Towards a Theory of Scalable Control

Anders Rantzer

Automatic Control LTH
Lund University
Sweden

New Generation of Heating Networks

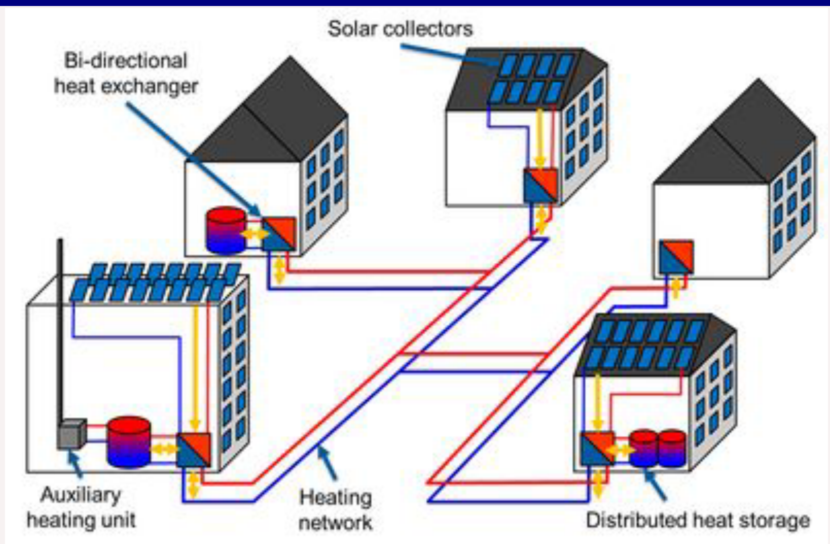

LUNDS NYA STADSDEL BRUNNSHÖG 23 september 2019 12:12

Lund först i världen med ljummen fjärrvärme

Kranen öppnas på tisdag. Brunnshög blir den första stadsdelen i världen som får ett lågvärmenät. Vattnet är tjugo grader kallare än i det vanliga fjärrvärmenätet. Men vad ska det vara bra för?

