that Need(ed) our Help

Optimal power flow (OPF)

- **Non-convex**, possibly **large** scale optimization

Software Defined Networking (SDN)

Active control of smart grid

Automated highway systems

- All **huge** scale
- All need real time **distributed** (optimal) **control**
- **Non-convex**

Application Areas that Need(ed) our Help

In general, these problems are non-convex and not scalable…

Use Structure to Relax

In general, these problems are non-convex and not scalable…

General

Hard problems

Main Theme of 1st Part:
Use Structure to Relax

Use Structure to Relax

In general, these problems are non-convex and not scalable…

General → Structured

takes

Hard problems → Easy problems

Main Theme of 1st Part:
Use Structure to Relax

Roadmap for 1st Part

DC OPF

- Connections to positive systems
- Connections to Sum of Squares Programming & Polynomial Optimization

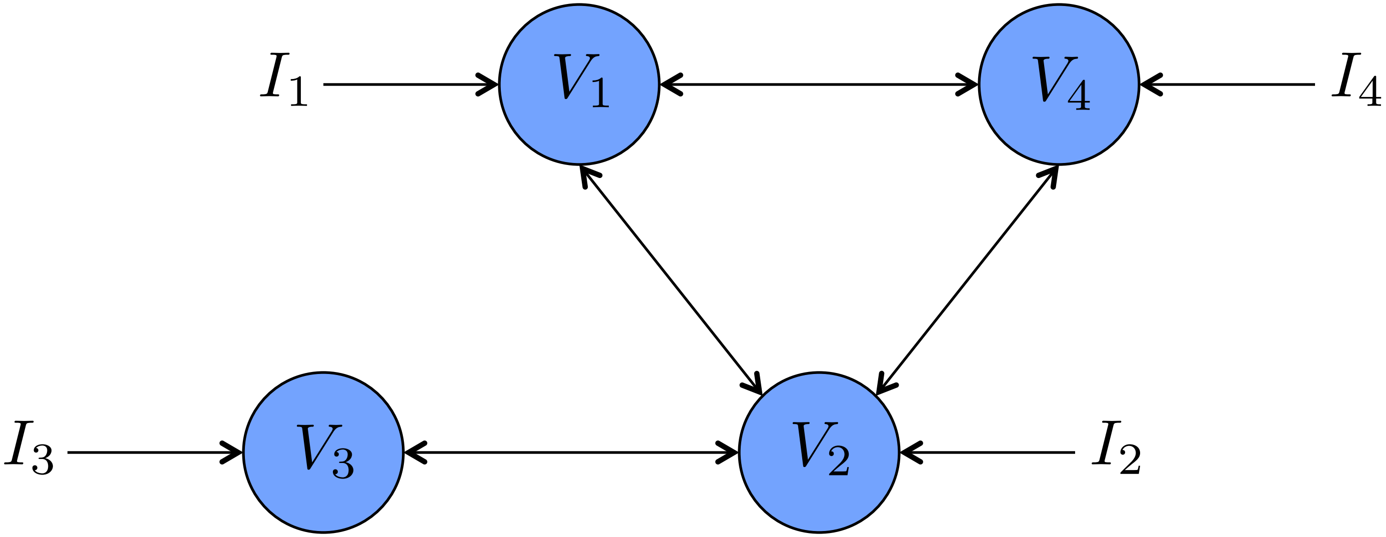
Distributed Optimal Control

- Why it’s hard: Witsenhausen
- How can we make it tractable: Quadratic Invariance
- How can we make it scalable: Localizable Systems

Setup for 2nd Part

Break

Case Study: DC OPF



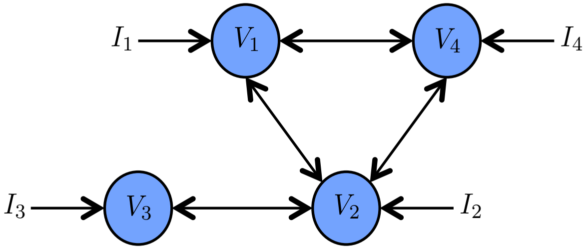
Kirchoff gives

$$\begin{bmatrix} I_1 \\ I_2 \\ I_3 \\ I_4 \end{bmatrix} = \begin{bmatrix} y_{12}+y_{14} & -y_{12} & 0 & -y_{14} \\ -y_{21} & y_{21}+y_{23}+y_{24} & -y_{23} & -y_{24} \\ 0 & -y_{32} & y_{32} & 0 \\ -y_{41} & -y_{42} & 0 & y_{41}+y_{42} \end{bmatrix} \begin{bmatrix} V_1 \\ V_2 \\ V_3 \\ V_4 \end{bmatrix}$$

Power Flow Optimization Using Positive Quadratic Optimization, by Lavaei, Rantzer and Low, 2011

Case Study: DC OPF

The DC OPF problem is



$$\begin{array}{lll} \underset{I_j, V_j}{\text{minimize}} & \sum_{j=1}^{N} I_j V_j & \\ \text{subject to} & I = YV & \text{(a)} \\ & V_k I_k \leq P_k, \; V_k^{\min} \leq V_k \leq V_k^{\max} & \text{(b)} \\ & y_{jk}(V_k - V_j)^2 \leq L_{jk} & \text{(c)} \\ & \text{for all } j, \, k = 1, \ldots, N & \end{array}$$

(a) Kirchoff's law
(b) Node power and voltage constraints
(c) Line constraints

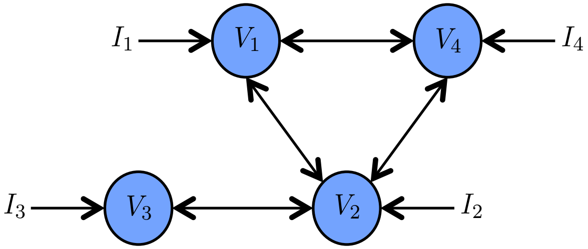
Indefinite Quadratic Objectives and Constraints → **Non-Convex**

Power Flow Optimization Using Positive Quadratic Optimization, by Lavaei, Rantzer and Low, 2011

Case Study: DC OPF

The DC OPF problem is of the form

$$\begin{aligned} \underset{x}{\text{maximize}} \quad & x^\top M_0 x \\ \text{subject to} \quad & x^\top M_k x \geq b_k \\ & \text{for } k = 1, \ldots, K \end{aligned}$$



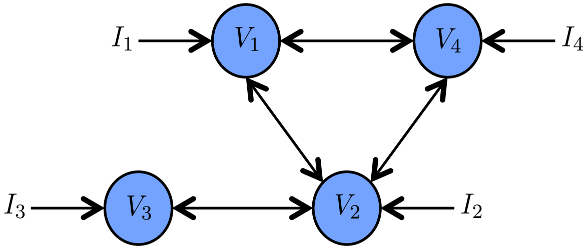
Indefinite Quadratic Objectives and Constraints → **Non-Convex**

Power Flow Optimization Using Positive Quadratic Optimization, by Lavaei, Rantzer and Low, 2011

Case Study: DC OPF

The DC OPF problem is of the form

$$\begin{aligned} \underset{x}{\text{maximize}} \quad & x^\top M_0 x \\ \text{subject to} \quad & x^\top M_k x \geq b_k \\ & \text{for } k = 1, \ldots, K \end{aligned}$$



Indefinite Quadratic Objectives and Constraints → **Non-Convex**

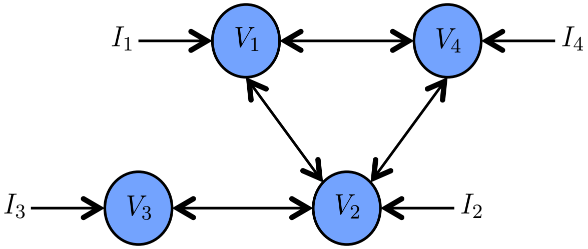
In general, NP-Hard

Power Flow Optimization Using Positive Quadratic Optimization, by Lavaei, Rantzer and Low, 2011

Case Study: DC OPF

The DC OPF problem is of the form

$$\begin{aligned} \underset{x}{\text{maximize}} \quad & x^\top M_0 x \\ \text{subject to} \quad & x^\top M_k x \geq b_k \\ & \text{for } k = 1, \ldots, K \end{aligned}$$



Indefinite Quadratic Objectives and Constraints → **Non-Convex**

In general, NP-Hard

A little bit of algebra shows that the *M_k* are Metzler

This case is NOT general

Power Flow Optimization Using Positive Quadratic Optimization, by Lavaei, Rantzer and Low, 2011

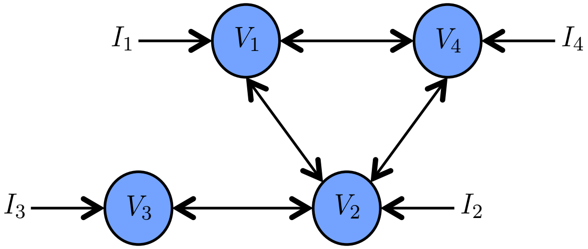
Case Study: DC OPF

The DC OPF problem is of the form

$$\begin{aligned} \underset{x}{\text{maximize}} \quad & x^\top M_0 x \\ \text{subject to} \quad & x^\top M_k x \geq b_k \\ & \text{for } k = 1, \ldots, K \end{aligned}$$

$$\Updownarrow$$

$$\begin{aligned} \underset{X \succeq 0}{\text{maximize}} \quad & \mathrm{Tr} M_0 X \\ \text{subject to} \quad & \mathrm{Tr} M_k X \geq b_k \\ & \text{for } k = 1, \ldots, K \\ & \mathrm{rank}(X) = 1 \end{aligned}$$



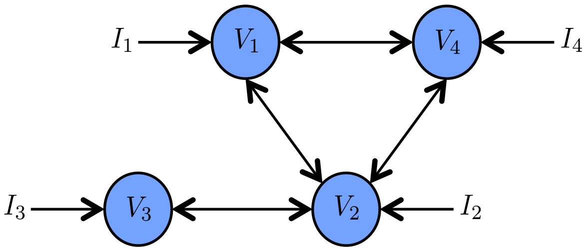
Still non-convex

Power Flow Optimization Using Positive Quadratic Optimization, by Lavaei, Rantzer and Low, 2011

Case Study: DC OPF

The DC OPF problem is of the form

$$\begin{array}{rl} \underset{x}{\text{maximize}} & x^\top M_0 x \\ \text{subject to} & x^\top M_k x \geq b_k \\ & \text{for } k = 1, \ldots, K \end{array}$$



$$\Updownarrow$$

$$\begin{array}{rl} \underset{X \succeq 0}{\text{maximize}} & \mathrm{Tr} M_0 X \\ \text{subject to} & \mathrm{Tr} M_k X \geq b_k \\ & \text{for } k = 1, \ldots, K \\ & \cancel{\mathrm{rank}(X) = 1} \end{array}$$

Convex!

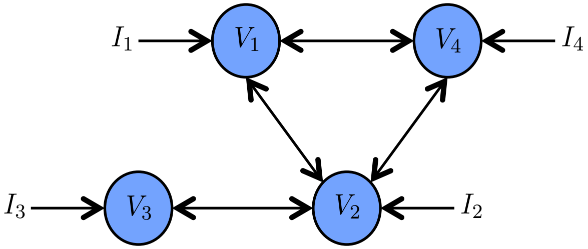
But are we solving the same problem?

Power Flow Optimization Using Positive Quadratic Optimization, by Lavaei, Rantzer and Low, 2011

Case Study: DC OPF

The DC OPF problem is of the form

$$\begin{array}{ll} \underset{X \succeq 0}{\text{maximize}} & \mathrm{Tr} M_0 X \\ \text{subject to} & \mathrm{Tr} M_k X \geq b_k \\ & \text{for } k = 1, \ldots, K \\ & \cancel{\mathrm{rank}(X) = 1} \end{array}$$



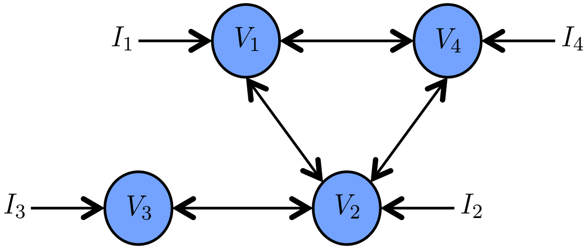
We are! Relaxation exact because of Metzler constraints

Power Flow Optimization Using Positive Quadratic Optimization, by Lavaei, Rantzer and Low, 2011

Case Study: DC OPF

The DC OPF problem is of the form

$$\begin{array}{ll} \underset{X \succeq 0}{\text{maximize}} & \operatorname{Tr} M_0 X \\ \text{subject to} & \operatorname{Tr} M_k X \geq b_k \\ & \text{for } k = 1, \ldots, K \\ & \cancel{\operatorname{rank}(X) = 1} \end{array}$$



We are! Relaxation exact because of Metzler constraints

Let $X = (x_{ij})$ be any positive semi-definite matrix satisfying constraints.

$$\begin{array}{rcl} x_{ii} & \geq & 0 \\ x_{ij} & \leq & \sqrt{x_{ii} x_{jj}} \end{array}$$

Let $x = (\sqrt{x_{ii}})$. Then $(xx^\top)_{ii} = X_{ii}$, but $(xx^\top)_{ij} = \sqrt{x_{ii} x_{jj}} \geq X_{ij}$. Then $x^\top M_k x \geq \operatorname{Tr} M_k X$ because M_k are Metzler.

Aside: Positive Systems Theory

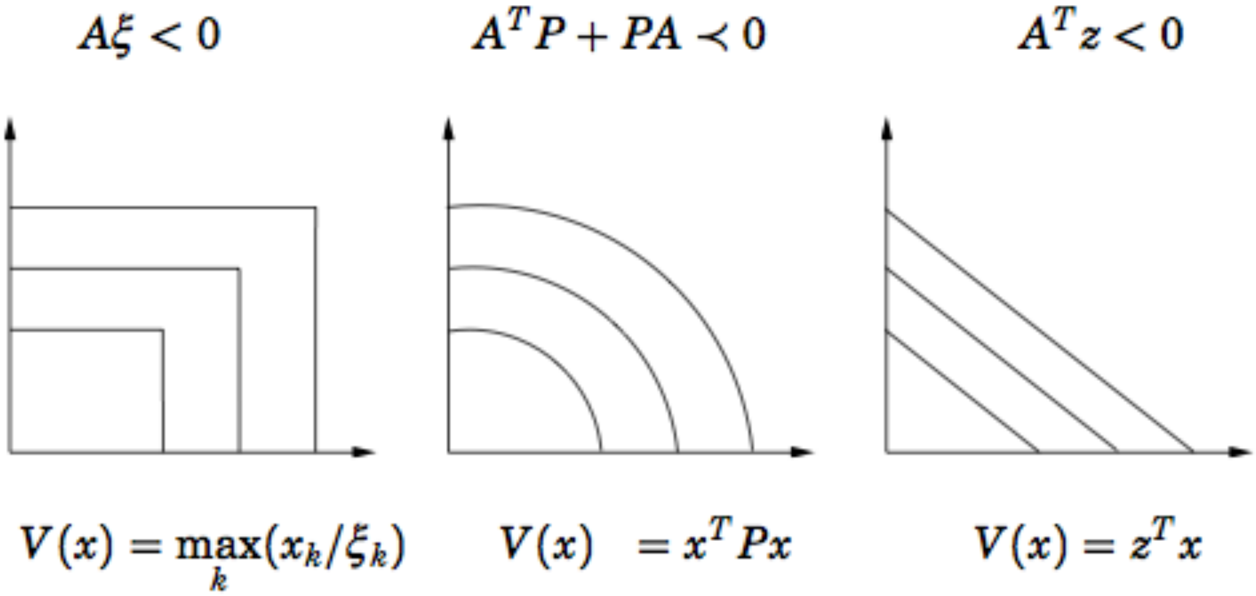
Dynamical system

$$\dot{x} = Ax$$

Suppose *A* is Metzler. Then:

$$x(0) \in \mathbb{R}_+ \implies x(t) \in \mathbb{R}_+ \, \forall t \geq 0$$

How does this help? Lyapunov/Storage functions can be linear!



Theory: Tanaka &Langbort, 2012, Rantzer 2012, 2013,
Biomed Applicatons: Jonsson, Matni & Murray, 2013, 2014, Jonsson, Rantzer & Murray 2013

Aside: Duality and Relaxations

Lagrangian of original problem:

$$\begin{aligned} L(x, \lambda_k) &= x^\top M_0 x + \textstyle\sum_{k=1}^{K} \lambda_k \left(x^\top M_k x - b_k\right) \\ &= -\textstyle\sum_{k=1}^{K} \lambda_k b_k + x^\top \left(M_0 + \textstyle\sum_{k=1}^{K} \lambda_k M_k\right) x \end{aligned}$$

Dual:

$$\begin{array}{ll} \underset{\lambda_k \geq 0}{\text{minimize}} & -\sum_{k=1}^{K} \lambda_k b_k \\ \text{subject to} & M_0 + \sum_{k=1}^{K} \lambda_k M_k \preceq 0 \end{array}$$

Dual of dual:

$$\begin{array}{ll} \underset{X \succeq 0}{\text{maximize}} & \mathrm{Tr} M_0 X \\ \text{subject to} & \mathrm{Tr} M_k X \geq b_k \\ & \text{for } k = 1, \ldots, K \end{array}$$

Aside: SOS Optimization

Polynomial optimization = polynomial non-negativity

$$\max p(x) = \min \gamma \text{ s.t. } \gamma - p(x) \geq 0$$

Problem: testing polynomial non-negativity NP-hard in general.

Solution: check weaker sufficient condition

$$\text{If } p(x) = \sum q(x)^2 \text{ then } p(x) \geq 0$$

Aside: SOS Optimization

Computational test for SOS is a semi-definite program.

For simplicity, fix *d=1*. Then

$$p(x) = \begin{bmatrix} 1 \\ x_1 \\ \vdots \\ x_n \end{bmatrix}^{\top} Q \begin{bmatrix} 1 \\ x_1 \\ \vdots \\ x_n \end{bmatrix} \text{ is SOS if and only if } Q \succeq 0$$

Coefficients of *p(x)* impose affine constraints on *Q*.

Aside: SOS Optimization

Constrained polynomial optimization

$$\max p(x) \text{ s.t. } g_i(x) \geq 0$$

Relax to

$$\min \gamma \text{ s.t. } \gamma - p(x) = s_0(x) + \textstyle\sum_i s_i(x) g_i(x)$$
$$s_0(x), s_i(x) \text{ are } SOS(2d)$$

Get smaller and smaller upper bounds by letting *d* increase and including more “polynomial Lagrange multipliers”.

So how does the DC OPF problem relate to this?

Aside: SOS Optimization

SOS relaxation of original problem:

$$\min \gamma \text{ s.t. } \gamma - x^\top M_0 x = s_0(x) + \textstyle\sum_k s_k(x) \left(x^\top M_k x - b_k\right)$$

$$s_k(x) = \begin{bmatrix} 1 \\ x \end{bmatrix}^\top Q_k \begin{bmatrix} 1 \\ x \end{bmatrix}, \; Q_k \succeq 0$$

Expand RHS and equate coefficients

$$\gamma = Q_0^{11} - \textstyle\sum_{k=1}^K Q_k^{11} b_k, \; Q_k^{1,j} = 0 \text{ for all } j \neq 1.$$

$$\text{For } k \geq 1, \; Q_k^{ij} = 0 \text{ for all } i, j \neq 1$$

$$-M_0 = Q_0^{2:n+1, 2:n+1} + \textstyle\sum_{k=1}^K Q_k^{11} M_k$$

Aside: SOS Optimization

SOS relaxation of original problem:

$$\begin{array}{ll} \underset{Q_k^{11} \geq 0, Q \succeq 0}{\text{minimize}} & Q_0^{11} - \sum_{k=1}^{K} Q_k^{11} b_k \\ \text{subject to} & \sum_{k=1}^{K} Q_k^{11} M_k + M_0 = -Q \end{array}$$

Aside: SOS Optimization

SOS relaxation of original problem:

$$\begin{array}{ll} \underset{Q_k^{11} \geq 0, Q \succeq 0}{\text{minimize}} & Q_0^{11} - \sum_{k=1}^{K} Q_k^{11} b_k \\ \text{subject to} & \sum_{k=1}^{K} Q_k^{11} M_k + M_0 = -Q \end{array}$$

⬇

$$\begin{array}{ll} \underset{Q_k^{11} \geq 0, Q \succeq 0}{\text{minimize}} & - \sum_{k=1}^{K} Q_k^{11} b_k \\ \text{subject to} & \sum_{k=1}^{K} Q_k^{11} M_k + M_0 \preceq 0 \end{array}$$

Aside: SOS Optimization

SOS relaxation of original problem:

$$\begin{array}{ll} \underset{Q_k^{11} \geq 0, Q \succeq 0}{\text{minimize}} & Q_0^{11} - \sum_{k=1}^{K} Q_k^{11} b_k \\ \text{subject to} & \sum_{k=1}^{K} Q_k^{11} M_k + M_0 = -Q \end{array}$$

$$\Downarrow$$

$$\begin{array}{ll} \underset{Q_k^{11} \geq 0, Q \succeq 0}{\text{minimize}} & -\sum_{k=1}^{K} Q_k^{11} b_k \\ \text{subject to} & \sum_{k=1}^{K} Q_k^{11} M_k + M_0 \preceq 0 \end{array}$$

This is the dual of our original problem!

Quadratic optimization with Metzler matrices is SOS(2) exact.

DC OPF: Summary

Optimal power flow (OPF)

- Convex Relaxations are exact for DC power flow
- Go see Steven Low's talk on Thursday for AC power and scalability

Solution from OPF problem provides reference trajectory for system to track.

