A CDS+CMS perspective on recent results in distributed control

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

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)

Different Reasonable Problem Sizes

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

Expressivity

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…
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 $\rightarrow$ Structured

takes

Hard problems $\rightarrow$ Easy problems

Main Theme of 1\textsuperscript{st} Part:
Use Structure to Relax
Roadmap for 1\textsuperscript{st} 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 2\textsuperscript{nd} Part

Break
Kirchhoff gives

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

13

Case Study: DC OPF

Case Study: DC OPF

The DC OPF problem is

\[
\begin{align*}
\text{minimize} & \quad \sum_{j=1}^{N} I_j V_j \\
\text{subject to} & \quad I = YV \\
& \quad V_k I_k \leq P_k, \quad V_k^{\text{min}} \leq V_k \leq V_k^{\text{max}} \\
& \quad y_{jk} (V_k - V_j)^2 \leq L_{jk} \\
& \quad \text{for all } j, k = 1, \ldots, N
\end{align*}
\]

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

Indefinite Quadratic Objectives and Constraints $\rightarrow$ Non-Convex
The DC OPF problem is of the form

\[
\begin{align*}
\text{maximize} & \quad x^\top M_0 x \\
\text{subject to} & \quad x^\top M_k x \geq b_k \\
& \quad \text{for } k = 1, \ldots, K
\end{align*}
\]

\textbf{Indefinite Quadratic Objectives and Constraints }\Rightarrow \textbf{Non-Convex}
Case Study: DC OPF

The DC OPF problem is of the form

\[
\begin{align*}
\text{maximize} \quad & x^\top M_0 x \\
\text{subject to} \quad & x^\top M_k x \geq b_k \\
& \text{for } k = 1, \ldots, K
\end{align*}
\]

**Indefinite Quadratic Objectives and Constraints \rightarrow Non-Convex**

In general, NP-Hard
The DC OPF problem is of the form

\[
\begin{align*}
\text{maximize} & \quad x^\top M_0 x \\
\text{subject to} & \quad x^\top M_k x \geq b_k \\
& \text{for } k = 1, \ldots, K
\end{align*}
\]

**Indefinite Quadratic Objectives and Constraints \rightarrow 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*
The DC OPF problem is of the form

\[
\begin{align*}
\text{maximize} & \quad x^\top M_0 x \\
\text{subject to} & \quad x^\top M_k x \geq b_k \\
& \quad \text{for } k = 1, \ldots, K
\end{align*}
\]

\[
\begin{align*}
\text{maximize} & \quad \text{Tr} M_0 X \\
\text{subject to} & \quad \text{Tr} M_k X \geq b_k \\
& \quad \text{for } k = 1, \ldots, K \\
& \quad \text{rank}(X) = 1
\end{align*}
\]

Still non-convex
Case Study: DC OPF

The DC OPF problem is of the form

\[
\begin{align*}
\text{maximize} & \quad x^\top M_0 x \\
\text{subject to} & \quad x^\top M_k x \geq b_k \\
& \quad \text{for } k = 1, \ldots, K
\end{align*}
\]

Convex!

But are we solving the same problem?

\[
\begin{align*}
\text{maximize} & \quad \text{Tr}\, M_0 X \\
\text{subject to} & \quad \text{Tr}\, M_k X \geq b_k \\
& \quad \text{for } k = 1, \ldots, K \\
& \quad \text{rank}(X) = 1
\end{align*}
\]
Case Study: DC OPF

The DC OPF problem is of the form

\[
\begin{align*}
\text{maximize} & \quad \text{Tr} M_0 X \\
\text{subject to} & \quad \text{Tr} M_k X \geq b_k \\
& \quad \text{for } k = 1, \ldots, K \\
& \quad \text{rank}(X) = 1
\end{align*}
\]

We are! Relaxation exact because of Metzler constraints
Case Study: DC OPF

The DC OPF problem is of the form

$$\begin{align*}
\text{maximize} & \quad \text{Tr}M_0X \\
\text{subject to} & \quad \text{Tr}M_kX \geq b_k \\
& \quad \text{for } k = 1, \ldots, K \\
& \quad \text{rank}(X) = 1
\end{align*}$$

We are! Relaxation exact because of Metzler constraints

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

$$\begin{align*}
x_{ii} & \geq 0 \\
x_{ij} & \leq \sqrt{x_{ii}x_{jj}}
\end{align*}$$

Let $x = (\sqrt{x_{ii}})$. Then $(xx^\top)_{ii} = X_{ii}$, but $(xx^\top)_{ij} = \sqrt{x_{ii}x_{jj}} \geq X_{ij}$. Then $x^\top M_kx \geq \text{Tr}M_kX$ because $M_k$ are Metzler.
Aside: Positive Systems Theory

Dynamical system

\[ \dot{x} = Ax \]

Suppose \( A \) is Metzler. Then:

\[ x(0) \in \mathbb{R}_+ \implies x(t) \in \mathbb{R}_+ \quad \forall t \geq 0 \]