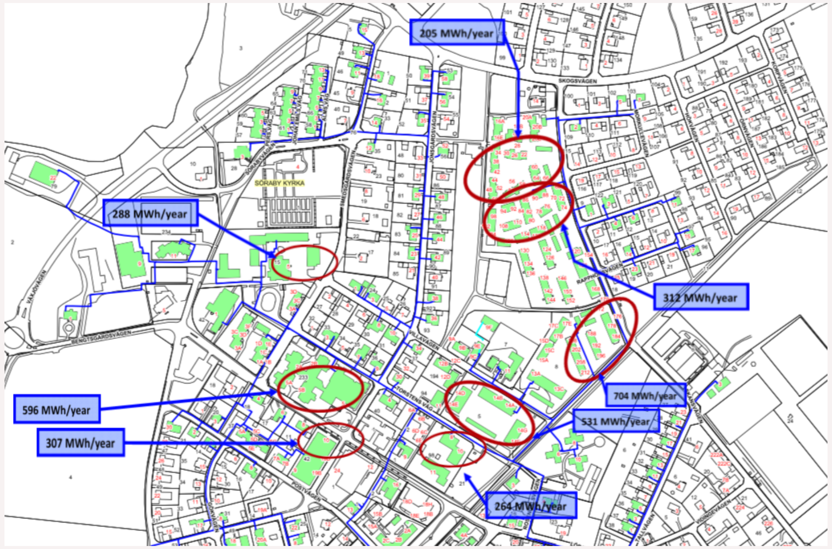
Kalle Kniivilä + Följ

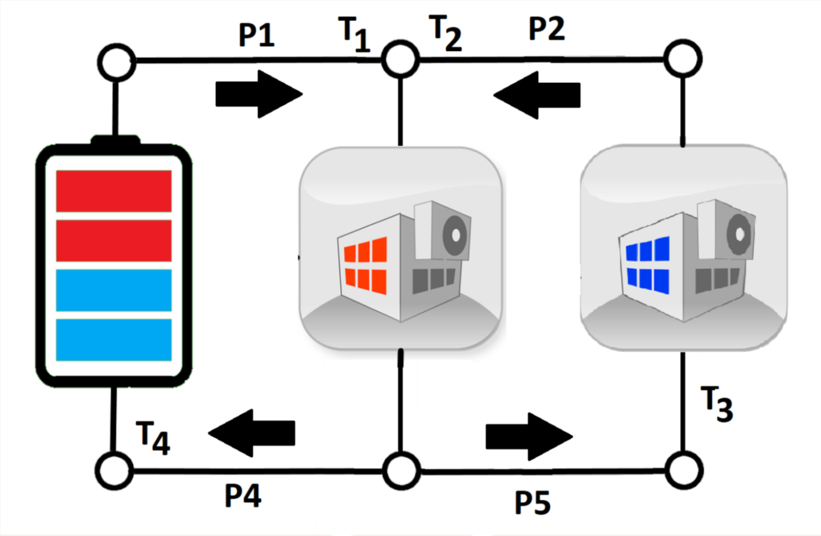
New Generation of Heating Networks

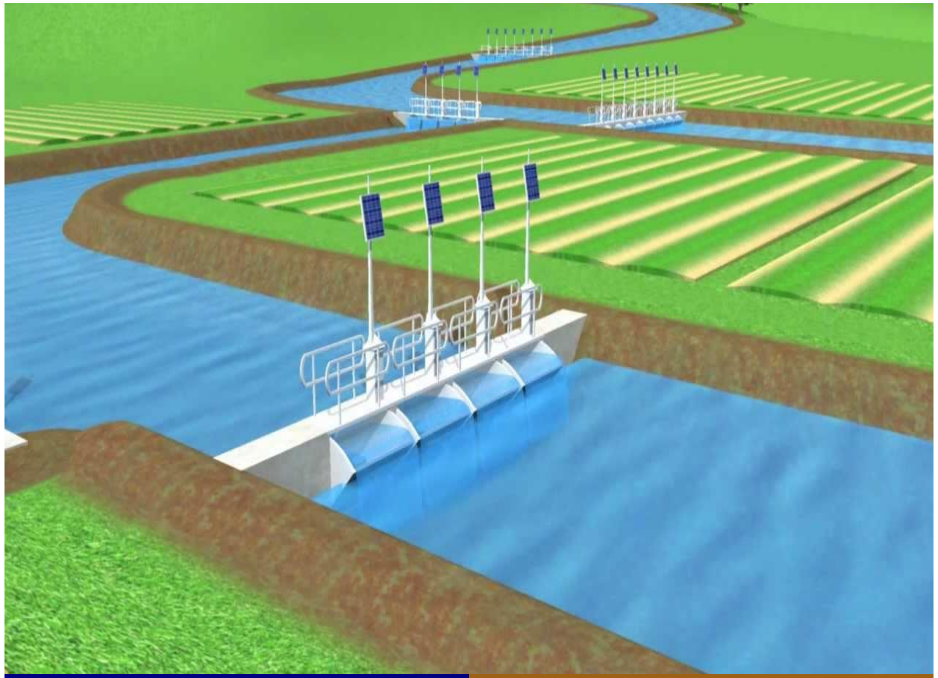


Fossil Free Sweden 2045 !

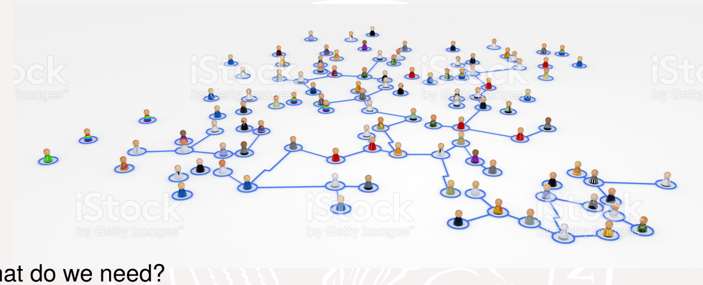
Heating Networks







Towards a Scalable Theory of Control



What do we need?