Future smart grid will need active control

Large scale → Distributed Architecture

Roadmap for 1st Part

DC OPF

- Connections to positive systems
- Connections to Sum of Squares Programming & Polynomial Optimization

Distributed Optimal Control

- Why it’s hard: Witsenhausen
- How can we make it tractable: Quadratic Invariance
- How can we make it scalable: Localizable Systems

Setup for 2nd Part

Break

Distributed Control

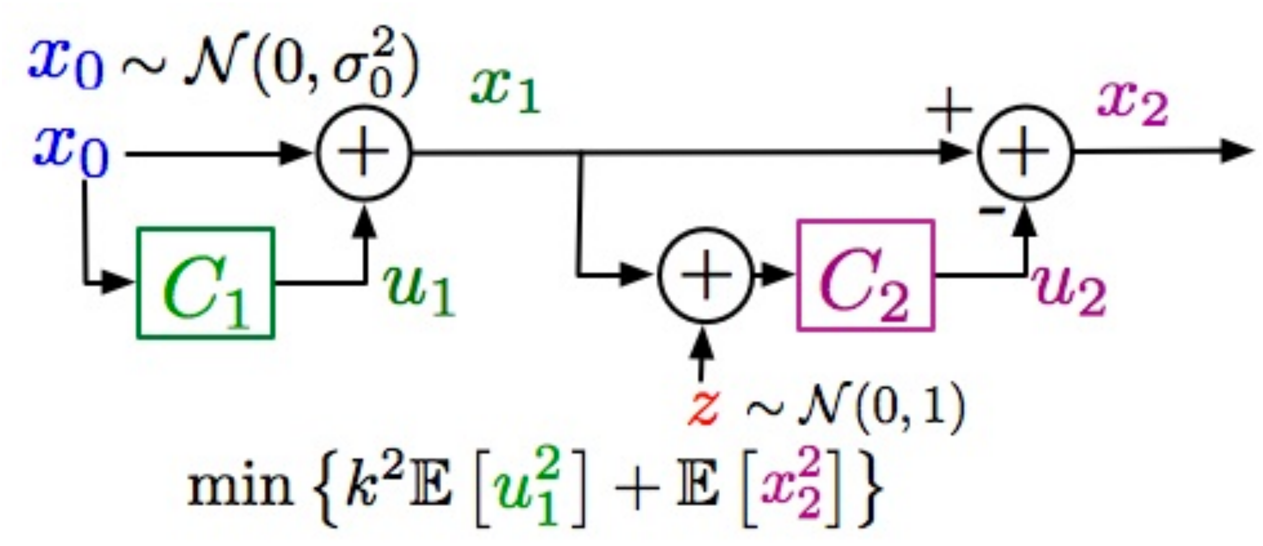
Large scale systems not amenable to centralized control

Idea: restrict information each controller has access to

Positives: control laws are local, and hence scalable to implement.

Negatives: in general non-convex. Witsenhausen.

Witsenhausen Counter-Example



Comms problem masquerading as a control problem

Roughly, C_1 needs to tell C_2 (via $x_1 = u_1 + x_0$) what x_0 was

- C_1's only goal is to *signal through the plant* as efficiently as possible
- Reliable communication through noisy channel → coding (i.e. non-linear)

Demystifying the Witsenhausen Counterexample, Grover & Sahai '10

Distributed Control

Witsenhausen shows that distributed control is **non-convex in general**

What **structure** do we need to regain convexity?

Witsenhausen hard because of comms aspect. Need to remove this incentive to signal.

Quadratic Invariance (Rotkowitz & Lall ‘06), Partial Nestedness (Ho & Chu ‘72), Funnel Causality (Bahmieh & Voulgaris ’03), Poset Causality (Shah & Parrilo ‘12)

Distributed Control

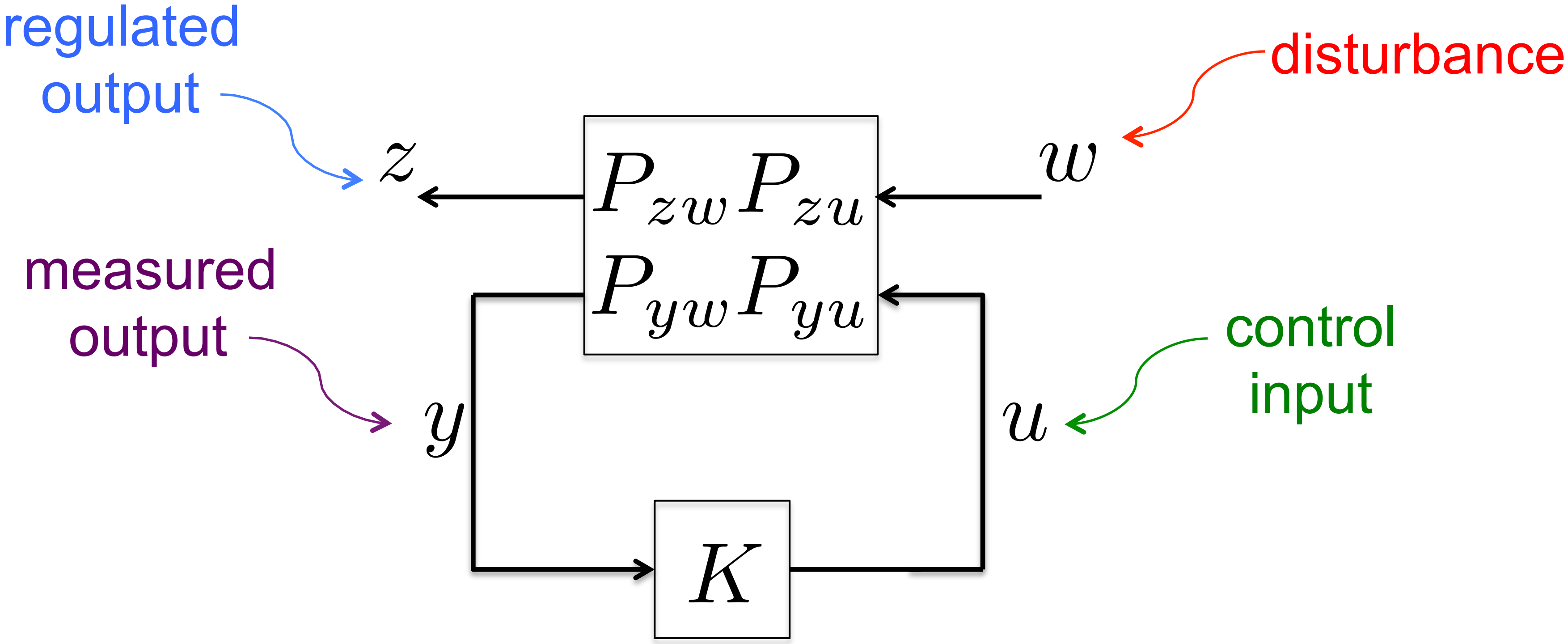
Witsenhausen shows that distributed control is **non-convex in general**

What **structure** do we need to regain convexity?

Witsenhausen hard because of comms aspect. Need to remove this incentive to signal.

Quadratic Invariance (Rotkowitz & Lall ‘06), Partial Nestedness (Ho & Chu ‘72), Funnel Causality (Bahmieh & Voulgaris ’03), Poset Causality (Shah & Parrilo ‘12)

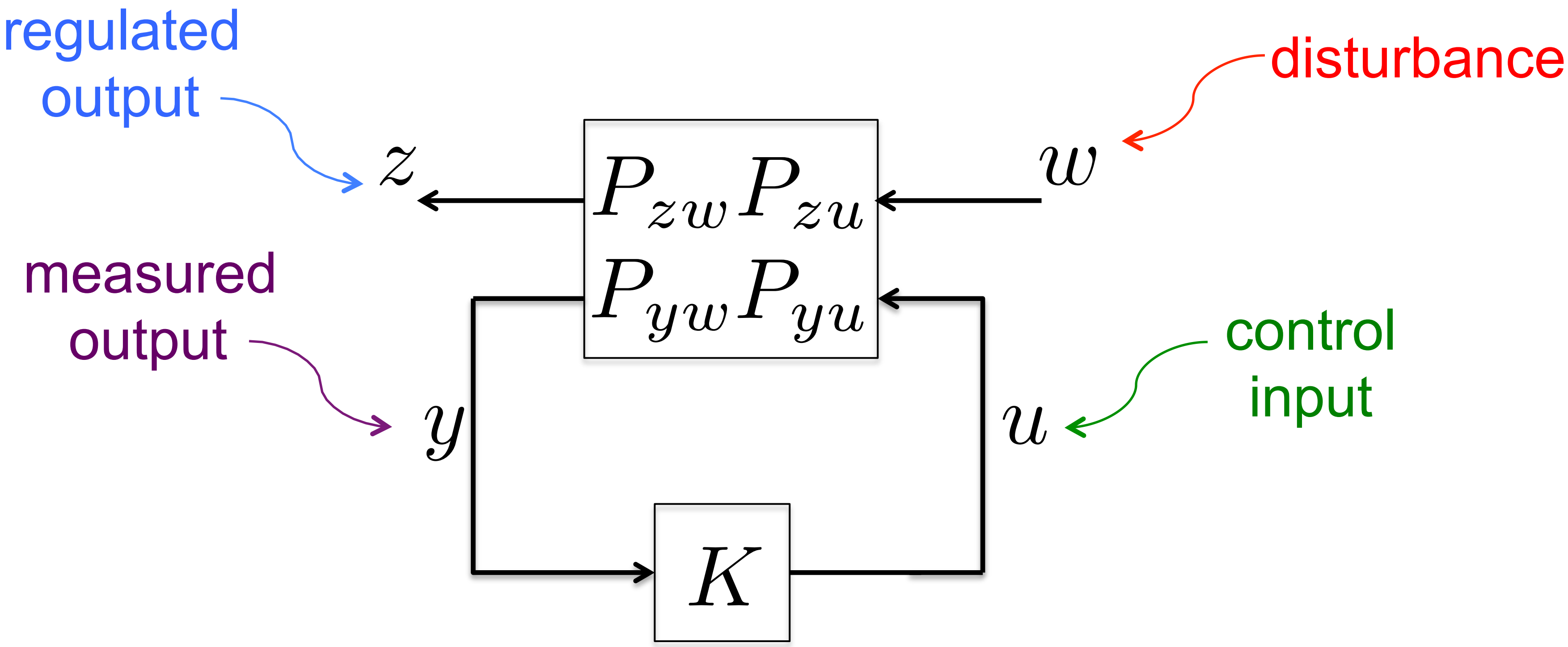
Classical Optimal Control Theory



$$\begin{aligned} \text{minimize}_K \, & \left\| \boxed{P_{zw} + P_{zu}K(I - P_{yu}K)^{-1}P_{yw}} \right\| \\ \text{s.t.} \ \ & K \text{ causal} \\ & K(I - P_{yu}K)^{-1} \text{ stable} \end{aligned}$$

closed loop map from disturbance → reg. output

Classical Optimal Control Theory



regulated output

disturbance

measured output

control input

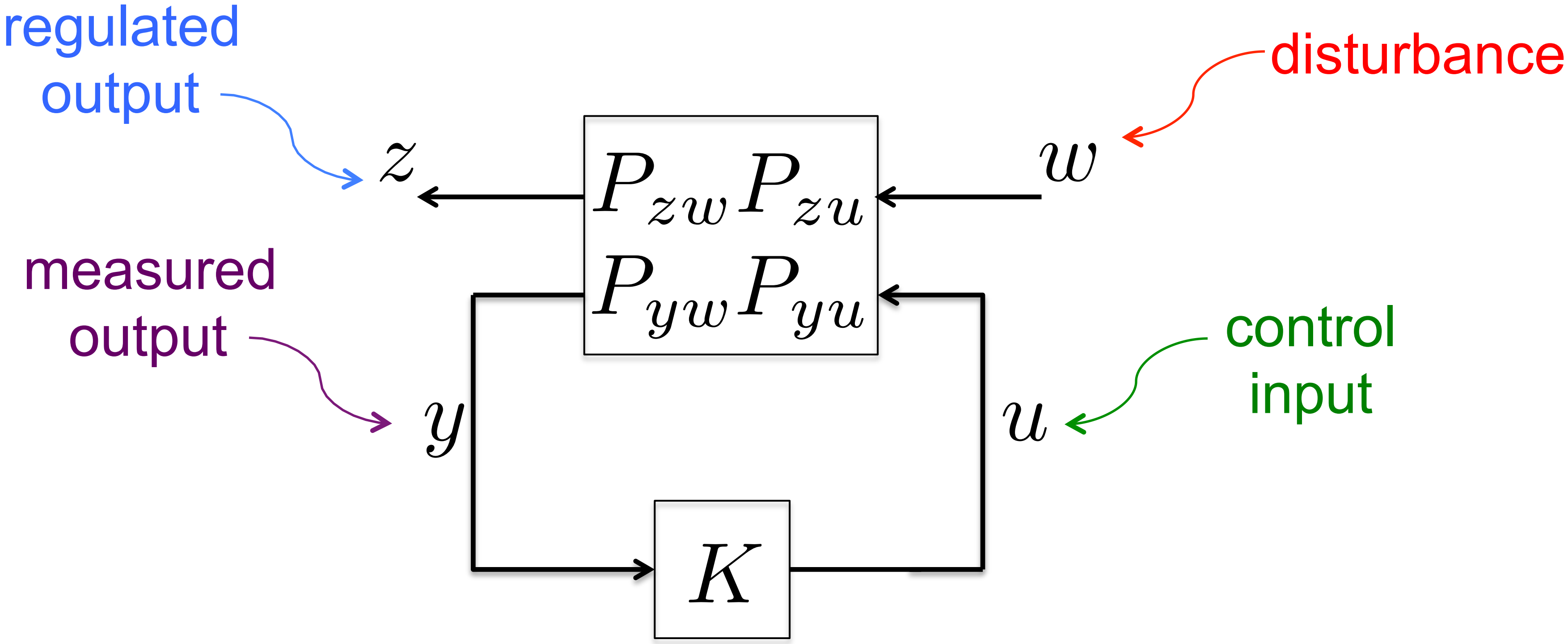
$$\text{minimize}_K \|P_{zw} + P_{zu}\boxed{K(I - P_{yu}K)^{-1}}P_{yw}\|$$

$$\text{s.t.} \;\; K \text{ causal}$$

$$\boxed{K(I - P_{yu}K)^{-1}} \text{ stable}$$

Feedback is non-convex

Classical Optimal Control Theory



regulated output

disturbance

measured output

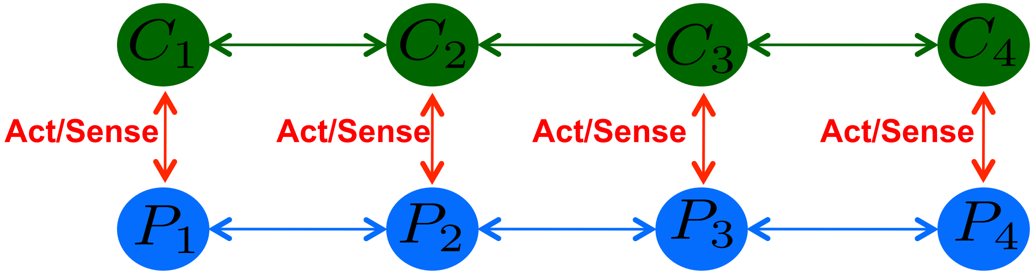
control input

$$\text{minimize}_Q \|P_{zw} + P_{zu} Q P_{yw}\|$$
$$\text{s.t. } Q \text{ stable \& causal}$$

Convex in Q

Distributed Optimal Control Theory

Many decision agents leads to information asymmetry



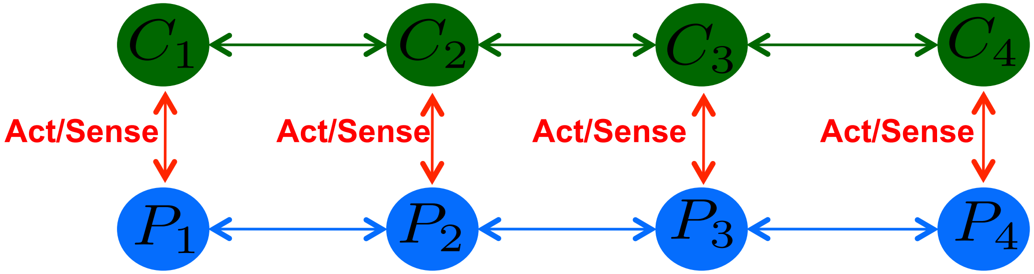
Manifests as *subspace constraints on K* in optimal control problem.

$$\begin{aligned} \text{minimize}_K \, & \|P_{zw} + P_{zu}K(I - P_{yu}K)^{-1}P_{yw}\| \\ \text{s.t.} \; & K \text{ causal} \\ & K(I - P_{yu}K)^{-1} \text{ stable} \\ & \boxed{K \in \mathcal{S}} \end{aligned}$$

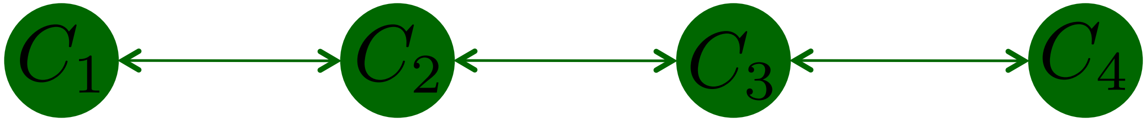
Distributed constraint

Distributed Optimal Control Theory

Many decision agents leads to information asymmetry



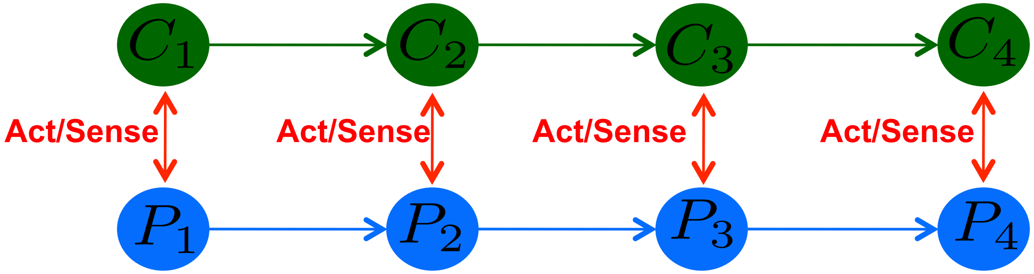
Manifests as *subspace constraints on K* in optimal control problem.



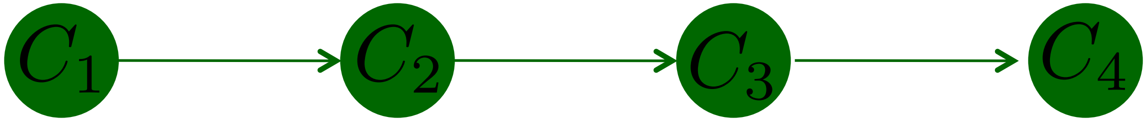
$$\mathcal{S} = \underset{t=-1}{\frac{1}{z}\begin{bmatrix} * & 0 & 0 & 0 \\ 0 & * & 0 & 0 \\ 0 & 0 & * & 0 \\ 0 & 0 & 0 & * \end{bmatrix}} \bigoplus \underset{t=-2}{\frac{1}{z^2}\begin{bmatrix} * & * & 0 & 0 \\ * & * & * & 0 \\ 0 & * & * & * \\ 0 & 0 & * & * \end{bmatrix}} \bigoplus \underset{t=-3}{\frac{1}{z^3}\begin{bmatrix} * & * & * & 0 \\ * & * & * & * \\ * & * & * & * \\ 0 & * & * & * \end{bmatrix}} \bigoplus \frac{1}{z^4}\mathcal{R}_p$$

Distributed Optimal Control Theory

Many decision agents leads to information asymmetry



Manifests as *subspace constraints on K* in optimal control problem.



$$\mathcal{S} = \begin{bmatrix} * & 0 & 0 & 0 \\ * & * & 0 & 0 \\ * & * & * & 0 \\ * & * & * & * \end{bmatrix}$$

Quadratic Invariance

A constraint set *S* is *QI under* P_{yu} if

$$K P_{yu} K \in \mathcal{S}, \; \forall K \in \mathcal{S}$$

If *S* is *QI under* $P_{yu,}$ then $K \in \mathcal{S}$ if and only if $Q \in \mathcal{S}$

If we have QI, model matching problem becomes

$$\begin{aligned} \text{minimize}_Q \quad & \|P_{zw} + P_{zu} Q P_{yw}\| \\ \text{s.t.} \quad & Q \text{ stable \& causal} \\ & Q \in \mathcal{S} \end{aligned}$$

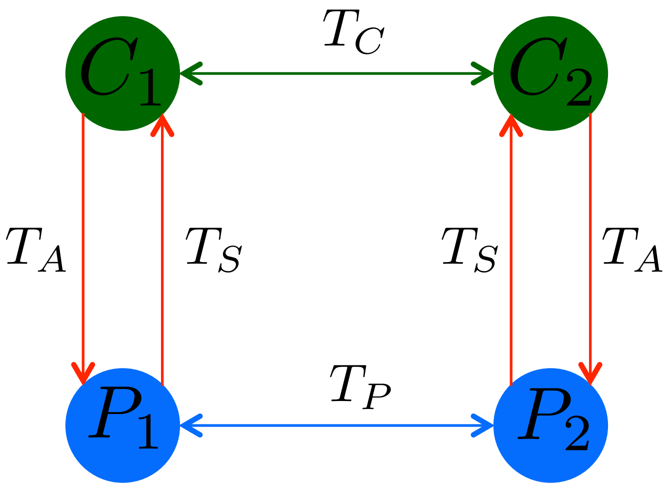
Convex in *Q!*

How does this relate to our intuition about signaling?