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

\[ A \xi < 0 \quad A^T P + PA < 0 \quad A^T z < 0 \]

Aside: Duality and Relaxations

Lagrangian of original problem:

\[
L(x, \lambda_k) = x^\top M_0 x + \sum_{k=1}^{K} \lambda_k \left( x^\top M_k x - b_k \right)
= - \sum_{k=1}^{K} \lambda_k b_k + x^\top \left( M_0 + \sum_{k=1}^{K} \lambda_k M_k \right) x
\]

Dual:

\[
\begin{align*}
\text{minimize} & \quad - \sum_{k=1}^{K} \lambda_k b_k \\
\text{subject to} & \quad M_0 + \sum_{k=1}^{K} \lambda_k M_k \preceq 0
\end{align*}
\]

Dual of dual:

\[
\begin{align*}
\text{maximize} & \quad \text{Tr} M_0 X \\
\text{subject to} & \quad \text{Tr} M_k X \geq b_k \\
& \quad \text{for } k = 1, \ldots, K
\end{align*}
\]
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

If \( p(x) = \sum q(x)^2 \) then \( p(x) \geq 0 \)
Aside: SOS Optimization

Computational test for SOS is a semi-definite program.

For simplicity, fix $d=1$. Then

$$ p(x) = \begin{bmatrix} 1 \\ x_1 \\ \vdots \\ x_n \end{bmatrix}^\top Q \begin{bmatrix} 1 \\ x_1 \\ \vdots \\ x_n \end{bmatrix} $$

is SOS if and only if $Q \succeq 0$

Coefficients of $p(x)$ impose affine constraints on $Q$. 
Aside: SOS Optimization

Constrained polynomial optimization

\[
\max p(x) \quad \text{s.t.} \quad g_i(x) \geq 0
\]

Relax to

\[
\min \gamma \quad \text{s.t.} \quad \gamma - p(x) = s_0(x) + \sum_i s_i(x) g_i(x)
\]

\[
s_0(x), s_i(x) \text{ are } SOS(2d)
\]

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:

\[
\begin{align*}
\min \gamma & \quad \text{s.t.} \quad \gamma - x^\top M_0 x = s_0(x) + \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}, \quad Q_k \succeq 0
\end{align*}
\]

Expand RHS and equate coefficients

\[
\gamma = Q_0^{11} - \sum_{k=1}^K Q_k^{11} b_k, \quad Q_k^{1,j} = 0 \text{ for all } j \neq 1.
\]

For \( k \geq 1 \), \( Q_k^{i,j} = 0 \) for all \( i, j \neq 1 \)

\[
-M_0 = Q_0^{2:n+1,2:n+1} + \sum_{k=1}^K Q_k^{11} M_k
\]
Aside: SOS Optimization

SOS relaxation of original problem:

\[
\begin{align*}
\text{minimize} & \quad Q_{0}^{11} - \sum_{k=1}^{K} Q_{k}^{11} b_{k} \\
\text{subject to} & \quad \sum_{k=1}^{K} Q_{k}^{11} M_{k} + M_{0} = -Q \\
& \quad Q_{k}^{11} \geq 0, Q \geq 0
\end{align*}
\]
Aside: SOS Optimization

SOS relaxation of original problem:

\[
\begin{align*}
\text{minimize} & \quad Q_0^{11} - \sum_{k=1}^{K} Q_k^{11} b_k \\
Q_{k}^{11} & \geq 0, Q \geq 0 \\
\text{subject to} \quad & \sum_{k=1}^{K} Q_k^{11} M_k + M_0 = -Q \\
\end{align*}
\]

\[
\begin{align*}
\text{minimize} & \quad -\sum_{k=1}^{K} Q_k^{11} b_k \\
Q_{k}^{11} & \geq 0, Q \geq 0 \\
\text{subject to} \quad & \sum_{k=1}^{K} Q_k^{11} M_k + M_0 \leq 0 \\
\end{align*}
\]
Aside: SOS Optimization

SOS relaxation of original problem:

\[
\begin{align*}
\text{minimize} & \quad Q^{11}_0 - \sum_{k=1}^{K} Q^{11}_k b_k \\
\text{subject to} & \quad \sum_{k=1}^{K} Q^{11}_k M_k + M_0 = -Q
\end{align*}
\]

\[
\begin{align*}
\text{minimize} & \quad -\sum_{k=1}^{K} Q^{11}_k b_k \\
\text{subject to} & \quad \sum_{k=1}^{K} Q^{11}_k M_k + M_0 \leq 0
\end{align*}
\]

This is the dual of our original problem!