- Scalable Synthesis
- Scalable Verification
- Scalable Modeling
- Scalable Objectives

Background

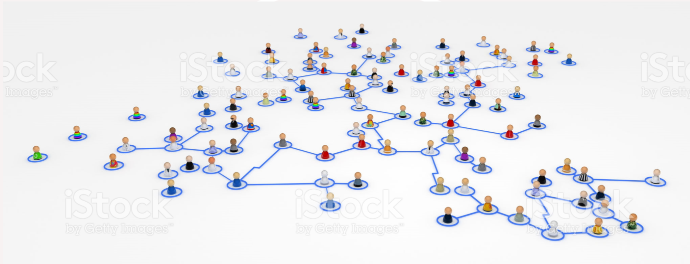
Books on large-scale control and coordination:

Mesarovic, Macko, Takahara (1970)
Singh, Titli (1978)
Findeisen (1980)

Optimization of structured controllers:

Spatially invariant systems: Bamieh, Paganini, Dahleh (2002)
Distributed controllers: D'Andrea and Dullerud (2003)
Quadratic invariance: Rotkowitz and Lall (2002)
Low rank coordination: Madjidian and Mirkin (2014)
Scalability using positivity: Rantzer (2015)
Systems Level Synthesis: Wang, Matni, Doyle (2018)

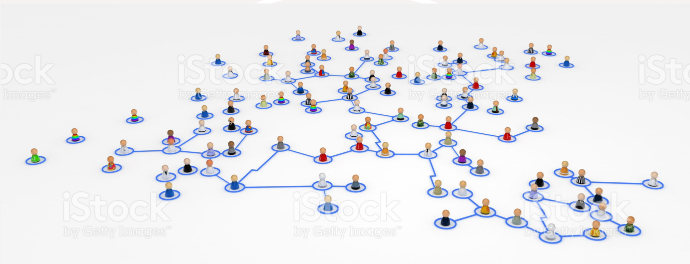
Three Widespread Myths



Three widespread myths

- Scalable controllers are hard to optimize
- H_∞ optimal controllers are not scalable
- H_2 optimal controllers are not scalable

Three Widespread Myths



Three widespread myths

- Scalable controllers are hard to optimize
- H_∞ optimal controllers are not scalable
- H_2 optimal controllers are not scalable

They are all wrong!

Outline

- **Network realizable control** (Systems Level Synthesis)
  - Don't look for sparse transfer functions
  - Connection to Internal Model Control
- Scalable H_∞ optimal synthesis
  - H_∞ optimal static controllers
  - H_∞ optimal dynamic controllers
- Scalable H_2 optimal synthesis
  - a transportation example
- Concluding remarks

Example: River Dams

Consider a model of three water dams along a river:

$$\begin{aligned} x_1(t+1) &= 0.9x_1(t) - u_1(t) \\ x_2(t+1) &= 0.1x_1(t) + 0.8x_2(t) + u_1(t) - u_2(t) \\ x_3(t+1) &= 0.2x_2(t) + 0.7x_3(t) + u_2(t) - u_3(t) \end{aligned}$$

Information propagates downstream.

The transfer function from (u_1, u_2, u_3) to (x_1, x_2, x_3) is triangular:

$$\mathbf{P}(z) = \begin{bmatrix} * & 0 & 0 \\ * & * & 0 \\ * & * & * \end{bmatrix}$$

The localized structure of the state realization is lost in the transfer matrix.

Example: River Dams

Consider a model of three water dams along a river:

$$\begin{aligned} x_1(t+1) &= 0.9x_1(t) - u_1(t) \\ x_2(t+1) &= 0.1x_1(t) + 0.8x_2(t) + u_1(t) - u_2(t) \\ x_3(t+1) &= 0.2x_2(t) + 0.7x_3(t) + u_2(t) - u_3(t) \end{aligned}$$

Information propagates downstream.

The transfer function from (u_1, u_2, u_3) to (x_1, x_2, x_3) is triangular:

$$\mathbf{P}(z) = \begin{bmatrix} * & 0 & 0 \\ * & * & 0 \\ * & * & * \end{bmatrix}$$

The localized structure of the state realization is lost in the transfer matrix.

Closed Loop Convexity

In general, specifications are computationally tractable only if they are convex constraints on the closed loop map

$$\mathbf{C}(I + \mathbf{PC})^{-1}.$$

Sparsity constraints on the matrix $\mathbf{C}$ are not closed loop convex and very difficult to enforce.

However, there is a better choice...

Closed Loop Convexity

In general, specifications are computationally tractable only if they are convex constraints on the closed loop map

$$\mathbf{C}(I + \mathbf{PC})^{-1}.$$

Sparsity constraints on the matrix $\mathbf{C}$ are not closed loop convex and very difficult to enforce.

However, there is a better choice...