Quadratic Invariance for Delay Patterns

QI if & only if $T_C \leq T_A + T_S + T_P$

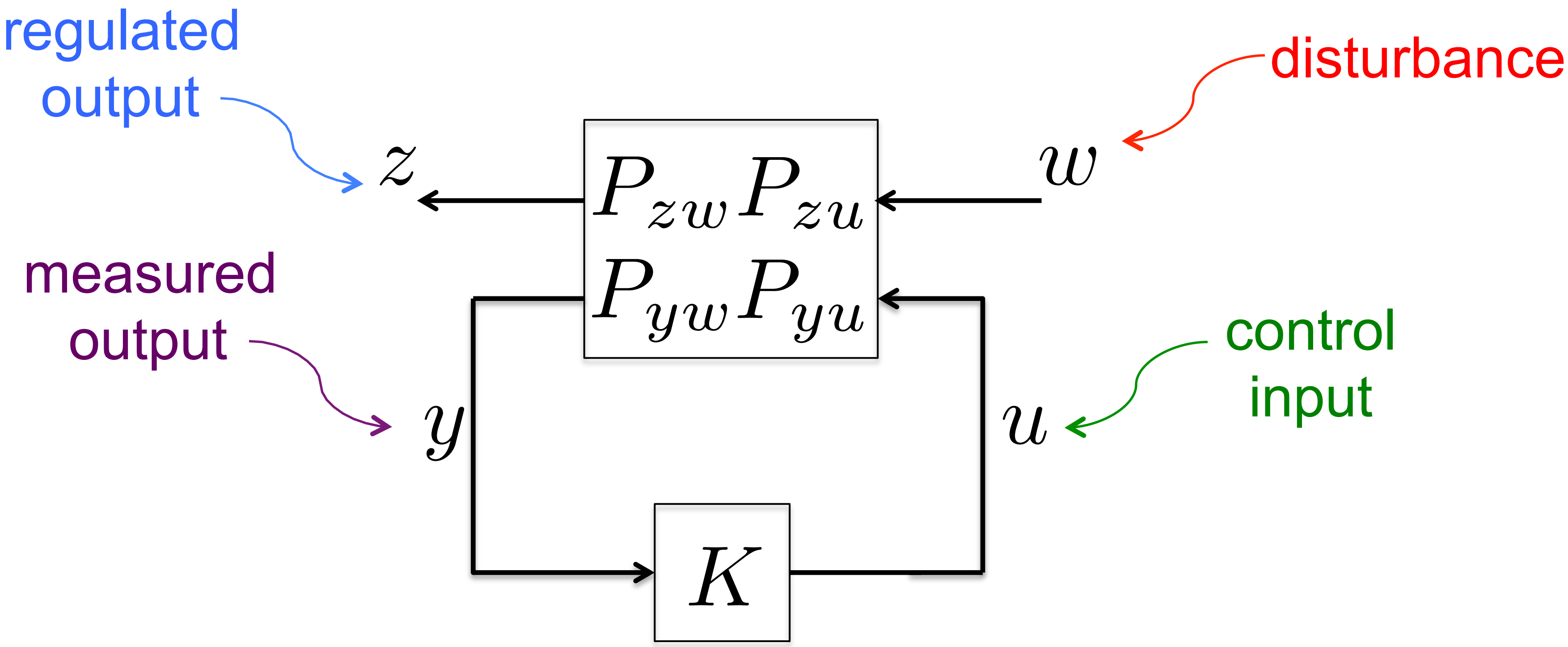
(Rotkowitz, Cogill & Lall *'10*)



T_C: communication delay
T_A: actuation delay
T_S: sensing delay
T_P: propagation delay

No incentive to "signal through the plant"

Distributed Optimal Control Theory



$$
\begin{aligned}
\text{minimize}_Q \quad & \|P_{zw} + P_{zu} Q P_{yw}\| \\
\text{s.t.} \quad & Q \text{ stable \& causal} \\
& \boxed{Q \in \mathcal{S}}
\end{aligned}
$$

Distributed constraint

Distributed Optimal Control Theory

Outline two recent results in H2 (LQG) distributed control:

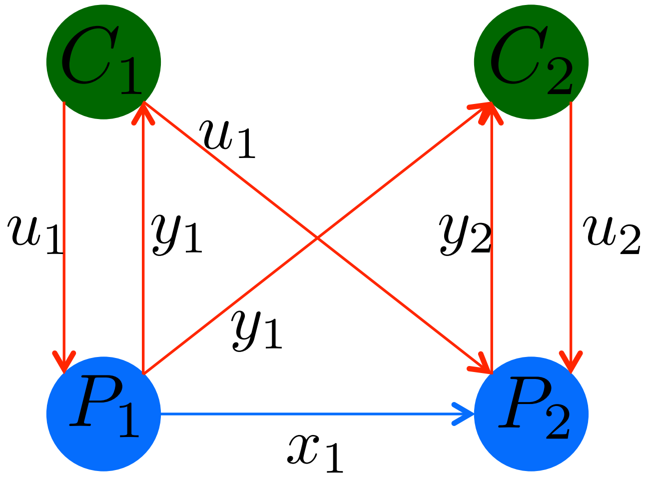
1) **two player nested information** structures (Lessard & Lall ‘12)

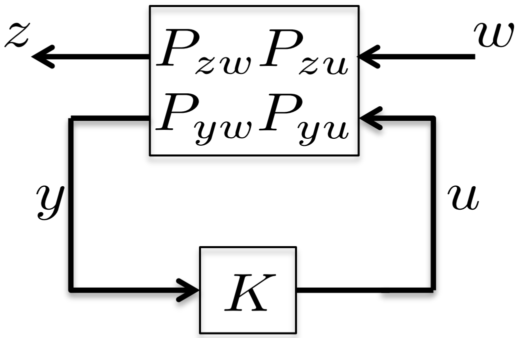
2) **strongly connected** communication graphs (Lamperski & Doyle ‘13)

To reduce to finite dimensional solution:
exploit structure to find centralized sub-problems
+ some other stuff

Other approaches : poset causal systems, finite subspace approximations, SDP based solutions

Two Player Nested Structure





Player 1 measures y_1 and chooses u_1
Player 2 measures y_1, y_2 and chooses u_2

Lower block triangular structure

$$P_{yu} = \begin{bmatrix} * & 0 \\ * & * \end{bmatrix} \quad K = \begin{bmatrix} * & 0 \\ * & * \end{bmatrix}$$

Two Player Nested Structure

How can we exploit lower block triangular structure to reduce to centralized problems?

Sweep stabilization issues, etc. under the rug –
see Lessard & Lall TAC '14 for details

$$\begin{aligned} \underset{Q}{\text{minimize}} \quad & \|P_{zw} + P_{zu} Q P_{uw}\|_{\mathcal{H}_2}^2 \\ \text{subject to} \quad & Q \text{ stable and lower} \end{aligned}$$

$$P_{yu} = \begin{bmatrix} * & 0 \\ * & * \end{bmatrix} \qquad K = \begin{bmatrix} * & 0 \\ * & * \end{bmatrix}$$

Player 1 measures *y_1* and chooses *u_1*
Player 2 measures *y_1, y_2 and* chooses *u_2*

Two Player Nested Structure

How can we exploit lower block triangular structure to reduce to centralized problems?

$$\begin{bmatrix} Q_{11} & 0 \\ Q_{12} & Q_{22} \end{bmatrix} = E_1 Q_{11} E_1^\top + E_2 \begin{bmatrix} Q_{12} & Q_{22} \end{bmatrix} = \begin{bmatrix} Q_{11} \\ Q_{12} \end{bmatrix} E_1^\top + E_2 Q_{22} E_2^\top$$

Centralized!!!

Two Player Nested Structure

How can we exploit lower block triangular structure to reduce to centralized problems?

$$\begin{bmatrix} Q_{11} & 0 \\ Q_{12} & Q_{22} \end{bmatrix} \quad = \quad E_1 Q_{11} E_1^\top + E_2 \begin{bmatrix} Q_{12} & Q_{22} \end{bmatrix} \quad = \quad \begin{bmatrix} Q_{11} \\ Q_{12} \end{bmatrix} E_1^\top + E_2 Q_{22} E_2^\top$$

Centralized!!!

Fix *Q₁₁* and solve

$$\begin{array}{ll} \underset{[Q_{12}\; Q_{22}]}{\text{minimize}} & \left\| (P_{zw} + P_{zu} E_1 Q_{11} E_1^\top P_{uw}) + P_{zu} E_2 \begin{bmatrix} Q_{12} & Q_{22} \end{bmatrix} P_{uw} \right\|_{\mathcal{H}_2}^2 \\ \text{subject to} & \begin{bmatrix} Q_{12} & Q_{22} \end{bmatrix} \text{ stable} \end{array}$$

To get optimal $\begin{bmatrix} Q_{12}^{\#} & Q_{22}^{\#} \end{bmatrix}$

Two Player Nested Structure

How can we exploit lower block triangular structure to reduce to centralized problems?

$$\begin{bmatrix} Q_{11} & 0 \\ Q_{12} & Q_{22} \end{bmatrix} = E_1 Q_{11} E_1^\top + E_2 \begin{bmatrix} Q_{12} & Q_{22} \end{bmatrix} = \begin{bmatrix} Q_{11} \\ Q_{12} \end{bmatrix} E_1^\top + E_2 Q_{22} E_2^\top$$

Centralized!!!

Fix Q_{22} and solve

$$\begin{array}{ll} \underset{\begin{bmatrix} Q_{11}^H & Q_{12}^H \end{bmatrix}^H}{\text{minimize}} & \left\|(P_{zw} + P_{zu} E_2 Q_{22} E_2^\top P_{uw}) + P_{zu} \begin{bmatrix} Q_{11} \\ Q_{12} \end{bmatrix} E_1^\top P_{uw}\right\|_{\mathcal{H}_2}^2 \\ \text{subject to} & \begin{bmatrix} Q_{11}^H & Q_{12}^H \end{bmatrix}^H \text{ stable} \end{array}$$

To get optimal $\begin{bmatrix} Q_{11}^* \\ Q_{12}^* \end{bmatrix}$

Two Player Nested Structure

How can we exploit lower block triangular structure to reduce to centralized problems?

By uniqueness of optimal solution

$$Q_{opt} = \begin{bmatrix} Q_{11}^{*} & 0 \\ Q_{12}^{*} & Q_{22}^{\#} \end{bmatrix} = \begin{bmatrix} Q_{11}^{*} & 0 \\ Q_{12}^{\#} & Q_{22}^{\#} \end{bmatrix}$$

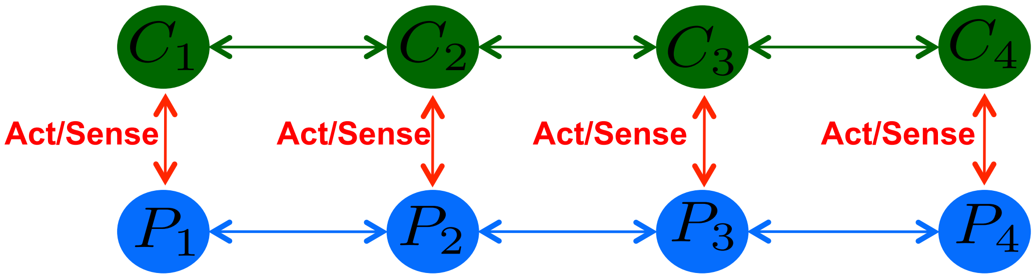
Main idea: use structure to get centralized problems, and then do some extra "stuff"

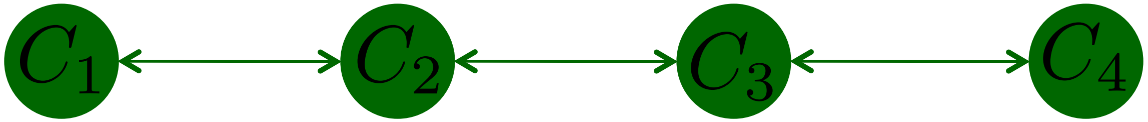
Generalizes to other nested topologies such as N-player chain

(Lessard et al. '14, Tanaka and Parrilo '14)

Strongly Connected Communication Graphs

How can we exploit strongly connected structure to reduce to centralized problems?





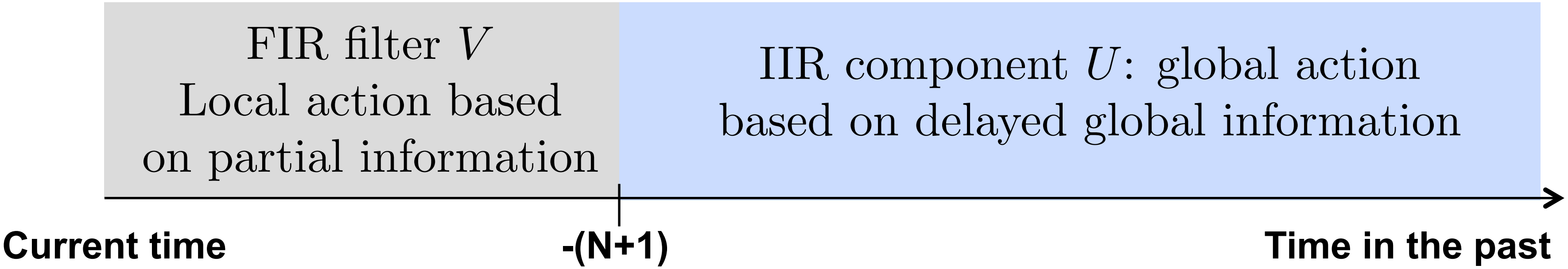
$$\mathcal{S} = \underset{t=-1}{\frac{1}{z}\begin{bmatrix} * & 0 & 0 & 0 \\ 0 & * & 0 & 0 \\ 0 & 0 & * & 0 \\ 0 & 0 & 0 & * \end{bmatrix}} \oplus \underset{t=-2}{\frac{1}{z^2}\begin{bmatrix} * & * & 0 & 0 \\ * & * & * & 0 \\ 0 & * & * & * \\ 0 & 0 & * & * \end{bmatrix}} \oplus \underset{t=-3}{\frac{1}{z^3}\begin{bmatrix} * & * & * & 0 \\ * & * & * & * \\ * & * & * & * \\ 0 & * & * & * \end{bmatrix}} \oplus \frac{1}{z^4}\mathcal{R}_p$$

Strongly Connected Communication Graphs

How can we exploit strongly connected structure to reduce to centralized problems?

$$\mathcal{S} = \mathcal{Y} \oplus \frac{1}{z^{N+1}}\mathcal{R}_p$$

$$Q = V \oplus U$$

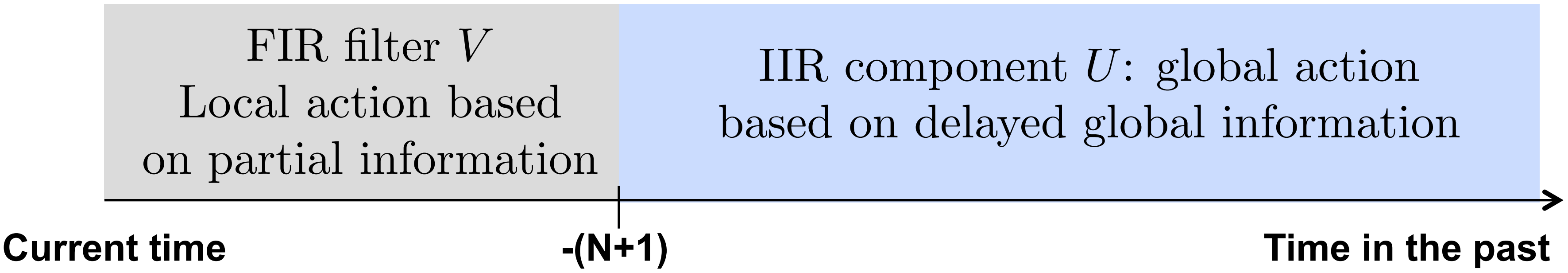


We can play the same game: rewrite *Q* and solve for *U* in terms of *V*

Strongly Connected Communication Graphs

How can we exploit strongly connected structure
to reduce to centralized problems?

$$Q = V \oplus U$$



FIR filter V
Local action based
on partial information

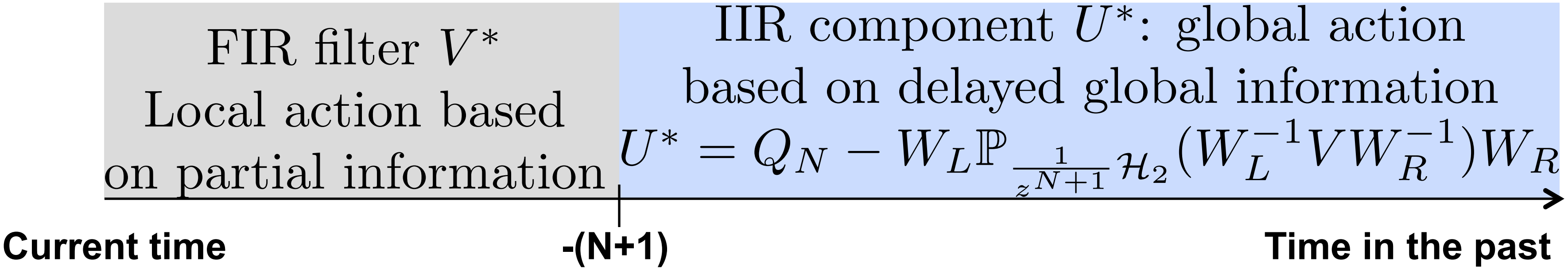
IIR component U: global action
based on delayed global information

$$\begin{aligned} \underset{U}{\text{minimize}} \quad & \|P_{zw} + P_{zu} V P_{uw} + P_{zu} U P_{uw}\|_{\mathcal{H}_2}^2 \\ \text{subject to} \quad & U \in \tfrac{1}{z^{N+1}} \mathcal{H}_2 \end{aligned}$$

Delayed but centralized: can get analytic solution in terms of *V*.
Again some magic happens, and problem reduces to…
(Lamperski & Doyle ’13 and ‘14)

Strongly Connected Communication Graphs

- Optimal controller has 2 regimes



FIR filter V^*

Local action based on partial information

IIR component U^*: global action based on delayed global information

$$U^* = Q_N - W_L \mathbb{P}_{\frac{1}{z^{N+1}}\mathcal{H}_2}(W_L^{-1} V W_R^{-1}) W_R$$

After *N+1* steps: each node has access to global delayed state.

Key feature: Finite impulse response (FIR) filter *V** solves:

$$\text{minimize}_V \sum_{i=1}^{N} \left(\text{Tr} G_i(V) \left(G_i(V)\right)^\top + 2\text{Tr} G_i(V) T_i^\top \right)$$

$$\text{s.t. } V_i \in \mathcal{Y}_i$$

Distributed Control

Large scale systems not amenable to centralized control

Idea: restrict information each controller has access to

Positives: control laws are local, and hence scalable to implement.

Negatives: in general non-convex. Witsenhausen.

Distributed Control

Large scale systems not amenable to centralized control

Idea: restrict information each controller has access to

Positives: control laws are local, and hence scalable to implement.

Negatives: in general non-convex. Witsenhausen.

Positives: with additional structure, regain convexity and finite dimensionality.

Distributed Control

Large scale systems not amenable to centralized control

Idea: restrict information each controller has access to

Positives: control laws are local, and hence scalable to implement.

Negatives: in general non-convex. Witsenhausen.

Positives: with additional structure, regain convexity and finite dimensionality.

Negatives: had to give up scalability in the process.

Distributed Control

In all cases, optimal controller is as expensive to compute as centralized counter part

and

Can be even more difficult to implement!

What structure do we need to impose to maintain convexity and regain scalability?

Distributed Control

In all cases, optimal controller is as expensive to compute as centralized counter part

and

Can be even more difficult to implement!

What structure do we need to impose to maintain convexity and regain scalability?