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

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 $\Rightarrow$ coding (i.e. non-linear)

Demystifying the Witsenhausen Counterexample, Grover & Sahai ’10
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 Nestededness (Ho & Chu ‘72), Funnel Causality (Bahmeh & Voulgaris ‘03), Poset Causality (Shah & Parrilo ‘12)
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 Nestededness (Ho & Chu ‘72), Funnel Causality (Bahmieh & Voulgaris ’03), Poset Causality (Shah & Parrilo ‘12)
Classical Optimal Control Theory

Regulated output

Disturbance

Measured output

Control input

Closed loop map from disturbance \rightarrow \text{reg. output}

\[
\begin{align*}
\text{minimize } & \|P_{zw} + P_{zu}K(I - P_{yu}K)^{-1}P_{yw}\| \\
\text{s.t. } & K \text{ causal} \\
K(I - P_{yu}K)^{-1} & \text{ stable}
\end{align*}
\]
Classical Optimal Control Theory

---

The optimization problem is:

\[
\text{minimize}_K \| P_{zw} + P_{zu}K(I - P_{yu}K)^{-1}P_{yw}\| \\
\text{s.t. } K \text{ causal} \\
K(I - P_{yu}K)^{-1} \text{ stable}
\]

---

Feedback is non-convex
**Classical Optimal Control Theory**

Minimize $\|P_{zw} + P_{zu}QP_{yw}\|$ subject to $Q$ stable & causal

Convex in $Q$
Many decision agents leads to information asymmetry

Manifests as \textit{subspace constraints on } K \textit{ in optimal control problem.}

\[
\begin{align*}
\text{minimize}_K & \left\| P_{zw} + P_{zu} K (I - P_{yu} K)^{-1} P_{yw} \right\| \\
\text{s.t.} & \ K \text{ causal} \\
& K (I - P_{yu} K)^{-1} \text{ stable} \\
& K \in \mathcal{S}
\end{align*}
\]
Many decision agents leads to information asymmetry

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

\[
S = \frac{1}{z} \begin{bmatrix} * & 0 & 0 & 0 \\ 0 & * & 0 & 0 \\ 0 & 0 & * & 0 \\ 0 & 0 & 0 & * \end{bmatrix} \oplus \frac{1}{z^2} \begin{bmatrix} * & * & * & 0 \\ 0 & * & * & * \\ 0 & 0 & * & * \end{bmatrix} \oplus \frac{1}{z^3} \begin{bmatrix} * & * & * & * \\ 0 & * & * & * \\ 0 & 0 & * & * \end{bmatrix} \oplus \frac{1}{z^4} \mathcal{R}_p
\]
Many decision agents leads to information asymmetry

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

$$
S = \begin{bmatrix}
\ast & 0 & 0 & 0 \\
\ast & \ast & 0 & 0 \\
\ast & \ast & \ast & 0 \\
\ast & \ast & \ast & \ast
\end{bmatrix}
$$
Quadratic Invariance

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

$$KP_{yu}K \in S, \ \forall K \in S$$

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

If we have QI, model matching problem becomes

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

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)

\[ \begin{array}{c}
C_1 \quad T_C \quad C_2 \\
T_A \quad T_S \quad T_S \quad T_A \\
P_1 \quad T_P \quad P_2
\end{array} \]

\( T_C \): 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

\[\text{minimize}_{Q} \quad \left\| P_{zw} + P_{zu} Q P_{yw} \right\|\]
\[\text{s.t.} \quad Q \text{ stable & causal}\]
\[Q \in S\]

regulated output
measured output
disturbance
control input
Distributed constraint
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

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

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

\[
\begin{align*}
\text{minimize}_{Q} & \quad \| P_{zw} + P_{zu}QP_{uw} \|_{H_2}^2 \\
\text{subject to} & \quad Q \text{ stable and lower}
\end{align*}
\]

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

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?

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

Fix \(Q_{11}\) and solve

\[
\begin{align*}
\text{minimize} & \quad \| (P_{zw} + P_{zu} E_1 Q_{11} E_1^\top P_{uw}) + P_{zu} E_2 \begin{bmatrix}
Q_{12} & Q_{22}
\end{bmatrix} P_{uw} \|_{\mathcal{H}_2}^2 \\
\text{subject to} & \quad \begin{bmatrix}
Q_{12} & Q_{22}
\end{bmatrix} \text{ stable}
\end{align*}
\]

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?

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

Fix \( Q_{22} \) and solve

\[
\text{minimize} \quad \| (P_{zw} + P_{zu} E_2 Q_{22} E_2^\top P_{uw} ) + P_{zu} \begin{bmatrix}
    Q_{11} \\
    Q_{12}
\end{bmatrix} E_1^\top P_{uw} \|_{\mathcal{H}_2}^2
\]

subject to \( \begin{bmatrix}
    Q_{11}^H & Q_{12}^H
\end{bmatrix} \) stable

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

\[
S = \frac{1}{z^{t=-1}} \begin{bmatrix} * & 0 & 0 & 0 \\ 0 & * & 0 & 0 \\ 0 & 0 & * & 0 \\ 0 & 0 & 0 & * \end{bmatrix} \bigoplus \frac{1}{z^{t=-2}} \begin{bmatrix} * & * & 0 & 0 \\ * & * & 0 & 0 \\ 0 & * & * & 0 \\ 0 & 0 & * & * \end{bmatrix} \bigoplus \frac{1}{z^{t=-3}} \begin{bmatrix} * & * & * & 0 \\ * & * & * & 0 \\ * & * & * & 0 \\ * & * & * & 0 \end{bmatrix} \bigoplus \frac{1}{z^4} R_p
\]
We can play the same game: rewrite $Q$ and solve for $U$ in terms of $V$. How can we exploit strongly connected structure to reduce to centralized problems?

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

\[
Q = V \oplus U
\]
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

\[
\begin{align*}
\text{minimize} & \quad \| P_{zw} + P_{zu} V P_{uw} + P_{zu} U P_{uw} \|_2^2 \\
\text{subject to} & \quad U \in \frac{1}{z^{N+1}} \mathcal{H}_2
\end{align*}
\]

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

$$U^* = Q_N - W_L^P \sum_{z=1}^{N+1} h_2 (W_L^{-1} V W_R^{-1}) W_R$$

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

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

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

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

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.


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.
In all cases, optimal controller is as expensive to compute as centralized counterpart.

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

\[
\begin{align*}
\text{minimize} & \quad f(x[0 : k], u[0 : k]) \\
\text{subject to} & \quad x[0] = e_i \\
& \quad x[k + 1] = Ax[k] + Bu[k] \\
& \quad x[k] \in S_x \\
& \quad u[1 : k] \in S_u \\
& \quad x[T] = 0
\end{align*}
\]
Localizability

Spatio-temporal deadbeat control at each node

\[
\begin{align*}
\text{minimize} & \quad f(x[0 : k], u[0 : k]) \\
\text{subject to} & \quad x[0] = e_i \\
& \quad x[k + 1] = Ax[k] + Bu[k] \\
& \quad x[k] \in S_x \\
& \quad u[1 : k] \in S_u \\
& \quad x[T] = 0
\end{align*}
\]
Localizability

Spatio-temporal deadbeat control at each node

minimize \[ f(x[0 : k], u[0 : k]) \]
subject to \[ x[0] = e_i \]
\[ x[k + 1] = Ax[k] + Bu[k] \]
\[ x[k] \in S_x \]
\[ u[1 : k] \in S_u \]
\[ x[T] = 0 \]

Favorite convex cost
Initial disturbance
Localizability

Spatio-temporal deadbeat control at each node

\[
\begin{align*}
\text{minimize} & \quad f(x[0:k], u[0:k]) \\
\text{subject to} & \quad x[0] = e_i \\
& \quad x[k+1] = Ax[k] + Bu[k] \\
& \quad x[k] \in S_x \\
& \quad u[1:k] \in S_u \\
& \quad x[T] = 0
\end{align*}
\]

Favorite convex cost

Initial disturbance

Dynamics
**Localizability**

Spatio-temporal deadbeat control at each node

minimize \( f(x[0 : k], u[0 : k]) \)

subject to

\[
\begin{align*}
    x[0] &= e_i \\
    x[k + 1] &= Ax[k] + Bu[k] \\
    x[k] &\in S_x \\
    u[1 : k] &\in S_u \\
    x[T] &= 0
\end{align*}
\]

Favorite convex cost

Initial disturbance

Dynamics

Spatial constraints
Localizability

Spatio-temporal deadbeat control at each node

\[
\begin{align*}
\text{minimize} & \quad f(x[0 : k], u[0 : k]) \\
\text{subject to} & \quad x[0] = e_i \\
& \quad x[k + 1] = Ax[k] + Bu[k] \\
& \quad x[k] \in S_x \\
& \quad u[1 : k] \in S_u \\
& \quad x[T] = 0
\end{align*}
\]

Favorite convex cost
Initial disturbance
Dynamics
Spatial constraints
Comm constraints
Localizability

Spatio-temporal deadbeat control at each node

\[
\begin{align*}
\text{minimize} & \quad f(x[0 : k], u[0 : k]) \\
\text{subject to} & \quad x[0] = e_i \\
& \quad x[k + 1] = Ax[k] + Bu[k] \\
& \quad x[k] \in S_x \\
& \quad u[1 : k] \in S_u \\
& \quad x[T] = 0
\end{align*}
\]

Favorite convex cost

Initial disturbance
Dynamics
Spatial constraints
Comm constraints
Temporal constraints
**Localizability**

Spatio-temporal deadbeat control at each node

\[
\text{minimize} \quad f(x[0 : k], u[0 : k]) \\
\text{subject to} \quad x[0] = e_i \\
x[k + 1] = Ax[k] + Bu[k] \\
x[k] \in S_x \\
u[1 : k] \in S_u \\
x[T] = 0
\]

Favorite convex cost

Initial disturbance

Dynamics

Spatial constraints

Comm constraints

Temporal constraints

\[ S_x, S_u \quad x, u = 0 \]
**Localizability**

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

\[
\begin{align*}
\text{minimize} & \quad f(x^i[0 : k], u^i[0 : k]) \\
\text{subject to} & \quad x^i[0] = e_i \\
& \quad x^i[k + 1] = A^i x^i[k] + B^i u^i[k] \\
& \quad x^i[k] \in S^i_x \\
& \quad u^i[1 : k] \in S^i_u \\
& \quad x^i[T] = 0
\end{align*}
\]