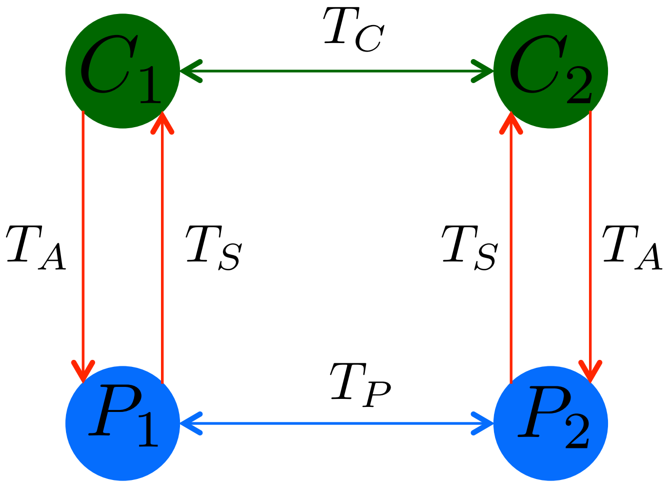
LOCALIZABILITY

(Wang, M., You & Doyle ‘13, Wang, M., & Doyle ‘13)

Quadratic Invariance for Delay Patterns

QI if & only if $T_C \leq T_A + T_S + T_P$

(Rotkowitz, Cogill & Lall *'10*)

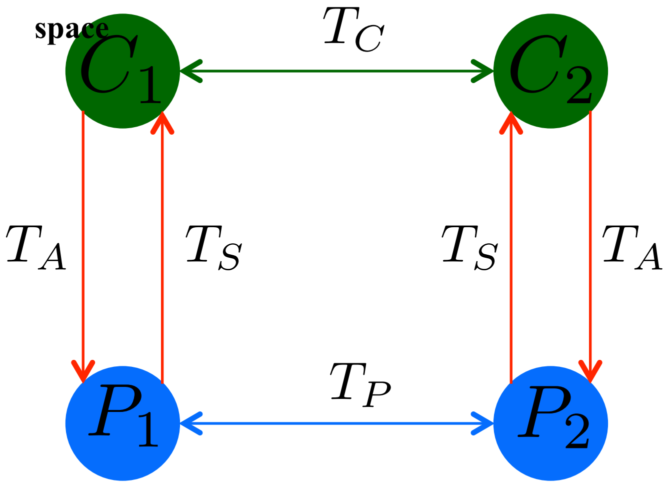


T_C: communication delay
T_A: actuation delay
T_S: sensing delay
T_P: propagation delay

No incentive to "signal through the plant"

Localizability

Localizability requires $T_C + T_A + T_S \leq T_P$

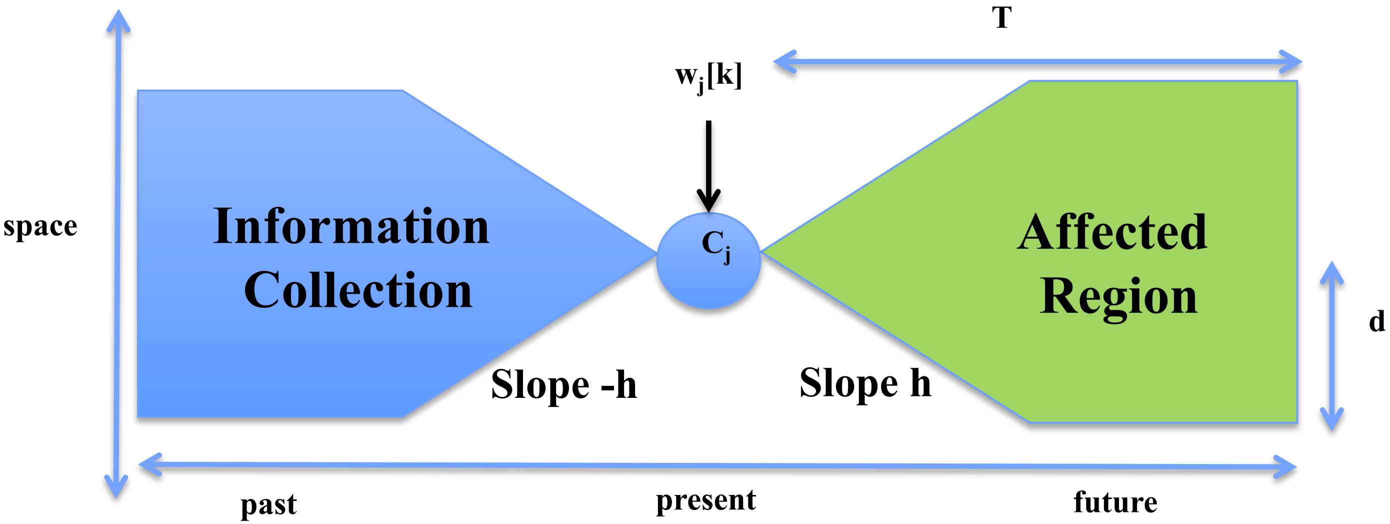


T_C: communication delay
T_A: actuation delay
T_S: sensing delay
T_P: propagation delay

Get ahead of disturbance and cancel it out

Localizability

Localizing Control Scheme



Get ahead of disturbance and cancel it out

Localizability

Spatio-temporal deadbeat control at each node

$$\begin{array}{rl} \underset{x[k],\, u[k]}{\text{minimize}} & f(x[0:k], u[0:k]) \\ \text{subject to} & x[0] = e_i \\ & x[k+1] = Ax[k] + Bu[k] \\ & x[k] \in \mathcal{S}_x \\ & u[1:k] \in \mathcal{S}_u \\ & x[T] = 0 \end{array}$$

Localizability

Spatio-temporal deadbeat control at each node

$$
\begin{array}{ll}
\underset{x[k],\,u[k]}{\text{minimize}} & f(x[0:k], u[0:k]) \\
\text{subject to} & x[0] = e_i \\
& x[k+1] = Ax[k] + Bu[k] \\
& x[k] \in \mathcal{S}_x \\
& u[1:k] \in \mathcal{S}_u \\
& x[T] = 0
\end{array}
$$

Favorite convex cost

Localizability

Spatio-temporal deadbeat control at each node

$$\begin{array}{ll} \underset{x[k],\,u[k]}{\text{minimize}} & f(x[0:k], u[0:k]) \\ \text{subject to} & x[0] = e_i \\ & x[k+1] = Ax[k] + Bu[k] \\ & x[k] \in \mathcal{S}_x \\ & u[1:k] \in \mathcal{S}_u \\ & x[T] = 0 \end{array}$$

Favorite convex cost

Initial disturbance

Localizability

Spatio-temporal deadbeat control at each node

$$
\begin{array}{ll}
\underset{x[k],\,u[k]}{\text{minimize}} & f(x[0:k], u[0:k]) \\
\text{subject to} & x[0] = e_i \\
& x[k+1] = Ax[k] + Bu[k] \\
& x[k] \in \mathcal{S}_x \\
& u[1:k] \in \mathcal{S}_u \\
& x[T] = 0
\end{array}
$$

Favorite convex cost

Initial disturbance

Dynamics

Localizability

Spatio-temporal deadbeat control at each node

$$\begin{aligned} \underset{x[k],\, u[k]}{\text{minimize}} \quad & f(x[0:k], u[0:k]) \\ \text{subject to} \quad & x[0] = e_i \\ & x[k+1] = Ax[k] + Bu[k] \\ & x[k] \in \mathcal{S}_x \\ & u[1:k] \in \mathcal{S}_u \\ & x[T] = 0 \end{aligned}$$

Favorite convex cost

Initial disturbance

Dynamics

Spatial constraints

Localizability

Spatio-temporal deadbeat control at each node

$$\begin{aligned} \underset{x[k],\,u[k]}{\text{minimize}} \quad & f(x[0:k],u[0:k]) \\ \text{subject to} \quad & x[0]=e_i \\ & x[k+1]=Ax[k]+Bu[k] \\ & x[k]\in\mathcal{S}_x \\ & u[1:k]\in\mathcal{S}_u \\ & x[T]=0 \end{aligned}$$

Favorite convex cost

Initial disturbance

Dynamics

Spatial constraints

Comm constraints

Localizability

Spatio-temporal deadbeat control at each node

$$\begin{array}{ll} \underset{x[k],\,u[k]}{\text{minimize}} & f(x[0:k], u[0:k]) \\ \text{subject to} & x[0] = e_i \\ & x[k+1] = Ax[k] + Bu[k] \\ & x[k] \in \mathcal{S}_x \\ & u[1:k] \in \mathcal{S}_u \\ & x[T] = 0 \end{array}$$

Favorite convex cost

Initial disturbance

Dynamics

Spatial constraints

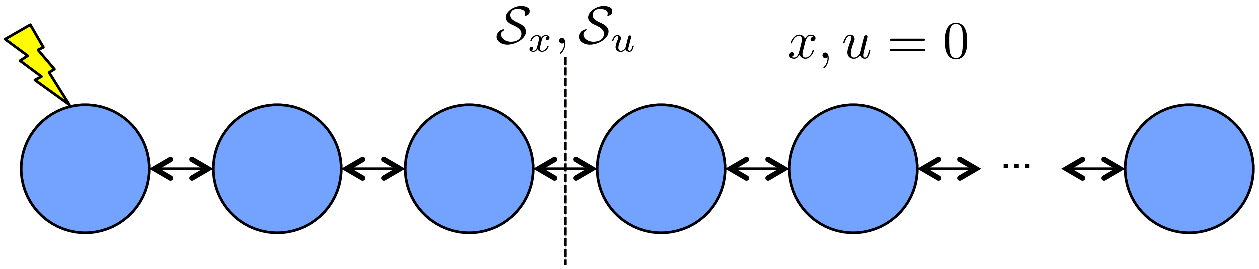
Comm constraints

Temporal constraints

Localizability

Spatio-temporal deadbeat control at each node

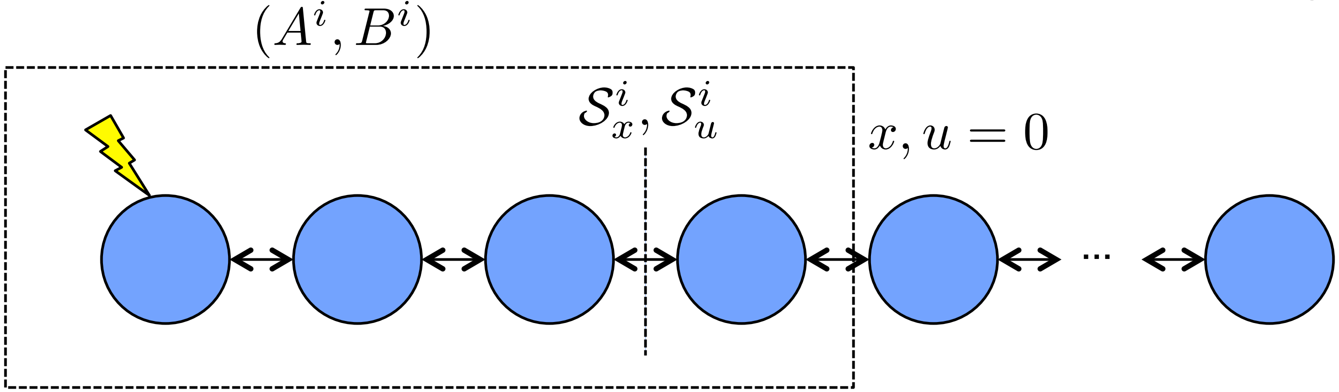
$\underset{x[k],\,u[k]}{\text{minimize}} \quad f(x[0:k], u[0:k])$	**Favorite convex cost**
$\text{subject to} \quad x[0] = e_i$	**Initial disturbance**
$x[k+1] = Ax[k] + Bu[k]$	**Dynamics**
$x[k] \in \mathcal{S}_x$	**Spatial constraints**
$u[1:k] \in \mathcal{S}_u$	**Comm constraints**
$x[T] = 0$	**Temporal constraints**



Localizability

Spatio-temporal deadbeat control at each node
lets us restrict to sub-models for design/implementation

$\underset{x^i[k],\, u^i[k]}{\text{minimize}} \quad f(x^i[0:k], u^i[0:k])$	**Favorite convex cost**
$\text{subject to} \quad x^i[0] = e_i$	**Initial disturbance**
$x^i[k+1] = A^i x[k] + B^i u[k]$	**Dynamics**
$x^i[k] \in \mathcal{S}_x^i$	**Spatial constraints**
$u^i[1:k] \in \mathcal{S}_u^i$	**Comm constraints**
$x^i[T] = 0$	**Temporal constraints**

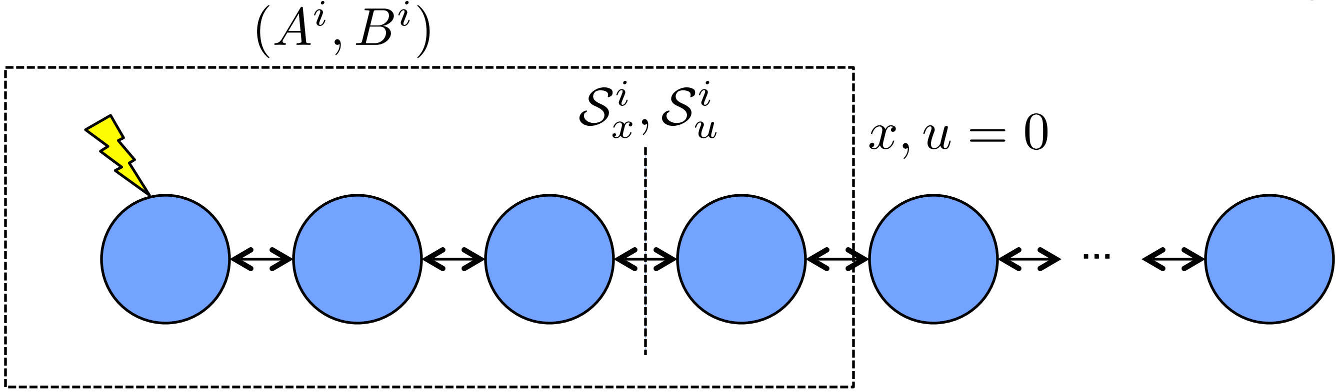


Localizability

LQR cost splits along disturbances:

Completely Local Globally Optimal Solution

$$
\begin{array}{rll}
\underset{x^i[k],\, u^i[k]}{\text{minimize}} & \|x^i[0:k]\|_2^2 + \|u^i[0:k]\|_2^2 & \textbf{LQR cost} \\
\text{subject to} & x^i[0] = e_i & \textbf{Initial disturbance} \\
& x^i[k+1] = A^i x[k] + B^i u[k] & \textbf{Dynamics} \\
& x^i[k] \in \mathcal{S}_x^i & \textbf{Spatial constraints} \\
& u^i[1:k] \in \mathcal{S}_u^i & \textbf{Comm constraints} \\
& x^i[T] = 0 & \textbf{Temporal constraints}
\end{array}
$$



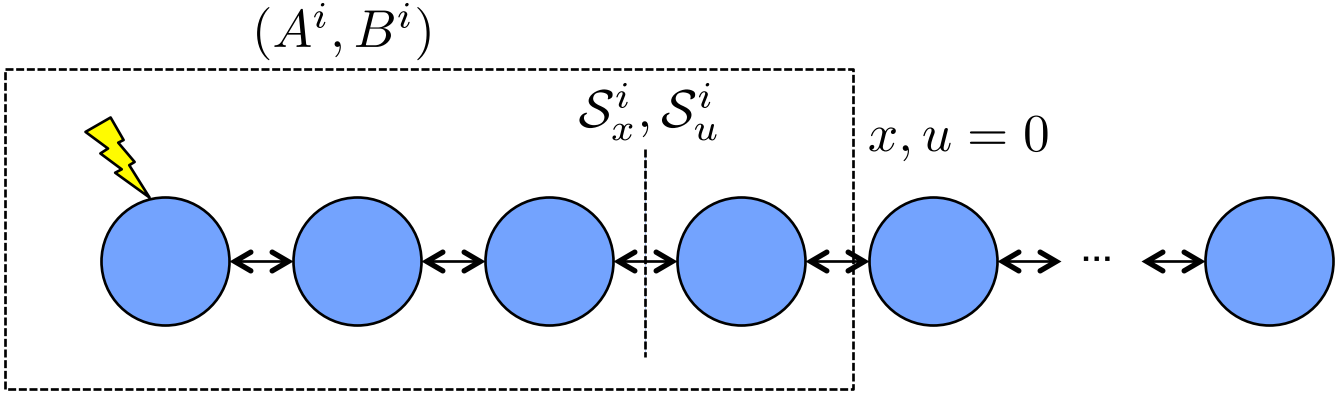
Localizability

Extensions in the works for

Output feedback

and

Non-separable cost functions



Roadmap for 1st Part

DC OPF

- Connections to positive systems
- Connections to Sum of Squares Programming & Polynomial Optimization

Distributed Optimal Control

- Why it's hard: Witsenhausen
- How can we make it tractable: Quadratic Invariance
- How can we make it scalable: Localizable Systems

Setup for 2nd Part

Break

Recap of 1st Part

“Easy” problems are convex and scalable

Interesting problems are large scale and non-convex

Solution: Exploit Structure to Relax

Indefinite QPs are hard in general

DC OPF is tractable because of **Metzler structure**

Distributed control is hard in general

Computationally tractable if we have **QI**

Scalable if we have **localizability**

What have we swept under the rug?

Made lots of assumptions for distributed control

Can communicate with **infinite bandwidth**

Communication occurs with **fixed delays**

Have a **known system model** with **known structure**

Have a **control architecture** (actuation, sensing, communication)

Roadmap for 2nd Part

Networked Control Systems

- Single plant/controller: connections with information theory
- Approaches for extending to distributed control

Varying Delays

- Recent progress

Distributed System Identification

- Known structure
- Unknown structure

Control Architecture Design

Roadmap for 2nd Part

Networked Control Systems

- Single plant/controller: connections with information theory
- Approaches for extending to distributed control

Varying Delays

- Recent progress

Distributed System Identification

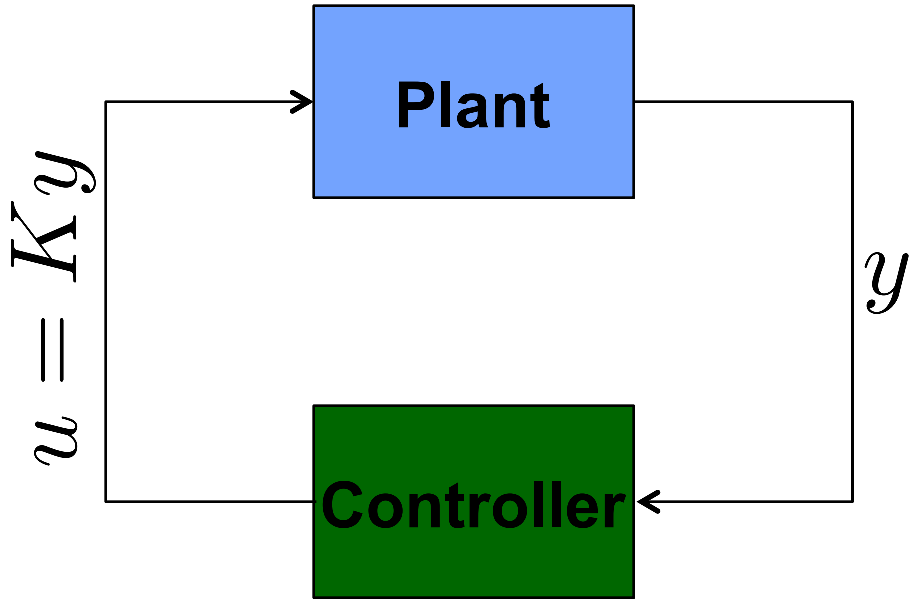
- Known structure
- Unknown structure

Control Architecture Design

Emphasize Connections to Optimization & Statistics

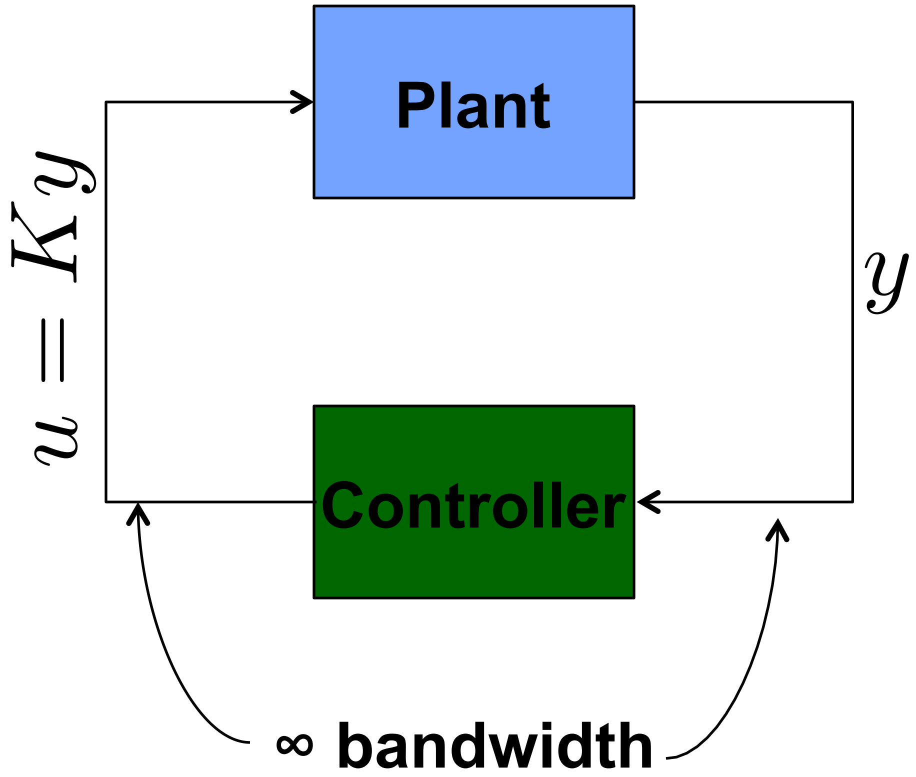
Networked Control Systems

Classical control system



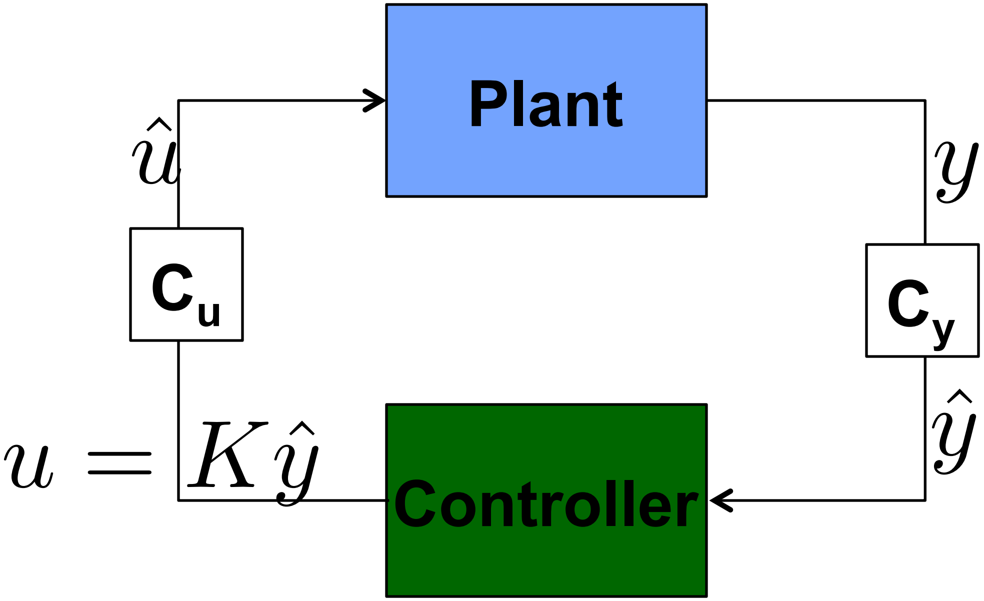
Networked Control Systems

Classical control system



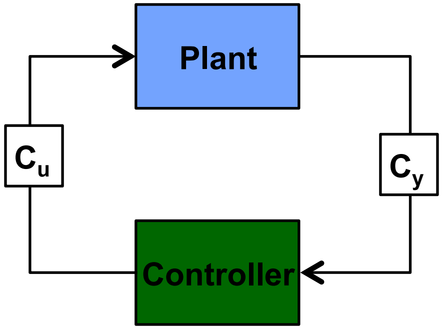
Networked Control Systems

Networked control system



Adding realistic channels leads to interplay between information and control theory