\((A^i, B^i)\)

Favorite convex cost
Initial disturbance
Dynamics
Spatial constraints
Comm constraints
Temporal constraints

\[x, u = 0\]
**Localizability**

LQR cost splits along disturbances:

**Completely Local Globally Optimal Solution**

\[
\begin{align*}
\text{minimize} & \quad \|x^i[0 : k]\|_2^2 + \|u^i[0 : k]\|_2^2 \\
\text{subject to} & \quad x^i[0] = e_i \\
& \quad x^i[k + 1] = A^i x^i[k] + B^i u^i[k] \\
& \quad x^i[k] \in S^i_x \\
& \quad u^i[1 : k] \in S^i_u \\
& \quad x^i[T] = 0 \\
\end{align*}
\]

LQR cost

- Initial disturbance
- Dynamics
- Spatial constraints
- Comm constraints
- Temporal constraints

\[(A^i, B^i)\]
Localizability

Extensions in the works for
Output feedback

and

Non-separable cost functions

$$(A^i, B^i)$$

$S^i_x, S^i_u$

$x, u = 0$$
**Roadmap for 1\textsuperscript{st} 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 2\textsuperscript{nd} 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 2\textsuperscript{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
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

\[ u = Ky \]

Diagram:
- Plant
- Controller
- \( u \)
- \( y \)
Networked Control Systems

Classical control system

\[ u = Ky \]

\[ \infty \text{ bandwidth} \]
Networked Control Systems

Networked control system

Adding realistic channels leads to interplay between information and control theory
Networked Control Systems

Stabilization well understood

Channel Capacity $\geq$ Plant “instability”
Networked Control Systems

Stabilization well understood

Channel Capacity $\geq$ Plant “instability"

Plant ”instability”: Entropy $H = \sum |\lambda_j| \geq 1 \log_2 \lambda_j$
Networked Control Systems

Stabilization well understood

Channel Capacity ≥ Plant “instability”

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

Examples

<table>
<thead>
<tr>
<th>Channel Type</th>
<th>Condition</th>
<th>Reference</th>
</tr>
</thead>
<tbody>
<tr>
<td>Limited data rate $R$</td>
<td>$R &gt; H$</td>
<td>Nair &amp; Evans ‘04</td>
</tr>
<tr>
<td>SNR constrained AWGN</td>
<td>$\frac{C}{\log_2 e} &gt; \sum_{\lambda_i: \text{Re}\lambda_i &gt; 0} \text{Re}\lambda_i$</td>
<td>Braslavsky, Middleton &amp; Freudenberg ’07</td>
</tr>
<tr>
<td>Noisy and quantized</td>
<td>Anytime reliability &gt; $H$</td>
<td>Sahai and Mitter ‘06</td>
</tr>
</tbody>
</table>
Networked Control Systems

Stabilization well understood

Channel Capacity ≥ Plant “instability”

Plant ”instability”: Entropy \( H = \sum |\lambda_j| \geq 1 \log_2 \lambda_j \)

Examples

<table>
<thead>
<tr>
<th>Channel Type</th>
<th>Condition</th>
<th>Reference</th>
</tr>
</thead>
<tbody>
<tr>
<td>Limited data rate ( R )</td>
<td>( R &gt; H )</td>
<td>Nair &amp; Evans ‘04</td>
</tr>
<tr>
<td>SNR constrained AWGN</td>
<td>( \frac{C}{\log_2 e} &gt; \sum_{\lambda_i: \text{Re} \lambda_i &gt; 0} \text{Re} \lambda_i )</td>
<td>Braslavsky, Middleton &amp; Freudenberg ‘07</td>
</tr>
<tr>
<td>Noisy and quantized</td>
<td>Anytime reliability &gt; ( H )</td>
<td>Sahai and Mitter ‘06</td>
</tr>
</tbody>
</table>

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

Tree codes for achieving anytime reliability (Sukhavasi & Hassibi ‘13)
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:
\[
S2 \leq C_f - \sum_{i=1}^{n} \max\{0, \log |\lambda_i(A)|\}
\]

*Note: Standard Bode integral bound is the common bound used in control systems.
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
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
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**

\[ x_{t+1} = Ax_t + Bu_t \]
\[ y_t = Cx_t + Du_t \]

**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

\[
\begin{align*}
x_{t+1} &= Ax_t + Bu_t \\
y_t &= Cx_t + Du_t
\end{align*}
\]

Input/output

\[
\begin{align*}
y_t &= \sum_{\tau=0}^{t} G_{\tau} u_{t-\tau} \\
G_0 &= D, \quad G_{\tau} = CA^{\tau-1}B
\end{align*}
\]

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

\[
U_{N,M,r} = \begin{bmatrix} u_{N-M} & u_{N-(M-1)} & \cdots & u_N \\
 u_{N-(M+1)} & u_{N-M} & \cdots & u_{N-1} \\
 \vdots & \vdots & \ddots & \vdots \\
 u_{N-(M+r)} & u_{N-(M+r-1)} & \cdots & u_{N-r} \end{bmatrix}
\]
SysID with Known Structure

Quick Review of Basic SysID

Dynamics
\[
x_{t+1} = Ax_t + Bu_t \\
y_t = Cx_t + Du_t
\]

Input/output
\[
y_t = \sum_{\tau=0}^{t} G_{\tau} u_{t-\tau} \\
G_0 = D, \quad G_\tau = CA^{\tau-1}B
\]

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

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

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:

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

If system order $n$ is less than $r$ then $\text{rank}(\mathcal{H}(G))=n$, 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

\[
\begin{align*}
\text{minimize} & \quad \text{rank}(\mathcal{H}(G)) \\
\text{subject to} & \quad \|Y_N - GU_{N,M,r}\|_F^2 \leq \delta^2
\end{align*}
\]
SysID with Known Structure

Combine to deal with process and observation noise

\[
\begin{align*}
\text{minimize} & \quad \text{rank}(\mathcal{H}(G)) \\
\text{subject to} & \quad \|Y_N - GU_{N,M,r}\|_F^2 \leq \delta^2
\end{align*}
\]

Non-convex!
Relax to

\[
\begin{align*}
\text{minimize} & \quad \|\mathcal{H}(G)\|_* \\
\text{subject to} & \quad \|Y_N - GU_{N,M,r}\|_F^2 \leq \delta^2
\end{align*}
\]

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

\[
\begin{align*}
\text{minimize} & \quad \| \mathcal{H}(G) \|_* \\
\text{subject to} & \quad \| Y_N - GU_{N,M,r} \|_F^2 \leq \delta^2
\end{align*}
\]

Where now \( U \) consists of local inputs and measured interconnecting signals.
SysID with Known Structure

Easy case: we can measure all interconnecting signals

\[
\begin{align*}
\text{minimize} & \quad \|\mathcal{H}(G)\|_* \\
\text{subject to} & \quad \|Y_N - GU_{N,M,r}\|_F^2 \leq \delta^2
\end{align*}
\]

Where now \( U \) consists of local inputs and measured interconnecting signals.

Need to get neighbors to inject excitation as well.
Tricky case: we miss some interconnecting signals

High Order Large Scale System

Local system

Local inputs

Measured interconnection signals

Hidden interconnection signals

Low+high order local measurements

Low-Rank and Low-Order Decompositions for Local System Identification, M. & Rantzer ‘14
SysID with Known Structure

Tricky case: we miss some interconnecting signals

\[ y_t = \sum_{\tau=0}^{t} G_{\tau} u_{t-\tau} + H_{\tau} u_{t-\tau} \]
Tricky case: we miss some interconnecting signals

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

- **Low-order** but full rank
- **High-order** but low rank

*SysID with Known Structure*

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

Tricky case: we miss some interconnecting signals

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

Low-order but full rank
High-order but low rank

Can we separate out the two components?
**SysID with Known Structure**

Tricky case: we miss some interconnecting signals

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

Low-order but full rank \hspace{1cm} High-order but low rank

Can we separate out the two components?

\[
\begin{align*}
\text{minimize} & \quad \text{rank}(\mathcal{H}(G)) \\
\text{subject to} & \quad \| Y_N - (G + H) U_{N,M,r} \|_F^2 \leq \delta^2 \\
& \quad \text{rank}(H(e^{i\omega})) \leq k
\end{align*}
\]
SysID with Known Structure

Tricky case: we miss some interconnecting signals

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

Can we separate out the two components?

\[
\begin{align*}
\text{minimize} & \quad \| \mathcal{H}(G) \|_* \\
\text{subject to} & \quad \| Y_N - (G + H) U_{N,M,r} \|_F^2 \leq \delta^2 \\
& \quad \| H(e^{j\omega}) \|_* \leq \kappa
\end{align*}
\]
Tricky case: we miss some interconnecting signals

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

Low-order but full rank

High-order but low rank

Key feature:
exploiting structure to de-convolve response

\[
\begin{align*}
\text{minimize} & \quad \| \mathcal{H}(G) \|_* \\
\text{subject to} & \quad \| Y_N - (G + H) U_{N,M,r} \|_F^2 \leq \delta^2 \\
& \quad \| H(e^{j\omega}) \|_* \leq k
\end{align*}
\]
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

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

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

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

\[ \Sigma^{-1} = \begin{bmatrix} * & 0 & 0 & 0 & 0 & * \\ 0 & * & 0 & 0 & 0 & * \\ 0 & 0 & * & 0 & 0 & * \\ 0 & 0 & 0 & * & 0 & * \\ * & * & * & * & * & * \end{bmatrix} \]
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
If we know model is sparse \textit{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

$$\minimize_{K} \quad \text{Tr} \hat{\Sigma} K - \log \det K + \lambda \|K\|_0$$

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

$$\minimize_{K} \quad \text{Tr} \hat{\Sigma} K - \log \det K + \lambda \|K\|_1$$

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

$$\text{minimize}_{K} \quad \text{Tr} \hat{\Sigma} K - \log \det K + \lambda \|K\|_1$$

convex

This works! Banerjee et al. ‘06, Ravikumar et al. ’08, …
Latent Variables in Graphical Models