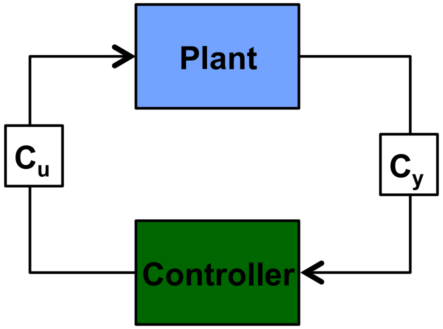
Networked Control Systems



Stabilization well understood

$$\text{Channel Capacity} \geq \text{Plant “instability”}$$

Networked Control Systems

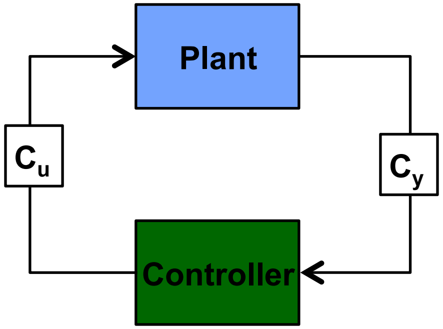


Stabilization well understood

$$\text{Channel Capacity} \geq \text{Plant ``instability''}$$

$$\text{Plant ''instability'': Entropy } H = \sum_{|\lambda_j| \geq 1} \log_2 \lambda_j$$

Networked Control Systems



Stabilization well understood

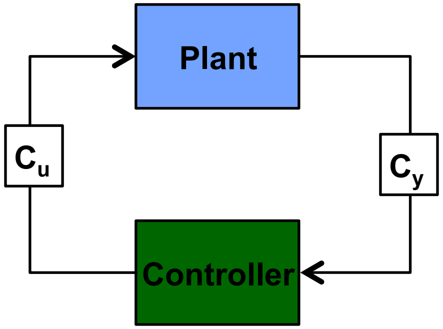
$$\text{Channel Capacity} \geq \text{Plant “instability”}$$

$$\text{Plant ”instability”: Entropy } H = \sum_{|\lambda_j| \geq 1} \log_2 \lambda_j$$

Examples

Channel Type	Condition	Reference
Limited data rate *R*	$R > H$	Nair & Evans ‘04
SNR constrained AWGN	$\frac{C}{\log_2 e} > \sum_{\lambda_i : \mathrm{Re}\lambda_i > 0} \mathrm{Re}\lambda_i$	Braslavsky, Middleton & Freudenberg ’07
Noisy and quantized	Anytime reliability > *H*	Sahai and Mitter ‘06

Networked Control Systems



Stabilization well understood

$$\text{Channel Capacity} \geq \text{Plant "instability"}$$

$$\text{Plant "instability": Entropy } H = \sum_{|\lambda_j|\geq 1} \log_2 \lambda_j$$

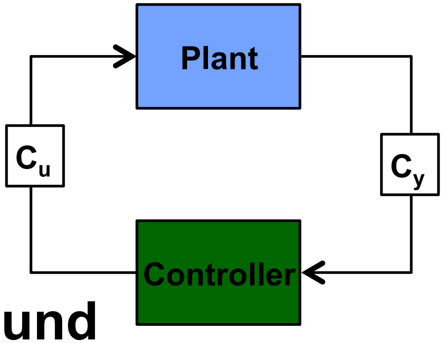
Examples

Channel Type	Condition	Reference
Limited data rate *R*	$R > H$	Nair & Evans '04
SNR constrained AWGN	$\frac{C}{\log_2 e} > \sum_{\lambda_i : \mathrm{Re}\lambda_i > 0} \mathrm{Re}\lambda_i$	Braslavsky, Middleton & Freudenberg '07
Noisy and quantized	Anytime reliability > *H*	Sahai and Mitter '06

Extensions to varying rates (Minero et. al '09, '13)

Tree codes for achieving anytime reliability (Sukhavasi & Hassibi '13)

Networked Control Systems



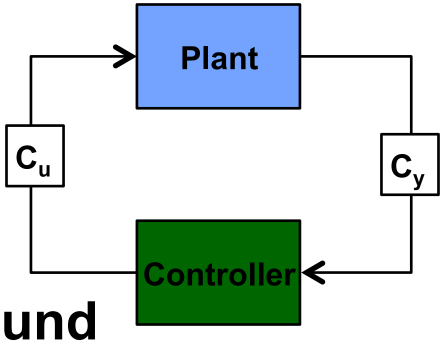
Performance limits well understood

Martins and Dahleh '08

No channel gives us standard* Bode integral bound

$$\frac{1}{2\pi}\int_{-\pi}^{\pi}\log(S(\omega))d\omega \geq \sum_{i=1}^{n}\max\{0,\log|\lambda_i(A)|\}$$

Networked Control Systems



Performance limits well understood

Martins and Dahleh '08

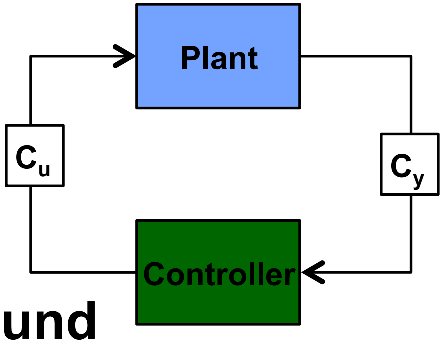
No channel gives us standard* Bode integral bound

$$\frac{1}{2\pi}\int_{-\pi}^{\pi}\log(S(\omega))d\omega \geq \sum_{i=1}^{n}\max\{0,\log|\lambda_i(A)|\}$$

Channel in the loop hurts us

$$\frac{1}{2\pi}\int_{-\pi}^{\pi}\min\{0,\log(S(\omega))\}d\omega \geq \sum_{i=1}^{n}\max\{0,\log|\lambda_i(A)|\} - C_f$$

Networked Control Systems



Performance limits well understood

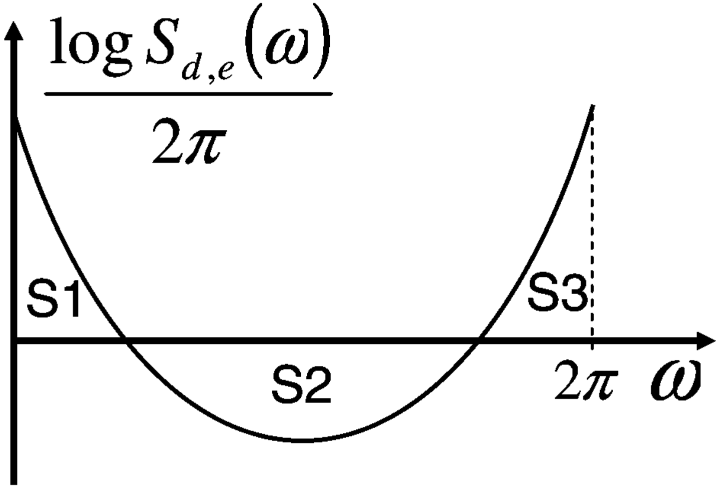
Martins and Dahleh '08

No channel gives us standard* Bode integral bound

$$\frac{1}{2\pi}\int_{-\pi}^{\pi}\log(S(\omega))d\omega \geq \sum_{i=1}^{n}\max\{0,\log|\lambda_i(A)|\}$$

Channel in the loop hurts us

$$\frac{1}{2\pi}\int_{-\pi}^{\pi}\min\{0,\log(S(\omega))\}d\omega \geq \sum_{i=1}^{n}\max\{0,\log|\lambda_i(A)|\} - C_f$$



Bode:

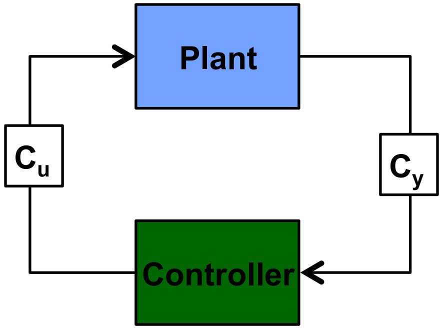
$$S1 + S3 - S2 \geq \sum_{i=1}^{n}\max\{0,\log|\lambda_i(A)|\}$$

New Inequality:

$$S2 \leq C_f - \sum_{i=1}^{n}\max\{0,\log|\lambda_i(A)|\}$$

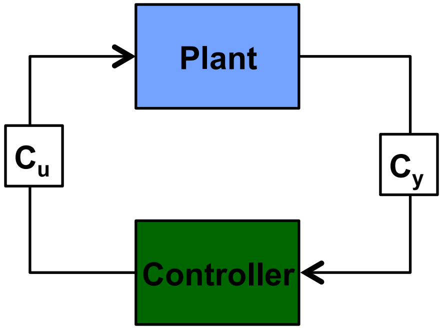
Figure borrowed from Martins, and Dahleh TAC'08

Networked Control Systems



Achieving these limits much less well understood

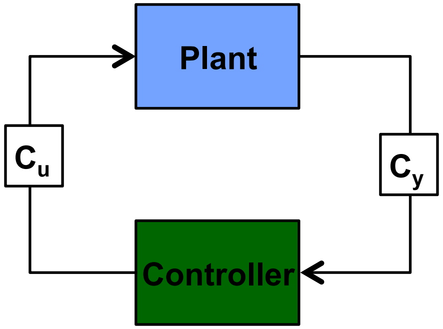
Networked Control Systems



Achieving these limits
much less well understood

Results exist for special cases

Networked Control Systems

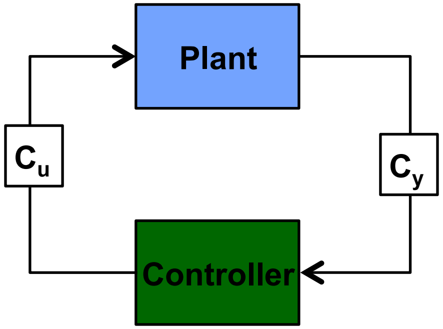


Achieving these limits much less well understood

Results exist for special cases

Even for a single plant and controller optimal control is difficult under noisy channels

Networked Control Systems



Achieving these limits
much less well understood

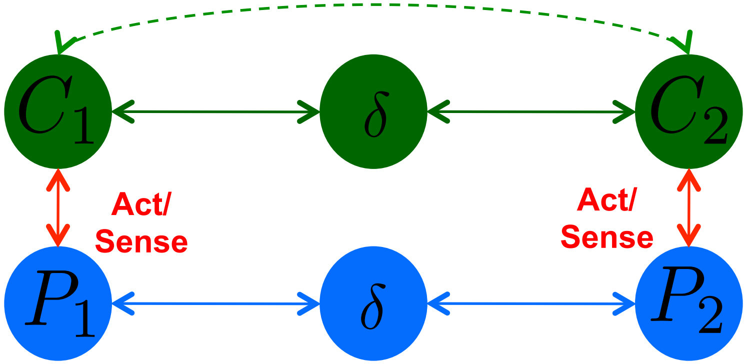
Results exist for special cases

Even for a single plant and controller
optimal control is difficult under noisy channels

Modeling assumption: underlying channel manifests
as possibly unbounded and varying delays

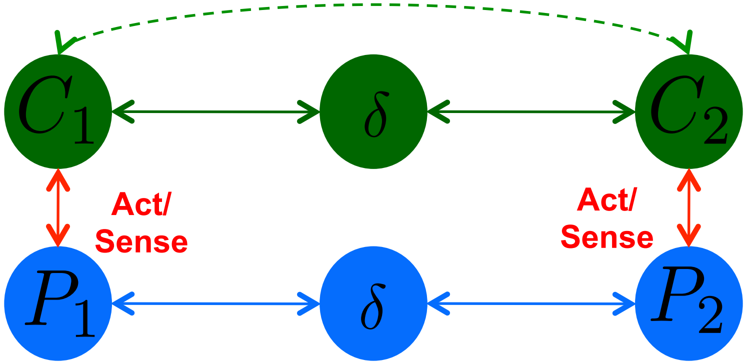
Varying Delays

Two player LQR state feedback with varying delay has explicit solution



Varying Delays

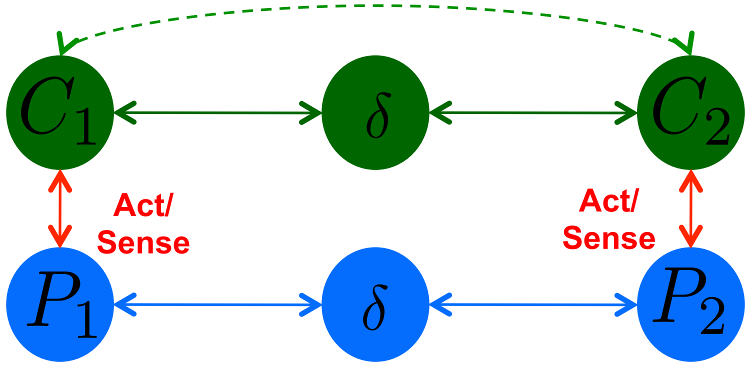
Two player LQR state feedback with varying delay has explicit solution



if delay pattern leads to partially nested information pattern throughout

Varying Delays

Two player LQR state feedback with varying delay has explicit solution



if delay pattern leads to partially nested information pattern throughout

Dynamic Programming based solution
(M. & Doyle ’13, M., Lamperski & Doyle ‘14)
Builds off of Lamperski & Doyle ‘12, Lamperski & Lessard ‘13

Varying Delays

Extensions to more general topologies?

Will require Dynamic Programming based solutions

Varying Delays

Extensions to more general topologies?

Will require Dynamic Programming based solutions

These should be available soon, as sufficient statistics are now well understood

"Sufficient statistics for linear control strategies in decentralized systems with partial history sharing, Mahajan & Nayyar", '14

"Sufficient statistics for team decision problems", Wu (& Lall), '13

Varying Delays

Extensions to more general topologies?

Will require Dynamic Programming based solutions

These should be available soon, as sufficient statistics are now well understood

"Sufficient statistics for linear control strategies in decentralized systems with partial history sharing, Mahajan & Nayyar", '14

"Sufficient statistics for team decision problems", Wu (& Lall), '13

Unbounded delays?

Varying Delays

Extensions to more general topologies?

Will require Dynamic Programming based solutions

These should be available soon, as sufficient statistics are now well understood

"Sufficient statistics for linear control strategies in decentralized systems with partial history sharing, Mahajan & Nayyar", '14

"Sufficient statistics for team decision problems", Wu (& Lall), '13

Unbounded delays?

Progress is promising on both the coding and control side

Roadmap for 2nd Part

Networked Control Systems

- Single plant/controller: connections with information theory
- Approaches for extending to distributed control

Varying Delays

- Recent progress

Distributed System Identification

- Known structure
- Unknown structure

Control Architecture Design

Emphasize Connections to Optimization & Statistics

SysID with Known Structure

Traditional subspace methods destroy structure

A good algorithm leverages structure rather than ignoring it

SysID with Known Structure

Traditional subspace methods destroy structure

A good algorithm leverages structure rather than ignoring it

We want convexity and scalability

SysID with Known Structure

Traditional subspace methods destroy structure

A good algorithm leverages structure rather than ignoring it

We want convexity and scalability

Can we exploit known structure to get an algorithm that is **local** (scalable) and **convex**

SysID with Known Structure

Quick Review of Basic SysID

Dynamics

$$\begin{aligned} x_{t+1} &= Ax_t + Bu_t \\ y_t &= Cx_t + Du_t \end{aligned}$$

Input/output

$$y_t = \sum_{\tau=0}^{t} G_\tau u_{t-\tau}$$

$$G_0 = D,\, G_\tau = CA^{\tau-1}B$$

SysID with Known Structure

Quick Review of Basic SysID

Dynamics

$$\begin{aligned} x_{t+1} &= Ax_t + Bu_t \\ y_t &= Cx_t + Du_t \end{aligned}$$

Input/output

$$y_t = \sum_{\tau=0}^{t} G_\tau u_{t-\tau}$$

$$G_0 = D,\ G_\tau = CA^{\tau-1}B$$

$$Y_N = \begin{bmatrix} y_{N-M} & y_{N-(M-1)} & \cdots & y_N \end{bmatrix} \quad G = \begin{bmatrix} G_0 & G_1 & \cdots & G_r \end{bmatrix}$$

$$U_{N,M,r} = \begin{bmatrix} u_{N-M} & u_{N-(M-1)} & \cdots u_N & \\ u_{N-(M+1)} & u_{N-M} & \cdots u_{N-1} & \\ \vdots & \vdots & \ddots & \vdots \\ u_{N-(M+r)} & u_{N-(M+r-1)} & \cdots u_{N-r} & \end{bmatrix}$$

SysID with Known Structure

Quick Review of Basic SysID

Dynamics

$$\begin{aligned} x_{t+1} &= Ax_t + Bu_t \\ y_t &= Cx_t + Du_t \end{aligned}$$

Input/output

$$y_t = \sum_{\tau=0}^{t} G_\tau u_{t-\tau}$$

$$G_0 = D,\ G_\tau = CA^{\tau-1}B$$

$$Y_N = \begin{bmatrix} y_{N-M} & y_{N-(M-1)} & \cdots & y_N \end{bmatrix} \quad G = \begin{bmatrix} G_0 & G_1 & \cdots & G_r \end{bmatrix}$$

$$U_{N,M,r} = \begin{bmatrix} u_{N-M} & u_{N-(M-1)} & \cdots u_N & \\ u_{N-(M+1)} & u_{N-M} & \cdots u_{N-1} & \\ \vdots & \vdots & \ddots & \vdots \\ u_{N-(M+r)} & u_{N-(M+r-1)} & \cdots u_{N-r} & \end{bmatrix}$$

I/O identification: $Y_N = GU_{N,M,r} \implies G = Y_N U_{N,M,r}^{\dagger}$

SysID with Known Structure

Quick Review of Basic Realization

Given $G_0, \ldots, G_r$, build Hankel matrix:

$$\mathcal{H}(G) = \begin{bmatrix} G_1 & G_2 & \cdots & G_{r/2} \\ G_2 & G_3 & \iddots & G_{r/2+1} \\ \vdots & \iddots & \ddots & \vdots \\ G_{r/2} & G_{r/2+1} & \cdots & G_r \end{bmatrix}$$

If system order *n* is less than *r* then rank$(H(G))=n$, and (A,C) can be identified via SVD, (B,D) can be identified via least-squares.

SysID with Known Structure

Combine to deal with process and observation noise

$$\begin{array}{ll} \underset{G_0,\ldots,G_r}{\text{minimize}} & \operatorname{rank}(\mathcal{H}(G)) \\ \text{subject to} & \|Y_N - GU_{N,M,r}\|_F^2 \leq \delta^2 \end{array}$$

SysID with Known Structure

Combine to deal with process and observation noise

$$\begin{array}{ll} \underset{G_0,\ldots,G_r}{\text{minimize}} & \operatorname{rank}(\mathcal{H}(G)) \\ \text{subject to} & \|Y_N - GU_{N,M,r}\|_F^2 \leq \delta^2 \end{array}$$

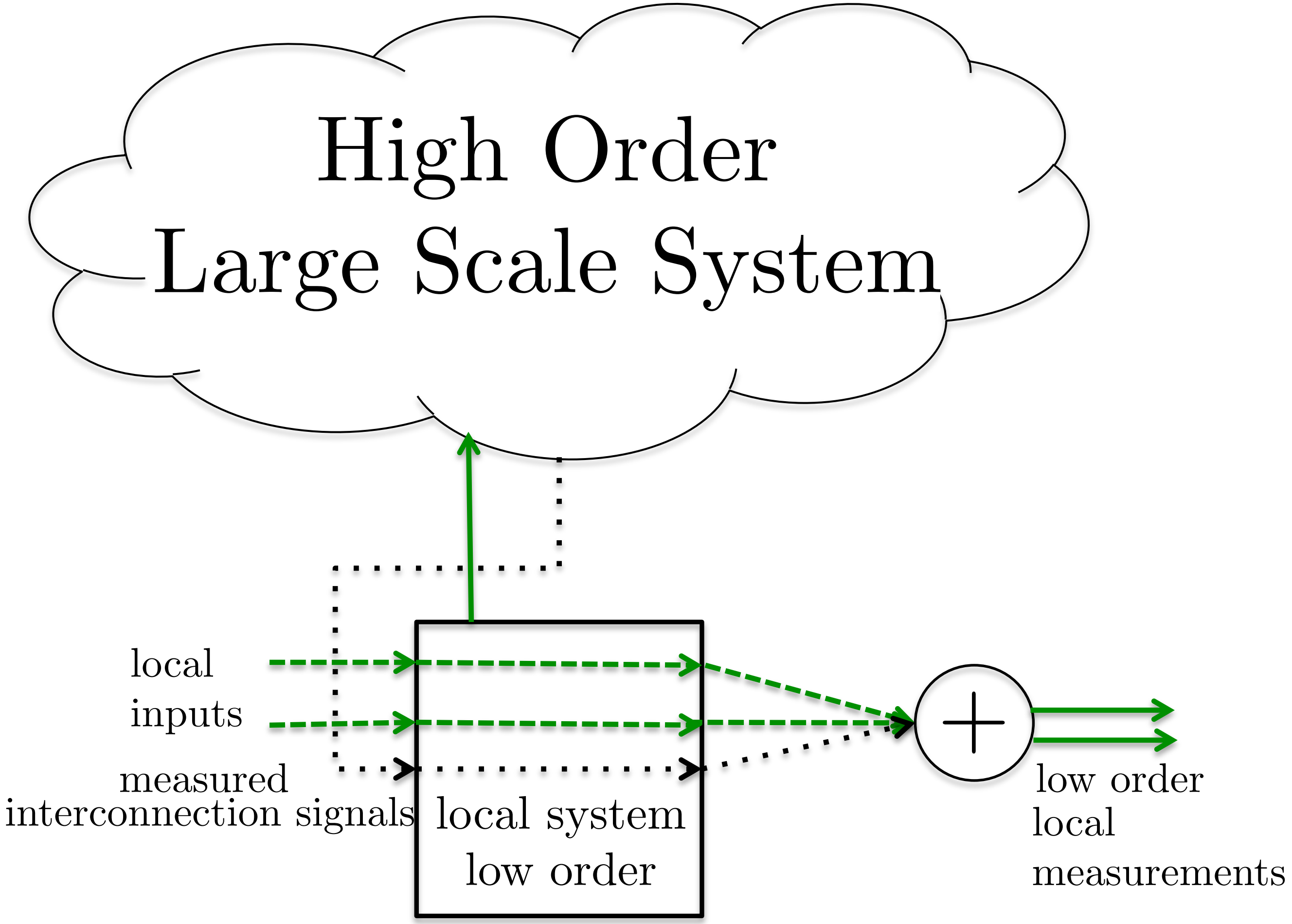
Non-convex!
Relax to

$$\begin{array}{ll} \underset{G_0,\ldots,G_r}{\text{minimize}} & \|\mathcal{H}(G)\|_* \\ \text{subject to} & \|Y_N - GU_{N,M,r}\|_F^2 \leq \delta^2 \end{array}$$

More on why this is the right thing to do later.