But what if we miss a variable?

\[
\Sigma^{-1} = \begin{bmatrix}
* & 0 & 0 & 0 & 0 & * \\
0 & * & 0 & 0 & 0 & * \\
0 & 0 & * & 0 & 0 & * \\
0 & 0 & 0 & * & 0 & * \\
* & * & * & * & * & * \\
* & * & * & * & * & * \\
\end{bmatrix}
\]
Latent Variables in Graphical Models

But what if we miss a variable?
Latent Variables in Graphical Models

But what if we miss a variable?

\[
(\Sigma_O)^{-1} = \begin{bmatrix}
\ast & \ast & \ast & \ast & \ast \\
\ast & \ast & \ast & \ast & \ast \\
\ast & \ast & \ast & \ast & \ast \\
\ast & \ast & \ast & \ast & \ast \\
\ast & \ast & \ast & \ast & \ast 
\end{bmatrix}
\]
Latent Variables in Graphical Models

But what if we miss a variable?

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

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

\[
(\Sigma_O)^{-1} = \begin{bmatrix}
* & * & * \\
* & * & * \\
* & * & * \\
* & * & *
\end{bmatrix}
\]
But what if we miss a variable?

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

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

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

Sparse

Low-rank
Latent Variables in Graphical Models

But what if we miss a variable?

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

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

\[
\text{minimize } \text{Tr} \hat{\Sigma}_O (S - L) - \log \det (S - L) + \lambda \| S \|_1 + \gamma \| L \|_*
\]

subject to \( S - L \succ 0, L \succeq 0 \)

This works! Chandrasekaran, Parrilo & Willsky ’12
Latent Variables in Graphical Models

But what if we miss a variable?

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

Key feature: exploiting structure to de-convolve response

$$
\text{minimize } \operatorname{Tr} \hat{\Sigma}_O (S - L) - \log \det (S - L) + \lambda \| S \|_1 + \gamma \| L \|_* \\
\text{subject to } S - L \succ 0, L \succeq 0
$$

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

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

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

\[
U^* = Q_N - W_L P \frac{1}{z_{N+1}} H_2 (W_L^{-1} V W_R^{-1}) W_R
\]

Current time \(-(N+1)\) Time in the past
Comm Delay Co-Design

\[
\text{minimize}_V \sum_{i=1}^{N} \left( \text{Tr} G_i(V) (G_i(V))^\top + 2\text{Tr} G_i(V)T_i^\top \right) \\
\text{s.t. } V_i \in \mathcal{Y}_i
\]

- **Entire** decentralized nature captured in \( V \)
Comm Delay Co-Design

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

s.t. \( V \leq Y_i \)

- Entire decentralized nature captured in \( V \)
- Remove constraints
Comm Delay Co-Design

\[
\text{minimize}_V \sum_{i=1}^{N} \left( \text{Tr} G_i(V) (G_i(V))^\top + 2 \text{Tr} G_i(V) T_i^\top \right) + \lambda \| V \|_A
\]

s.t. \( V \geq Y_i \)

- **Entire** decentralized nature captured in \( V \)
- Remove constraints
- Add penalty to *induce* simple structure
Comm Delay Co-Design

\[
\text{minimize}_V \sum_{i=1}^{N} \left( \text{Tr} G_i(V) (G_i(V))^\top + 2\text{Tr} G_i(V) T_i^\top \right) + \lambda \| V \|_A \\
\text{s.t. } V \preceq \mathcal{Y}_i
\]

- **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

\[ \|X\|_A := \inf\{ t > 0 \mid X \in t\text{conv}(A) \} \]

- Sparse vectors
- Low-rank matrices
- "Good" graphs

[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
Start with base that is QI and SC

\[ S = \frac{1}{z} \begin{bmatrix} * & 0 & 0 & 0 \\ 0 & * & 0 & 0 \\ 0 & 0 & * & 0 \\ 0 & 0 & 0 & * \end{bmatrix} \oplus \frac{1}{z^2} \begin{bmatrix} * & * & 0 & 0 \\ * & * & * & 0 \\ 0 & 0 & * & * \\ 0 & 0 & * & * \end{bmatrix} \oplus \frac{1}{z^3} \begin{bmatrix} * & * & * & 0 \\ * & * & * & * \\ 0 & * & * & * \\ 0 & * & * & * \end{bmatrix} \oplus \frac{1}{z^4} \mathcal{H}_2 \]