SysID with Known Structure

Easy case: we can measure all interconnecting signals

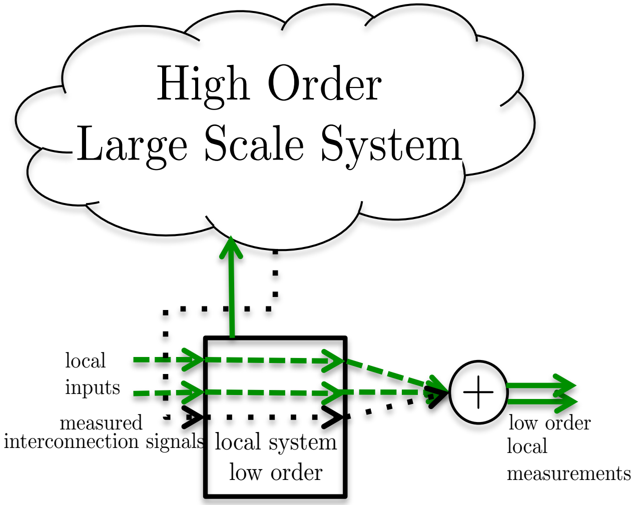


Low-Rank and Low-Order Decompositions for Local System Identification, M. & Rantzer '14

SysID with Known Structure

Easy case: we can measure all interconnecting signals

$$\begin{aligned} \underset{G_0,\ldots,G_r}{\text{minimize}} \quad & \|\mathcal{H}(G)\|_* \\ \text{subject to} \quad & \|Y_N - GU_{N,M,r}\|_F^2 \leq \delta^2 \end{aligned}$$



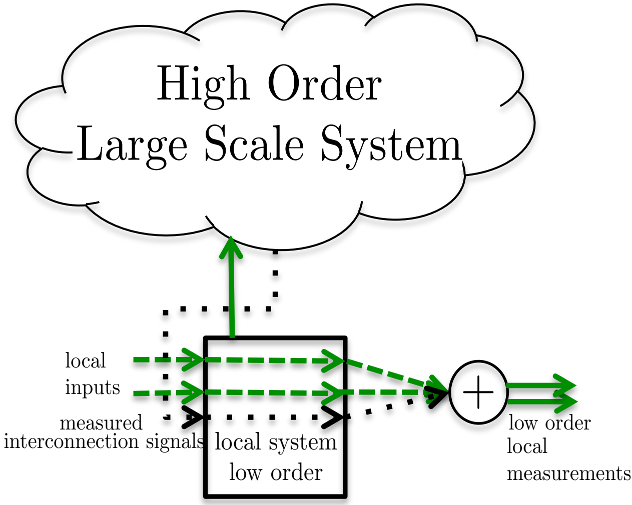
Where now *U* consists of local inputs and measured interconnecting signals.

Low-Rank and Low-Order Decompositions for Local System Identification, M. & Rantzer '14

SysID with Known Structure

Easy case: we can measure all interconnecting signals

$$\begin{array}{ll} \underset{G_0,\ldots,G_r}{\text{minimize}} & \|\mathcal{H}(G)\|_* \\ \text{subject to} & \|Y_N - GU_{N,M,r}\|_F^2 \leq \delta^2 \end{array}$$



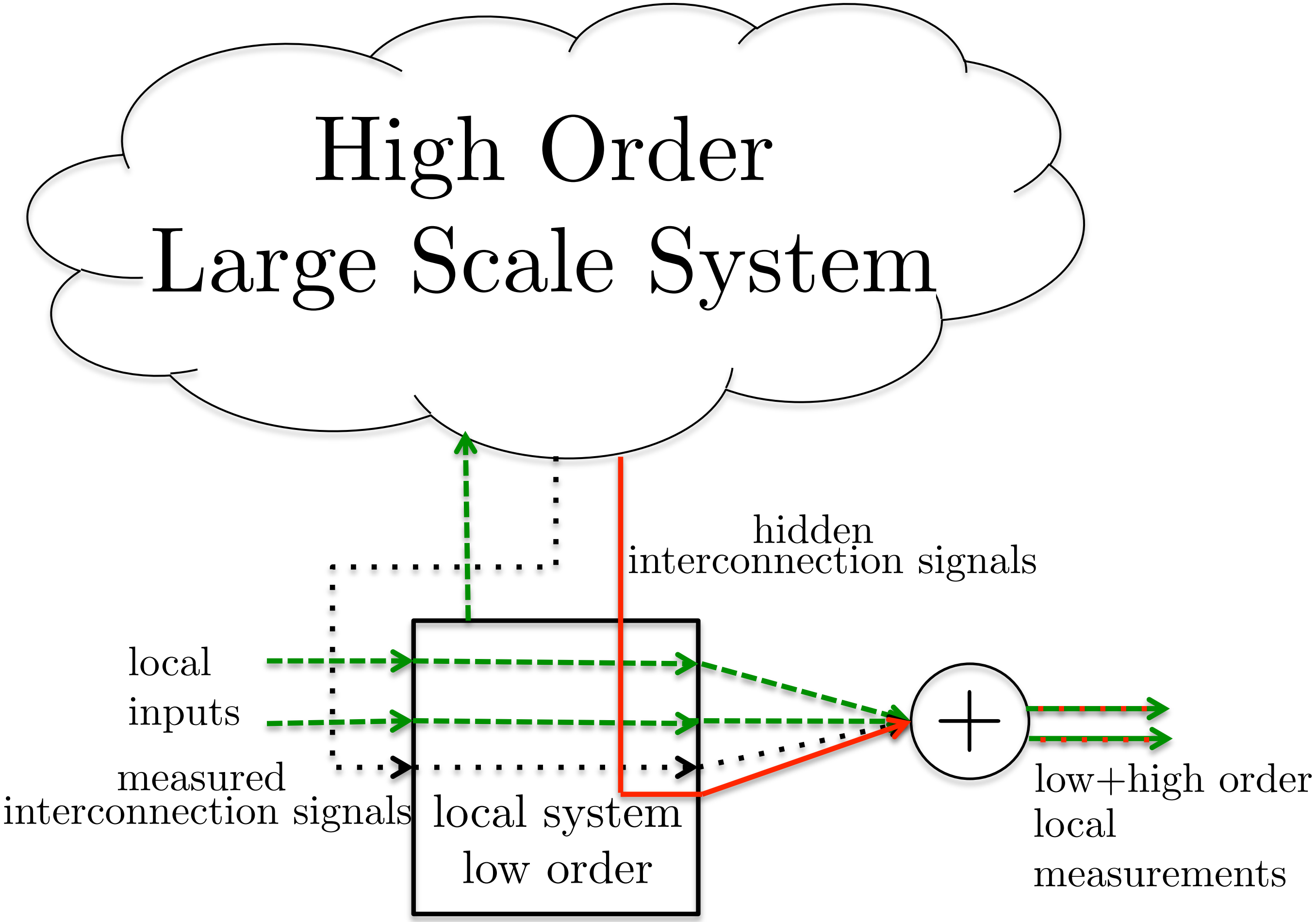
Where now *U* consists of local inputs and measured interconnecting signals.

Need to get neighbors to inject excitation as well.

Low-Rank and Low-Order Decompositions for Local System Identification, M. & Rantzer '14

SysID with Known Structure

Tricky case: we miss some interconnecting signals

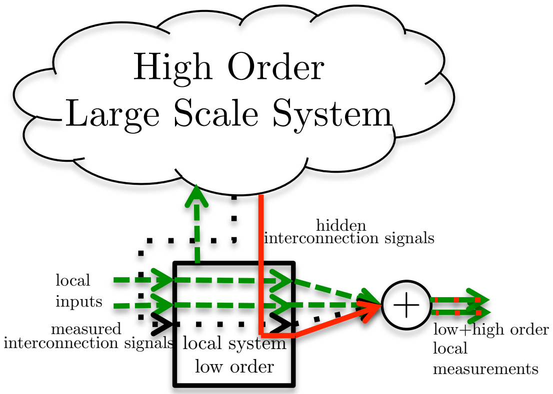


Low-Rank and Low-Order Decompositions for Local System Identification, M. & Rantzer '14

SysID with Known Structure

Tricky case: we miss some interconnecting signals

$$y_t = \sum_{\tau=0}^{t} G_\tau u_{t-\tau} + H_\tau u_{t-\tau}$$



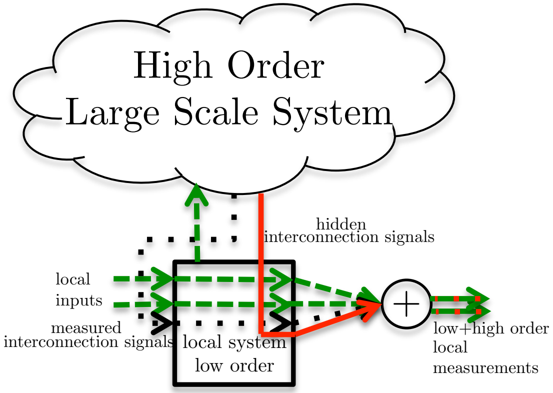
SysID with Known Structure

Tricky case: we miss some interconnecting signals

$$y_t = \sum_{\tau=0}^{t} \boxed{G_\tau u_{t-\tau}} + \boxed{H_\tau u_{t-\tau}}$$

Low-order but full rank

High-order but low rank



Low-Rank and Low-Order Decompositions for Local System Identification, M. & Rantzer '14

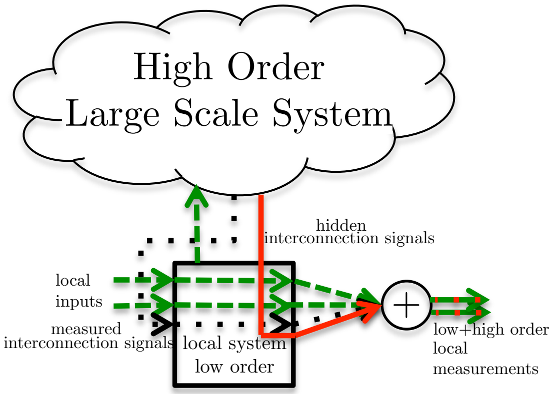
SysID with Known Structure

Tricky case: we miss some interconnecting signals

$$y_t = \sum_{\tau=0}^{t} \boxed{G_\tau u_{t-\tau}} + \boxed{H_\tau u_{t-\tau}}$$

Low-order but full rank

High-order but low rank



Can we separate out the two components?

Low-Rank and Low-Order Decompositions for Local System Identification, M. & Rantzer '14

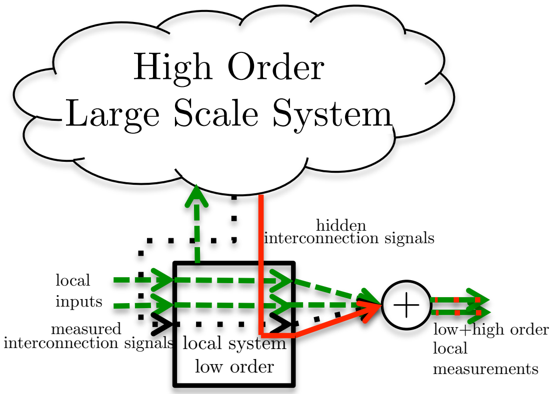
SysID with Known Structure

Tricky case: we miss some interconnecting signals

$$y_t = \sum_{\tau=0}^{t} \boxed{G_\tau u_{t-\tau}} + \boxed{H_\tau u_{t-\tau}}$$

Low-order but full rank

High-order but low rank



Can we separate out the two components?

$$\begin{array}{ll}\underset{\{G_k\},\{H_k\}}{\text{minimize}} & \text{rank}(\mathcal{H}(G)) \\ \text{subject to} & \|Y_N - (G+H)U_{N,M,r}\|_F^2 \le \delta^2 \\ & \text{rank}(H(e^{j\omega})) \le k \end{array}$$

Low-Rank and Low-Order Decompositions for Local System Identification, M. & Rantzer '14

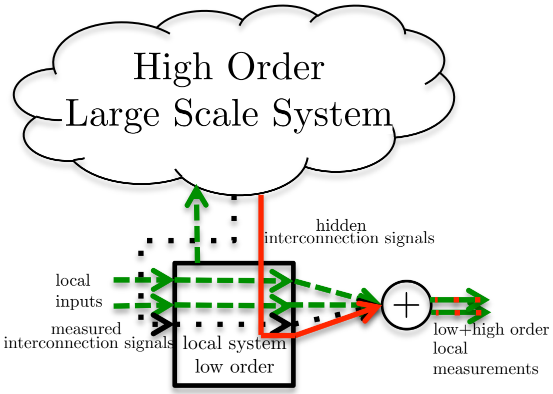
SysID with Known Structure

Tricky case: we miss some interconnecting signals

$$y_t = \sum_{\tau=0}^{t} \boxed{G_\tau u_{t-\tau}} + \boxed{H_\tau u_{t-\tau}}$$

Low-order but full rank

High-order but low rank



Can we separate out the two components?

$$\begin{array}{ll} \underset{\{G_k\},\{H_k\}}{\text{minimize}} & \|\mathcal{H}(G)\|_* \\ \text{subject to} & \|Y_N - (G+H)U_{N,M,r}\|_F^2 \leq \delta^2 \\ & \|H(e^{j\omega})\|_* \leq k \end{array}$$

Low-Rank and Low-Order Decompositions for Local System Identification, M. & Rantzer '14

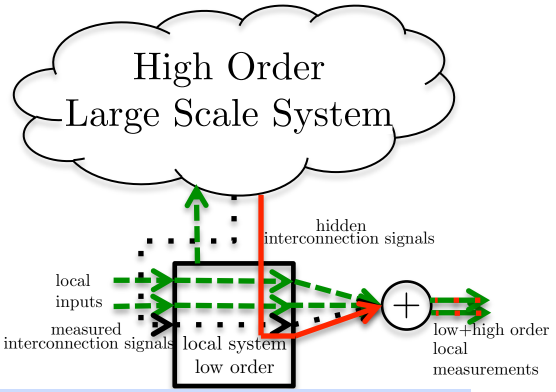
SysID with Unknown Structure

Tricky case: we miss some interconnecting signals

$$y_t = \sum_{\tau=0}^{t} \boxed{G_\tau u_{t-\tau}} + \boxed{H_\tau u_{t-\tau}}$$

Low-order but full rank

High-order but low rank



Key feature:
exploiting structure to de-convolve response

$$\begin{array}{ll} \underset{\{G_k\},\{H_k\}}{\text{minimize}} & \|\mathcal{H}(G))\|_* \\ \text{subject to} & \|Y_N - (G+H)U_{N,M,r}\|_F^2 \leq \delta^2 \\ & \|H(e^{j\omega})\|_* \leq k \end{array}$$

Low-Rank and Low-Order Decompositions for Local System Identification, M. & Rantzer '14

Roadmap for 2nd Part

Networked Control Systems

- Single plant/controller: connections with information theory
- Approaches for extending to distributed control

Varying Delays

- Recent progress

Distributed System Identification

- Known structure
- Unknown structure

Control Architecture Design

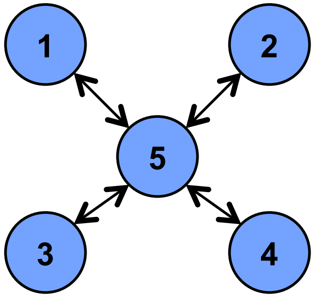
Emphasize Connections to Optimization & Statistics

Latent Variables in Graphical Models

Will consider simpler case of identifying structure in Graphical Models

$$X \sim \mathcal{N}(0, \Sigma)$$

X_i and X_j
independent conditioned
on other vars



Latent Variables in Graphical Models

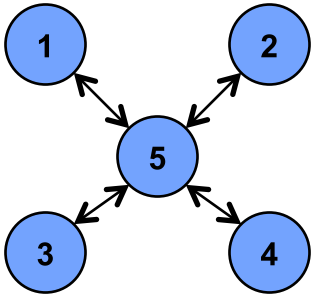
Will consider simpler case of identifying structure in Graphical Models

$$X \sim \mathcal{N}(0, \Sigma)$$

$$X_i \text{ and } X_j$$
$$\text{independent conditioned}$$
$$\text{on other vars}$$

$$\Updownarrow$$

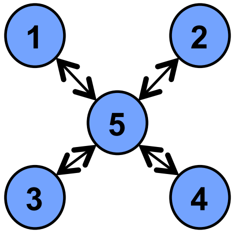
$$(\Sigma^{-1})_{ij} = 0$$



$$\Updownarrow$$

$$\Sigma^{-1} = \begin{bmatrix} * & 0 & 0 & 0 & * \\ 0 & * & 0 & 0 & * \\ 0 & 0 & * & 0 & * \\ 0 & 0 & 0 & * & * \\ * & * & * & * & * \end{bmatrix}$$

Latent Variables in Graphical Models



Traditional estimation procedure

Collect samples $X^1, \ldots, X^N$

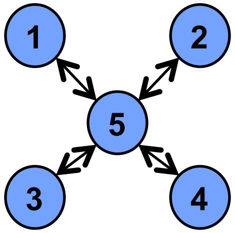
Build sample covariance matrix

$$\hat{\Sigma} = \frac{1}{N}\sum_{i=1}^{N}(X^i)(X^i)^\top$$

For *N*>*n*, sample covariance is invertible.

Threshold $\hat{\Sigma}^{-1}$ to identify structure

Latent Variables in Graphical Models



If we know model is sparse *a priori*

Collect samples $X^1, \ldots, X^N$

Build sample covariance matrix

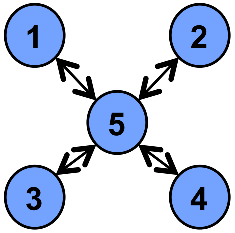
$$\hat{\Sigma} = \frac{1}{N} \sum_{i=1}^{N} (X^i)(X^i)^\top$$

For *N*<*n*, solve

$$\underset{K}{\text{minimize}} \quad \mathrm{Tr}\hat{\Sigma}K - \log\det K + \lambda \|K\|_{\boxed{0}}$$

Non-convex

Latent Variables in Graphical Models



If we know model is sparse *a priori*

Collect samples $X^1, \ldots, X^N$

Build sample covariance matrix

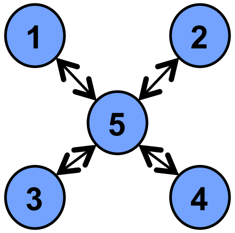
$$\hat{\Sigma} = \frac{1}{N} \sum_{i=1}^{N} (X^i)(X^i)^\top$$

For *N*<*n*, solve

$$\underset{K}{\text{minimize}} \quad \mathrm{Tr}\hat{\Sigma}K - \log\det K + \lambda \|K\|_{\boxed{1}}$$

convex

Latent Variables in Graphical Models



If we know model is sparse *a priori*

Collect samples $X^1, \ldots, X^N$

Build sample covariance matrix

$$\hat{\Sigma} = \frac{1}{N} \sum_{i=1}^{N} (X^i)(X^i)^\top$$

For *N*<*n*, solve

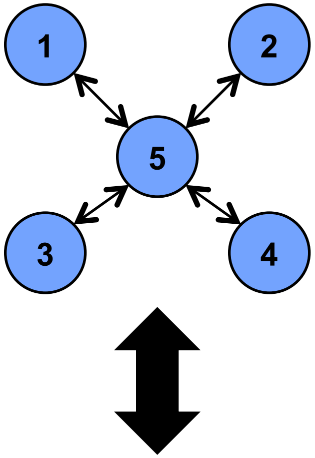
$$\underset{K}{\text{minimize}} \quad \text{Tr}\hat{\Sigma}K - \log\det K + \lambda \|K\|_{\boxed{1}}$$

convex

This works! Banerjee et al. ‘06, Ravikumar et al. ’08, …

Latent Variables in Graphical Models

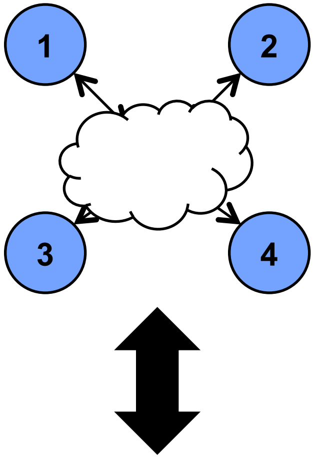
But what if we miss a variable?



$$\Sigma^{-1} = \begin{bmatrix} * & 0 & 0 & 0 & * \\ 0 & * & 0 & 0 & * \\ 0 & 0 & * & 0 & * \\ 0 & 0 & 0 & * & * \\ * & * & * & * & * \end{bmatrix}$$

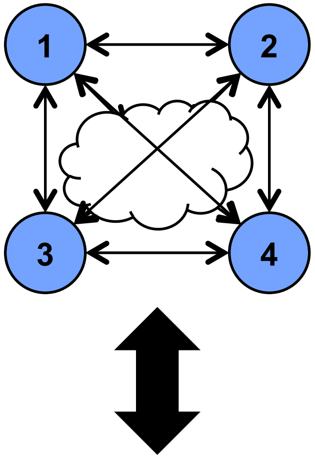
Latent Variables in Graphical Models

But what if we miss a variable?