Add shortcuts

\[ S = \frac{1}{z} \begin{bmatrix} * & 0 & 0 & 0 \\ 0 & * & 0 & 0 \\ 0 & 0 & * & 0 \\ 0 & 0 & 0 & * \end{bmatrix} \oplus \frac{1}{z^2} \begin{bmatrix} * & * & * & 0 \\ * & * & * & 0 \\ 0 & 0 & * & * \\ 0 & 0 & * & * \end{bmatrix} \oplus \frac{1}{z^3} \begin{bmatrix} * & * & * & * \\ * & * & * & * \\ 0 & * & * & * \\ 0 & * & * & * \end{bmatrix} \oplus \frac{1}{z^4} \mathcal{H}_2 \]

Project out base

\[ a_{13} = \frac{1}{z} \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} \oplus \frac{1}{z^2} \begin{bmatrix} 0 & 0 & * & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & * \end{bmatrix} \oplus \frac{1}{z^3} \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} \oplus \frac{1}{z^4} \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} \]
The Graph Enhancement “Norm”

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

\[ \|x\|_{\mathcal{A}} = \min_{x_1, x_2} \|x_1\|_2 + \|x_2\|_2 \]
subject to
\[ \sum x_i = x \]
\[ \text{supp}(x_i) \subset \text{supp}(a_i) \]
\[ \mathcal{A} = \{[*,*,*], [0,*,*]\} \]

Convex hull of low dimensional unit disks
**Theorem** [N.M. CDC ‘13, TCNS ’14]

Solving

\[
\begin{align*}
\text{minimize}_Q & \quad \|Q\|_A \\
\text{s.t.} & \quad N(Q)^2 \\
& \quad N_c^2 \leq \delta^2
\end{align*}
\]

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

Closed Loop Norm vs. # Links

Base

Augmented

Centralized (infeasible)
**Actuator Regularization**

**Goal**  
Choose which actuators we need

**Approach**  
Assume $B$ is block-diagonal.

minimize \( Q \| P_{zw} + P_{zu} Q P_{yw} \| \)  
s.t. \( Q \) stable & causal
**Actuator Regularization**

**Goal**

Choose which actuators we need

**Approach**

Assume $B$ is block-diagonal.

\[
\begin{align*}
\text{minimize}_{Q} & \| P_{zw} + P_{zu} Q P_{yw} \| \\
\text{s.t.} & \quad Q \text{ stable & causal}
\end{align*}
\]

Then each block-row of $Q$ corresponds to an actuator.
Actuator Regularization

Goal
Choose which actuators we need

Approach
Assume $B$ is block-diagonal.

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”

\[
\text{minimize}_Q \| P_{zw} + P_{zu}QP_{yw} \| \quad \text{s.t. } Q \text{ stable & causal}
\]
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)
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 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.

\[ \text{ill-posed on full support} = \text{well-posed on restricted support} \]
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

\[
\text{minimize}_x \quad \|C(x, y)\| + \lambda \|x\|_A
\]

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

$$\min_v \frac{1}{2} \| y - \mathcal{L}E_G(v) \|^2_F + \lambda \| v \|_G$$

Performance

Open loop system

Simplicity
The Group Norm

\[ g = \| v_1 \| + \| v_2 \| + \| v_3 \| + \| v_4 \| \]

With dual norm

\[ g, \infty = \max \left\{ \| v_1 \|, \| v_2 \|, \| v_3 \|, \| v_4 \| \right\} \]
Focus on Structure

$\mathcal{E}_g$-support accurate

$\text{supp} \subseteq \text{supp}$

Recover a subset of the structure

$G$-support accurate

$\text{gsupp} = \text{gsupp}$

Recover full structure
Accurate Approximations

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

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

Sparse nominal controller \quad Nominal closed loop

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

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

\[
\frac{\gamma}{\alpha} \leq \nu
\]

Total Incoherence
Support Accurate Approximations

**Theorem** [N.M. and V. Chandrasekaran, CDC ‘14]
Suppose previous assumptions hold, and \( \| \mathcal{E}^+ G^+ \epsilon \|_{\mathcal{G}, \infty} \leq (\kappa - 1) \lambda \) for some \( 1 \leq \kappa < \frac{2}{(\nu+1)} \).
Then

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

**Corollary**
If \( \| v_g^* \| > \lambda \left( \frac{\kappa}{\alpha} \right) \) for all \( g \in \mathcal{G}^* \). Then \( \hat{v} \) is \( \mathcal{G} \)-support accurate.
Suppose previous assumptions hold, and \( \|\mathcal{E}_G^+ \mathcal{L}^+ \epsilon\|_{\mathcal{G},\infty} \leq (\kappa - 1)\lambda \) for some \( 1 \leq \kappa < \frac{2}{(\nu+1)} \).

Then

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

**Corollary**

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

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

$$\min_v \frac{1}{2} \| y - \mathcal{L} \mathcal{E}_G(v) \|_F^2 + \lambda \| v \|_g$$

**Performance**

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

Sparse nominal controller Nominal closed loop

**Simplicity**

$\min_Q \| P_{zw} + P_{zu} Q P_{yw} \|$ s.t. $Q$ stable & causal
**Actuator Regularization: Sample Path**

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

$\mathcal{G}$-support accurate
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 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
Recap of 2\textsuperscript{nd} 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
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?