Latent Variables in Graphical Models

But what if we miss a variable?



$$(\Sigma_O)^{-1} = \begin{bmatrix} * & * & * & * \\ * & * & * & * \\ * & * & * & * \\ * & * & * & * \end{bmatrix}$$

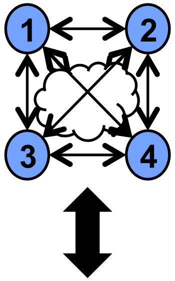
Latent Variables in Graphical Models

But what if we miss a variable?

$$\Sigma = \begin{bmatrix} \Sigma_O & \Sigma_{O,H} \\ \Sigma_{H,O} & \Sigma_{H,H} \end{bmatrix}$$

$$\downarrow$$

$$(\Sigma)^{-1} = K = \begin{bmatrix} K_O & K_{O,H} \\ K_{H,O} & K_{H,H} \end{bmatrix}$$



$$(\Sigma_O)^{-1} = \begin{bmatrix} * & * & * & * \\ * & * & * & * \\ * & * & * & * \\ * & * & * & * \end{bmatrix}$$

Latent Variables in Graphical Models

But what if we miss a variable?

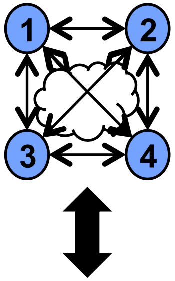
$$\Sigma = \begin{bmatrix} \Sigma_O & \Sigma_{O,H} \\ \Sigma_{H,O} & \Sigma_{H,H} \end{bmatrix}$$

$$(\Sigma)^{-1} = K = \begin{bmatrix} K_O & K_{O,H} \\ K_{H,O} & K_{H,H} \end{bmatrix}$$

$$(\Sigma_O)^{-1} = K_O - K_{O,H} K_H^{-1} KH,O$$

Sparse

Low-rank



$$(\Sigma_O)^{-1} = \begin{bmatrix} * & * & * & * \\ * & * & * & * \\ * & * & * & * \\ * & * & * & * \end{bmatrix}$$

Latent Variables in Graphical Models

But what if we miss a variable?

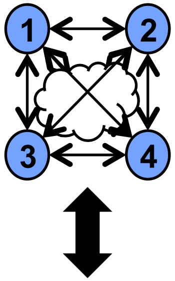
$$\Sigma = \begin{bmatrix} \Sigma_O & \Sigma_{O,H} \\ \Sigma_{H,O} & \Sigma_{H,H} \end{bmatrix}$$

$$\downarrow$$

$$(\Sigma)^{-1} = K = \begin{bmatrix} K_O & K_{O,H} \\ K_{H,O} & K_{H,H} \end{bmatrix}$$

$$\downarrow$$

$$\begin{array}{ll} \underset{S,L}{\text{minimize}} & \mathrm{Tr}\hat{\Sigma}_O(S-L) - \log\det(S-L) + \lambda\|S\|_1 + \gamma\|L\|_* \\ \text{subject to} & S - L \succ 0, L \succeq 0 \end{array}$$



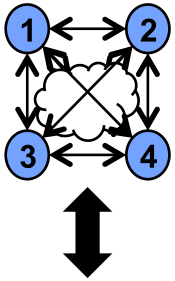
$$(\Sigma_O)^{-1} = \begin{bmatrix} * & * & * & * \\ * & * & * & * \\ * & * & * & * \\ * & * & * & * \end{bmatrix}$$

This works! Chandrasekaran, Parrilo & Willsky '12

Latent Variables in Graphical Models

But what if we miss a variable?

$$\Sigma = \begin{bmatrix} \Sigma_O & \Sigma_{O,H} \\ \Sigma_{H,O} & \Sigma_{H,H} \end{bmatrix}$$



$$(\Sigma_O)^{-1} = \begin{bmatrix} * & * & * & * \\ * & * & * & * \\ & & & * \\ & & & * \end{bmatrix}$$

Key feature:
exploiting structure to de-convolve response

$$\begin{array}{ll} \underset{S,L}{\text{minimize}} & \mathrm{Tr}\hat{\Sigma}_O(S-L) - \log\det(S-L) + \lambda\|S\|_1 + \gamma\|L\|_* \\ \text{subject to} & S - L \succ 0, L \succeq 0 \end{array}$$

This works! Chandrasekaran, Parrilo & Willsky ’12

Roadmap for 2nd Part

Networked Control Systems

- Single plant/controller: connections with information theory
- Approaches for extending to distributed control

Varying Delays

- Recent progress

Distributed System Identification

- Known structure
- Unknown structure

Control Architecture Design

Emphasize Connections to Optimization & Statistics

Control Architecture Design

In SysID, induced structure in solution to identify models

Can we induce structure to ***design*** *control architectures?*

Control Architecture Design

In SysID, induced structure in solution to identify models

Can we induce structure to ***design*** *control architectures?*

Communication Delay Design

&

Actuator placement

Control Architecture Design

In SysID, induced structure in solution to identify models

Can we induce structure to ***design*** *control architectures?*

Communication Delay Design

&

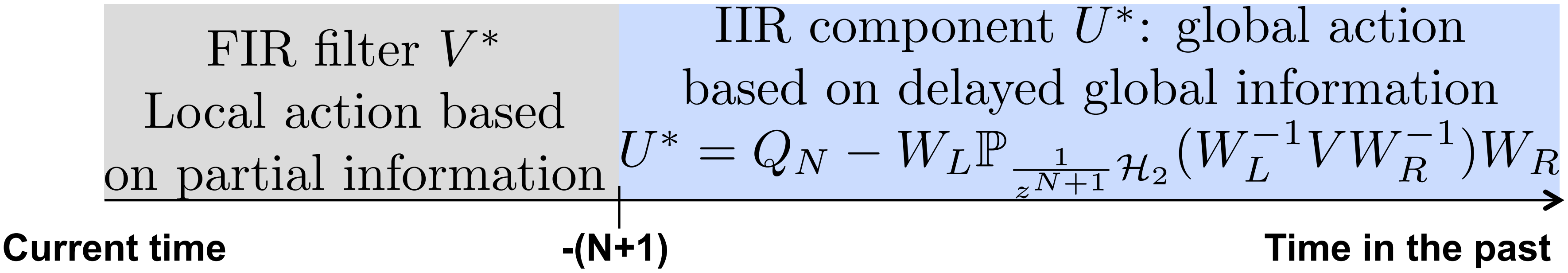
Actuator placement

Key Feature: Convex Co-Design Procedure

Comm Delay Co-Design

$$\text{minimize}_V \sum_{i=1}^{N} \left(\text{Tr} G_i(V) \left(G_i(V)\right)^\top + 2\text{Tr} G_i(V) T_i^\top \right)$$

$$\text{s.t. } V_i \in \mathcal{Y}_i$$



FIR filter V^*
Local action based
on partial information

IIR component U^*: global action
based on delayed global information
$U^* = Q_N - W_L \mathbb{P}_{\frac{1}{z^{N+1}}} \mathcal{H}_2 (W_L^{-1} V W_R^{-1}) W_R$

Comm Delay Co-Design

$$\text{minimize}_V \sum_{i=1}^{N} \left(\text{Tr} G_i(V) \left(G_i(V) \right)^\top + 2 \text{Tr} G_i(V) T_i^\top \right)$$

$$\text{s.t. } V_i \in \mathcal{Y}_i$$

- **Entire** decentralized nature captured in *V*

Comm Delay Co-Design

$$\text{minimize}_V \sum_{i=1}^{N} \left(\mathrm{Tr} G_i(V) \left(G_i(V)\right)^\top + 2 \mathrm{Tr} G_i(V) T_i^\top \right)$$

$$\text{s.t.} \quad \xcancel{V_i \in \mathcal{V}_i}$$

- **Entire** decentralized nature captured in *V*

- Remove constraints

Comm Delay Co-Design

$$\text{minimize}_V \sum_{i=1}^{N} \left(\mathrm{Tr} G_i(V) \left(G_i(V) \right)^\top + 2 \mathrm{Tr} G_i(V) T_i^\top \right) + \lambda \|V\|_{\mathcal{A}}$$

$$\text{s.t.} \quad \xcancel{V_i \in \mathcal{V}_i}$$

- **Entire** decentralized nature captured in *V*
- Remove constraints
- Add penalty to *induce* simple structure

Comm Delay Co-Design

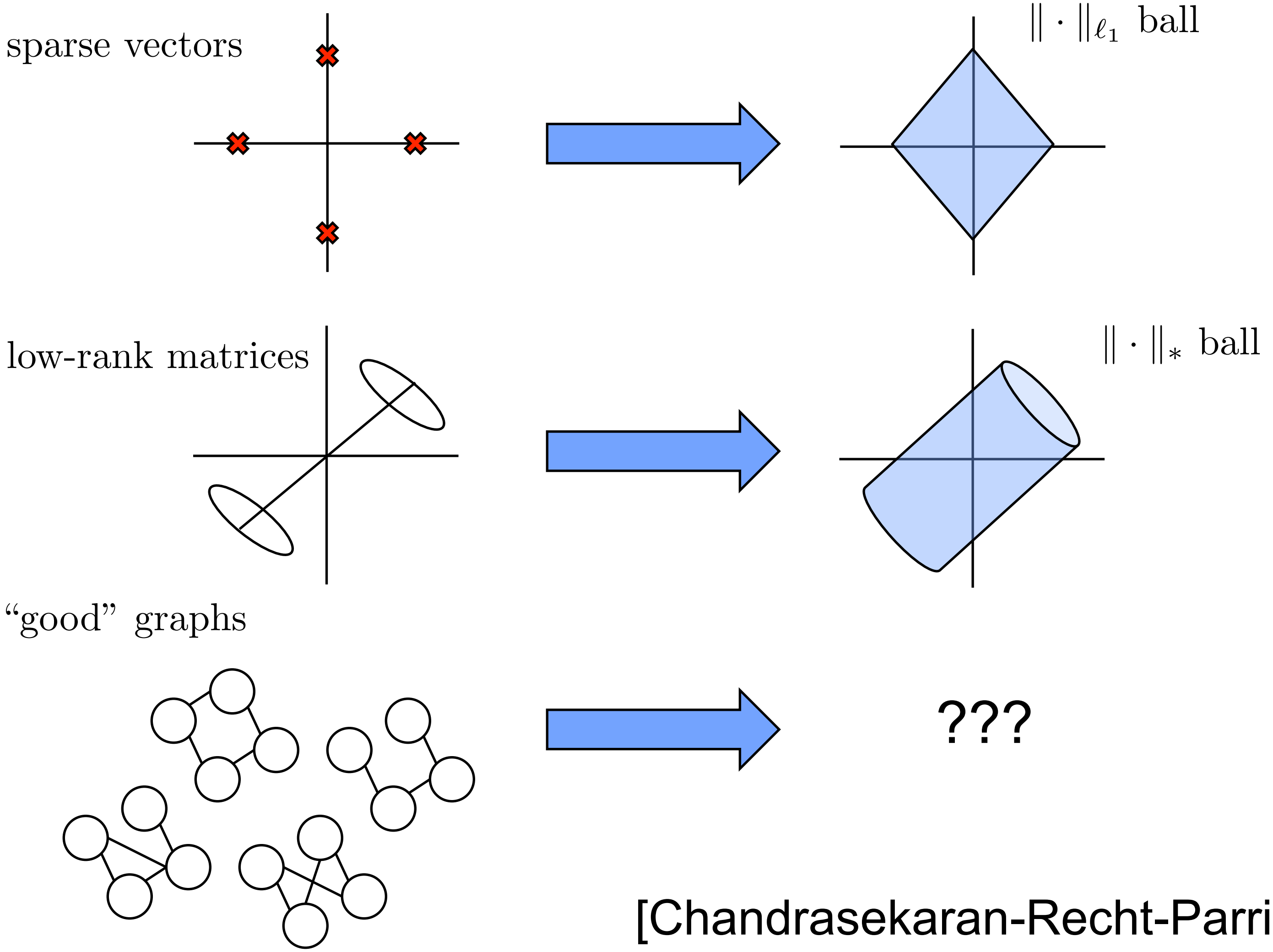
$$\text{minimize}_V \sum_{i=1}^{N} \left(\text{Tr} G_i(V) \left(G_i(V) \right)^\top + 2 \text{Tr} G_i(V) T_i^\top \right) + \lambda \|V\|_{\mathcal{A}}$$

$$\text{s.t.} \quad \xcancel{V_i \in \mathcal{V}_i}$$

- **Entire** decentralized nature captured in *V*
- Remove constraints
- Add penalty to *induce* simple structure
- *What kind of structure in V?*
- *How to induce it in a convex way?*

Main Tool: Atomic Norms

$$\|X\|_{\mathcal{A}} := \inf\{t > 0 \mid X \in t\,\mathrm{conv}(\mathcal{A})\}$$



[Chandrasekaran-Recht-Parrilo-Willsky]

The Graph Enhancement "Norm"

Designed communication graph should

1. Satisfy tractability requirements (QI)
2. Be strongly connected (SC)
3. Be simple
4. Yield acceptable closed loop performance

Insight: Adjacency matrices of graphs satisfying 1 and 2 are closed under addition.

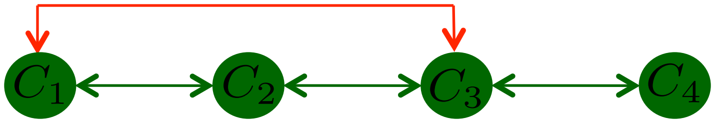
Approach: Minimize structure inducing norm subject to performance constraint

The Graph Enhancement "Norm"

Start with base that is QI and SC

$$\mathcal{S} = \underset{t=-1}{\frac{1}{z}\begin{bmatrix} \ast & 0 & 0 & 0 \\ 0 & \ast & 0 & 0 \\ 0 & 0 & \ast & 0 \\ 0 & 0 & 0 & \ast \end{bmatrix}} \oplus \underset{t=-2}{\frac{1}{z^2}\begin{bmatrix} \ast & \ast & 0 & 0 \\ \ast & \ast & \ast & 0 \\ 0 & \ast & \ast & \ast \\ 0 & 0 & \ast & \ast \end{bmatrix}} \oplus \underset{t=-3}{\frac{1}{z^3}\begin{bmatrix} \ast & \ast & \ast & 0 \\ \ast & \ast & \ast & \ast \\ \ast & \ast & \ast & \ast \\ 0 & \ast & \ast & \ast \end{bmatrix}} \oplus \frac{1}{z^4}\mathcal{H}_2$$

Add shortcuts



$$\mathcal{S} = \underset{t=-1}{\frac{1}{z}\begin{bmatrix} \ast & 0 & 0 & 0 \\ 0 & \ast & 0 & 0 \\ 0 & 0 & \ast & 0 \\ 0 & 0 & 0 & \ast \end{bmatrix}} \oplus \underset{t=-2}{\frac{1}{z^2}\begin{bmatrix} \ast & \ast & \ast & 0 \\ \ast & \ast & \ast & 0 \\ \ast & \ast & \ast & \ast \\ 0 & 0 & \ast & \ast \end{bmatrix}} \oplus \underset{t=-3}{\frac{1}{z^3}\begin{bmatrix} \ast & \ast & \ast & \ast \\ \ast & \ast & \ast & \ast \\ \ast & \ast & \ast & \ast \\ \ast & \ast & \ast & \ast \end{bmatrix}} \oplus \frac{1}{z^4}\mathcal{H}_2$$

Project out base

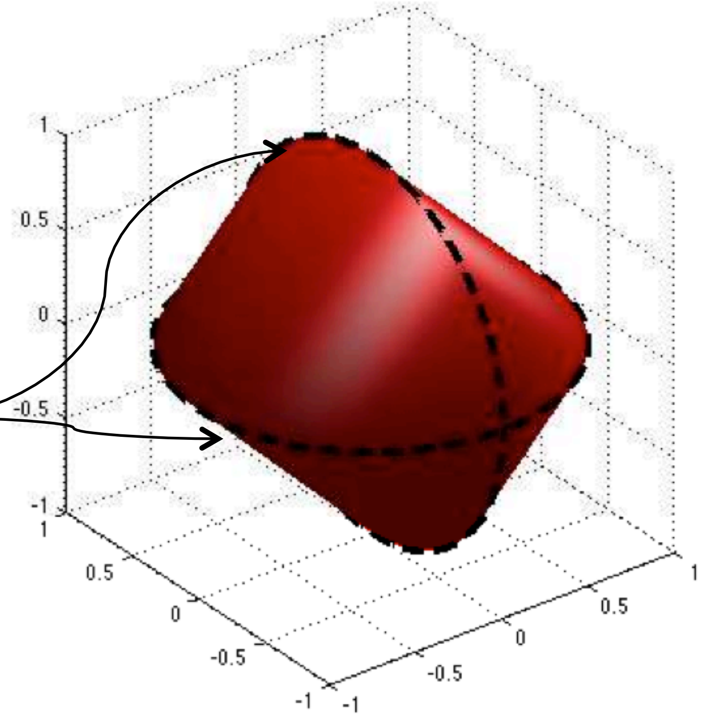
$$a_{13} = \frac{1}{z}\begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} \oplus \frac{1}{z^2}\begin{bmatrix} 0 & 0 & \ast & 0 \\ 0 & 0 & 0 & 0 \\ \ast & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} \oplus \frac{1}{z^3}\begin{bmatrix} 0 & 0 & 0 & \ast \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ \ast & 0 & 0 & 0 \end{bmatrix}$$

The Graph Enhancement "Norm"

Special case of group norm with overlap [Jacob-Obozinski-Vert]

$$
\begin{aligned}
||x||_{\mathcal{A}} = \min_{x_1,x_2} & \, ||x_1||_2 + ||x_2||_2 \\
\text{subject to} & \\
& \textstyle\sum x_i = x \\
& \operatorname{supp}(x_i) \subset \operatorname{supp}(a_i)
\end{aligned}
$$

$$\mathcal{A} = \{[*,*,0],\ [0,*,*]\}$$



Convex hull of
low dimensional unit disks

Communication Delay Co-Design

Theorem [N.M. CDC ‘13, TCNS ’14]

Solving

Designed norm

Centralized norm

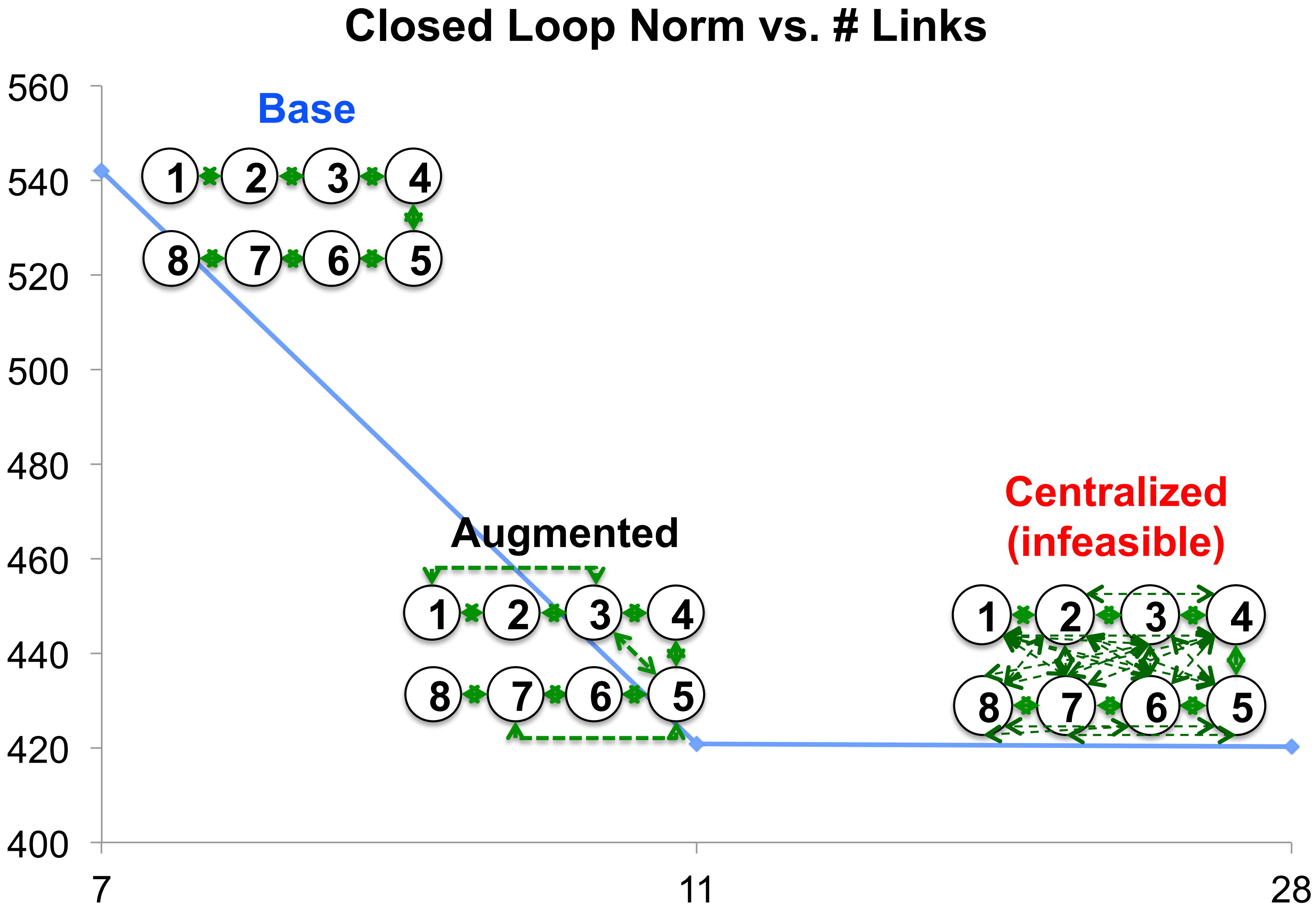
$$\begin{aligned} \text{minimize}_Q \quad & \|Q\|_{\mathcal{A}} \\ \text{s.t.} \quad & \boxed{N(Q)^2} - \boxed{N_c^2} \leq \boxed{\delta^2} \end{aligned}$$

Tuning param

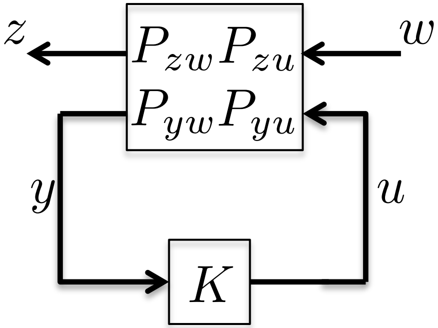
yields a “simple” SC and QI communication graph satisfying *a priori* performance bounds.

Proof is a synthesis of results from Lamperski & Doyle ’12; Rotkowitz, Cogill & Lall ’10; and Chandrasekaran et al. ’12.

Communication Delay Co-Design



Actuator Regularization



Goal

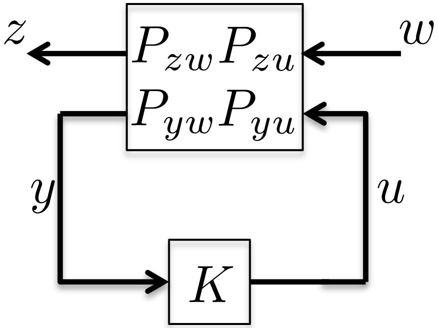
Choose which actuators we need

Approach

Assume *B* is block-diagonal.

$$\text{minimize}_Q \|P_{zw} + P_{zu} Q P_{yw}\|$$
$$\text{s.t. } Q \text{ stable \& causal}$$

Actuator Regularization



Goal

Choose which actuators we need

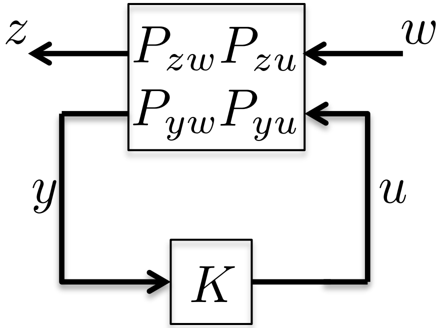
Approach

Assume *B* is block-diagonal.

$$\text{minimize}_Q \|P_{zw} + P_{zu} Q P_{yw}\|$$
$$\text{s.t. } Q \text{ stable \& causal}$$

Then each block-row of *Q* corresponds to an actuator.

Actuator Regularization



Goal

Choose which actuators we need

Approach

Assume *B* is block-diagonal.

$$\text{minimize}_Q \|P_{zw} + P_{zu} Q P_{yw}\|$$
$$\text{s.t. } Q \text{ stable \& causal}$$

Then each block-row of *Q* corresponds to an actuator.

Atoms are controllers with one non-zero block-row.

Leads to “group norm without overlap”

Other Application Areas

Sparse static feedback design

A scalable formulation for engineering combination therapies for evolutionary dynamics of disease, Jonsson, Rantzer, Murray, ACC ‘14

Sparsity-promoting optimal control for a class of distributed systems, Fardad, Lin & Jovanovic ACC ‘11

Design of optimal sparse feedback gains via the alternating direction method of multipliers, Lin, Fardad & Jovanovic TAC ’13

Sparse consensus

On identifying sparse representations of consensus networks, Dhingra, Lin, Fardad, and Jovanovic, IFAC DENCS ’13

Fast linear iterations for distributed averaging, Xiao, Boyd SCL ‘04

Sparse synchronization

Design of optimal sparse interconnection graphs for synchronization of oscillator networks, Fardad, Lin, and Jovanovic, TAC ‘13 (Submitted)

Roadmap for 2nd Part

Networked Control Systems

- Single plant/controller: connections with information theory
- Approaches for extending to distributed control

Varying Delays

- Recent progress

Distributed System Identification

- Known structure
- Unknown structure

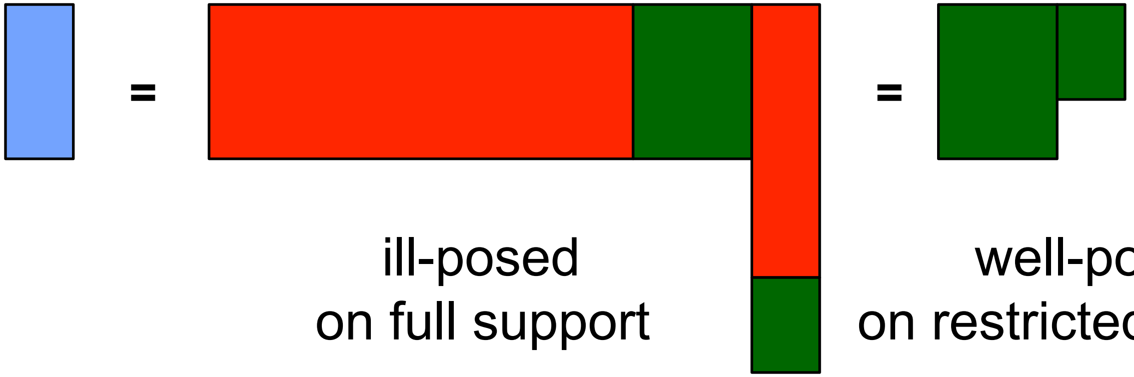
Control Architecture Design

Emphasize Connections to Optimization & Statistics

Regularization: A Success Story

- Regularization incredibly successful in model/system identification
  - Basis pursuit [e.g. Donoho, Candes-Romberg-Tao, Tropp]
  - Matrix completion [e.g. Candes-Recht, Recht-Fazel-Parrilo]
  - Statistical regression [e.g. Wainwright, Ravikumar]
  - System identification [e.g. Shah et al., Ljung]

- Common theme: exploit *structure* and “restricted well-posedness” to solve hard problems using *convex methods.*



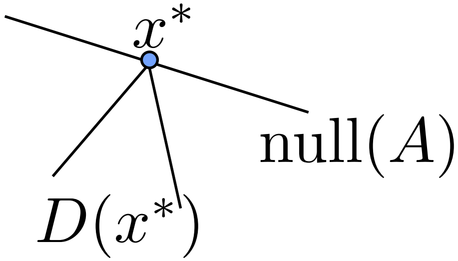
ill-posed
on full support

well-posed
on restricted support

Regularization in Inference/Model Selection

Inference/reconstruction $y = Ax^*(+\epsilon)$

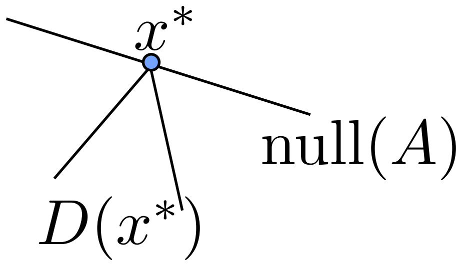
- Minimum restricted gains, null space conditions (Gordon's escape through a mesh, Vershynin, Chadrasekaran et al., Tropp)
- Gives exact reconstruction conditions for no noise
- Estimation bounds for noisy case (no structure)



Regularization in Inference/Model Selection

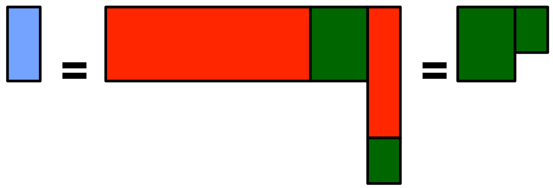
Inference/reconstruction $y = Ax^*(+\epsilon)$

- Minimum restricted gains, null space conditions (Gordon's escape through a mesh, Vershynin, Chadrasekaran et al., Tropp)
- Gives exact reconstruction conditions for no noise
- Estimation bounds for noisy case (no structure)



Primal/Dual Certificates

- Use an "oracle", and show that oracle solution solves original problem
- Still based on restricted gains
- Provides estimation bounds and **structure**



Regularization for Design

$$\text{minimize}_x \quad \|C(x,y)\| + \lambda \|x\|_{\mathcal{A}}$$

	Regularized Distributed Control	**Model/System Identification**
Priors	“Base” controller structure	Simple structure
Structure	Need to **design** subspace	Need to **identify** subspace
Computation	Convex optimization	Convex optimization
Cost	**Closed-loop performance**	**Estimation/ prediction error**
Design Product	Optimal **controller** and **control architecture**	Optimal **estimate** and/or **predictor**

Regularization for Design

So far:

Principled algorithmic connections

- Illustrated with co-design of communication topologies well suited to distributed control

Our goal now:

Theoretical connections

- Define and provide co-design approximation guarantees

How do we measure success?

For estimation/identification

measured in terms of **estimation and/or predictive** power

For design

measured in terms of **structure and approximation** quality

To make things concrete, consider square loss and group norm

$$\mathrm{minimize}_v \, \frac{1}{2}\|y - \mathcal{L}\mathcal{E}_{\mathcal{G}}(v)\|_F^2 + \lambda\|v\|_{\mathcal{G}}$$

Performance

Open loop system

Simplicity

The Group Norm

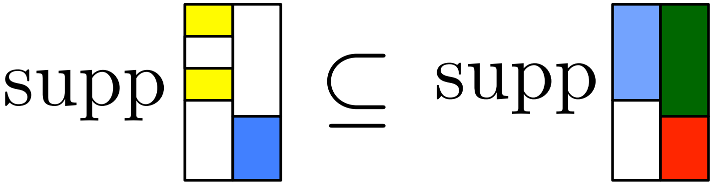
$$\left\| \begin{matrix} v_1 \\ v_2 \\ v_3 \\ v_4 \end{matrix} \right\|_{\mathcal{G}} = \left\| v_1 \right\| + \left\| v_2 \right\| + \left\| v_3 \right\| + \left\| v_4 \right\|$$

With dual norm

$$\left\| \begin{matrix} v_1 \\ v_2 \\ v_3 \\ v_4 \end{matrix} \right\|_{\mathcal{G},\infty} = \max \left\{ \left\| v_1 \right\|, \left\| v_2 \right\|, \left\| v_3 \right\|, \left\| v_4 \right\| \right\}$$

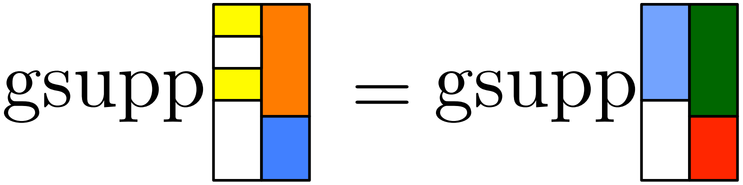
Focus on Structure

$\mathcal{E}_{\mathcal{G}}$-support accurate



Recover a subset of the structure

$\mathcal{G}$-support accurate



Recover full structure

Accurate Approximations

$$\text{minimize}_v \, \frac{1}{2}\|y - \mathcal{L}\mathcal{E}_{\mathcal{G}}(v)\|_F^2 + \lambda\|v\|_{\mathcal{G}}$$

Assume: $$y = \mathcal{L}\mathcal{E}_{\mathcal{G}}(v^*) + \epsilon$$

Sparse nominal controller

Nominal closed loop

$$\text{Self-incoherence: minimum gain of } \mathcal{L} \text{ on } \mathcal{G}^* \geq \alpha$$

$$\text{Corss-incoherence: maximum gain of } \mathcal{L} \text{ from } (\mathcal{G}^*)^{\perp} \to \mathcal{G}^* \leq \gamma$$

$$\boxed{\frac{\gamma}{\alpha} \leq \nu}$$

Total Incoherence

Support Accurate Approximations

Theorem [N.M. and V. Chandrasekaran, CDC '14]

Suppose previous assumptions hold, and $\|\mathcal{E}_{\mathcal{G}}^{+}\mathcal{L}^{+}\epsilon\|_{\mathcal{G},\infty} \leq (\kappa - 1)\lambda$ for some $1 \leq \kappa < \frac{2}{(\nu+1)}$.

Then

Closed loop performance affects approximation error

1. The solution $\hat{v}$ is $\mathcal{E}_{\mathcal{G}}$-support accurate, and
2. $\|\hat{v} - v^*\|_{\mathcal{G},\infty} \leq \lambda\left(\frac{\kappa}{\alpha}\right)$

Corollary

If $\boxed{\|v_g^*\| > \lambda\left(\frac{\kappa}{\alpha}\right)}$ for all $g \in \mathcal{G}^*$. Then $\hat{v}$ is $\mathcal{G}$-support accurate.

only recover dominant control components

Support Accurate Approximations

Theorem [N.M. and V. Chandrasekaran, CDC '14]

Suppose previous assumptions hold, and $\|\mathcal{E}_{\mathcal{G}}^{+}\mathcal{L}^{+}\epsilon\|_{\mathcal{G},\infty} \leq (\kappa - 1)\lambda$ for some $1 \leq \kappa < \frac{2}{(\nu+1)}$.

Then

1. The solution $\hat{v}$ is $\mathcal{E}_{\mathcal{G}}$-support accurate, and
2. $\|\hat{v} - v^*\|_{\mathcal{G},\infty} \leq \lambda\left(\frac{\kappa}{\alpha}\right)$

Closed loop performance affects approximation error

And which controller components we are able to identify

Corollary

If $\boxed{\|v_g^*\| > \lambda\left(\frac{\kappa}{\alpha}\right)}$ for all $g \in \mathcal{G}^*$. Then $\hat{v}$ is $\mathcal{G}$-support accurate.

only recover dominant control components

Support Accurate Approximations

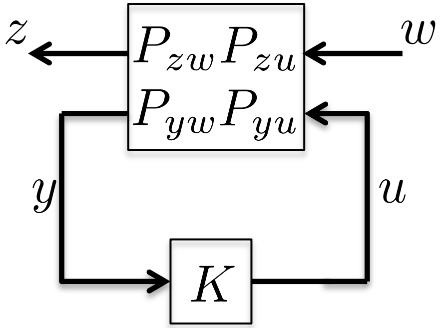
In co-design problems, **closed loop norm** plays the role of **estimation noise** in identification problems

Support Accurate Approximations

In co-design problems, **closed loop norm** plays the role of **estimation noise** in identification problems

Within each class of *k*-sparse controllers the controller leading to **best performance** is **easiest** to identify via **convex** programming

Actuator Regularization



Goal

Choose which actuators we need

Approach

Under mild assumptions each row of *Q* corresponds to an actuator

$$\text{minimize}_Q \|P_{zw} + P_{zu} Q P_{yw}\|$$
$$\text{s.t. } Q \text{ stable \& causal}$$

To make finite dimensional, set a horizon *T* and order *N*

$$\text{minimize}_v \frac{1}{2}\|y - \mathcal{L}\mathcal{E}_{\mathcal{G}}(v)\|_F^2 + \lambda \|v\|_{\mathcal{G}}$$

Performance

Simplicity

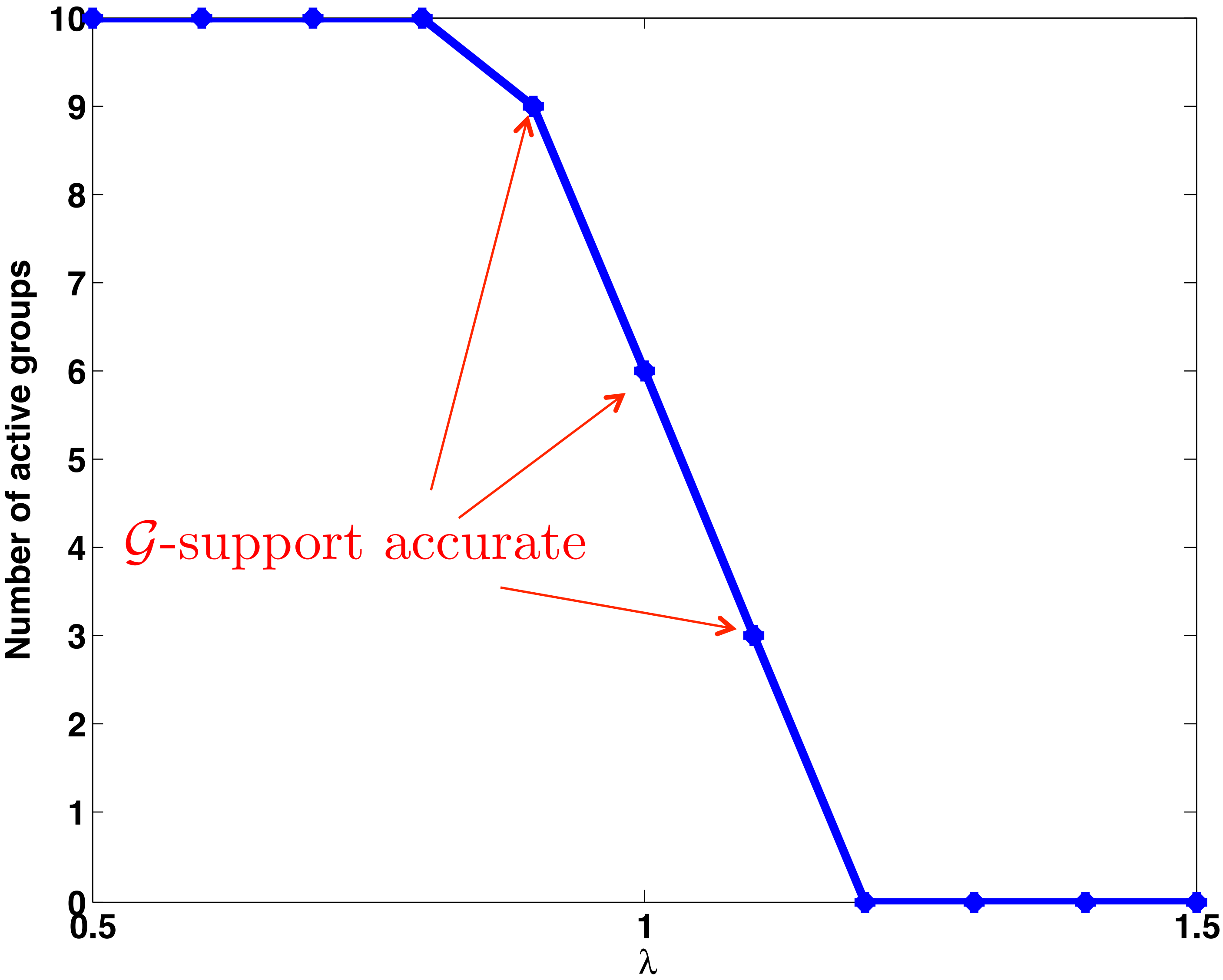
$$y = \mathcal{L}\mathcal{E}_{\mathcal{G}}(v^*) + \epsilon$$

Sparse nominal controller

Nominal closed loop

Actuator Regularization: Sample Path

T =20, N=3, #inputs = 10, #outputs = 10, #states = 10



Incoherence Assumptions

Are these realistic?

Incoherence Assumptions

Are these realistic?

Do not have good theory yet

Incoherence Assumptions

Are these realistic?

Do not have good theory yet

Structure & Stability Help

Banded matrices, Spatially decaying impulse responses, etc.

Incoherence Assumptions

Are these realistic?

Do not have good theory yet

Structure & Stability Help

Banded matrices, Spatially decaying impulse responses, Toeplitz “sensing” matrices, etc.

Randomization Helps

Homogenous systems a simplifying assumption

Incoherence Assumptions

Are these realistic?

Do not have good theory yet

Structure & Stability Help

Banded matrices, Spatially decaying impulse responses, Toeplitz “sensing” matrices, etc.

Randomization Helps

Homogenous systems a simplifying assumption

Overly conservative?

Gains restricted to cones instead of subspaces?

Roadmap for 2^{nd} Part

Networked Control Systems

- Single plant/controller: connections with information theory
- Approaches for extending to distributed control

Varying Delays

- Recent progress

Distributed System Identification

- Known structure
- Unknown structure

Control Architecture Design

Emphasize Connections to Optimization & Statistics

Recap of 2nd Part

Networked Control Systems & Varying Delays

- Connections with information theory
- Assume channels manifest themselves as varying delays

Distributed System Identification & Control Architecture Design

- When nothing is hidden, not too tough
- Hidden variables lead to de-convolution problems: we have good convex methods

Control Architecture Design

- Inherently combinatorial problem can be addressed using ideas from structured identification
- Deeper theoretical connections: estimation noise = closed loop

Going Forward

Integration

- Layering as optimization decomposition, Chiang, Low, Calderbank & Doyle ‘07

Adapt our expectations

- Results that are not scalable to implement: fundamental limits
- Identify new metrics that lead to scalable architectures that approximate these fundamental limits

Combine control, optimization and statistics

- All different sides of the same coin (simplex?)
- Principled theory for analysis and design of large-scale systems no longer out of our reach
- An exciting time to be in CDS + CMS!

Thank you!!!

We will post slides and reference list on workshop website and at

http://www.cds.caltech.edu/~nmatni

Questions?