Closed Loop Convexity

In general, specifications are computationally tractable only if they are convex constraints on the closed loop map

$$\mathbf{C}(I + \mathbf{PC})^{-1}.$$

Sparsity constraints on the matrix $\mathbf{C}$ are not closed loop convex and very difficult to enforce.

However, there is a better choice...

Network Realizability

Given a graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$, a transfer matrix $\mathbf{G}$ is said to be *network realizable on* $\mathcal{G}$ if it has a stabilizable and detectable realization $\mathbf{G}(z) = C(zI - A)^{-1}B + D$ with

$$\left[\begin{array}{c|c} A & B \\ \hline C & D \end{array}\right] = \left[\begin{array}{ccc|ccc} A_{11} & \dots & A_{1N} & B_1 & & 0 \\ \vdots & & \vdots & & \ddots & \\ A_{N1} & \dots & A_{NN} & 0 & & B_N \\ \hline C_{11} & \dots & C_{1n} & D_1 & & 0 \\ \vdots & & \vdots & & \ddots & \\ C_{N1} & \dots & C_{NN} & 0 & & D_N \end{array}\right]$$

where $A_{ij} = 0$ and $C_{ij} = 0$ for $(i, j) \notin \mathcal{E}$.

Network Realizability is a Closed Loop Convex Property!

Theorem 1
Network Realizability is preserved by addition and proper inversion!
If $\mathbf{G}_1$ and $\mathbf{G}_2$ are stable and network realizable, then so is $\mathbf{G}_1\mathbf{G}_2$.

The proof is straightforward
For example, if $\mathbf{G}(z) = C(zI - A)^{-1}B + D$ and D is invertible, then $\mathbf{G}^{-1}$ has the realization

$$\left[\begin{array}{c|c} A - BD^{-1}C & BD^{-1} \\ \hline -D^{-1}C & D^{-1} \end{array}\right].$$

Network Realizability is a Closed Loop Convex Property!

Theorem 1
Network Realizability is preserved by addition and proper inversion!
If $\mathbf{G}_1$ and $\mathbf{G}_2$ are stable and network realizable, then so is $\mathbf{G}_1\mathbf{G}_2$.

The proof is straightforward
For example, if $\mathbf{G}(z) = C(zI - A)^{-1}B + D$ and D is invertible, then $\mathbf{G}^{-1}$ has the realization

$$\left[\begin{array}{c|c} A - BD^{-1}C & BD^{-1} \\ \hline -D^{-1}C & D^{-1} \end{array}\right].$$

Network Realizable Control of Stable Plants

Theorem 2
Suppose that the transfer matrix $\mathbf{P}$ is strictly proper, stable and network realizable on $\mathcal{G}$. Then, the controller $\mathbf{C}$ is stabilizing and network realizable if and only if $\mathbf{Q} = \mathbf{C}(I + \mathbf{PC})^{-1}$ is stable and network realizable.

Enforcing network realizability of $\mathbf{Q}$ can be done by convex optimization.

But how do we construct a realization of $\mathbf{C}$ after optimizing $\mathbf{Q}$?

Network Realizable Control of Stable Plants

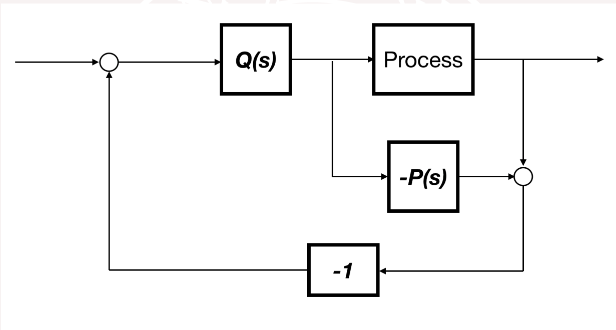
Theorem 2
Suppose that the transfer matrix $\mathbf{P}$ is strictly proper, stable and network realizable on $\mathcal{G}$. Then, the controller $\mathbf{C}$ is stabilizing and network realizable if and only if $\mathbf{Q} = \mathbf{C}(I + \mathbf{PC})^{-1}$ is stable and network realizable.

Enforcing network realizability of $\mathbf{Q}$ can be done by convex optimization.

But how do we construct a realization of $\mathbf{C}$ after optimizing $\mathbf{Q}$?

Internal Model Control (IMC)

[Garcia/Morari, 1982]



If both $\mathbf{P}$ and $\mathbf{Q}$ have network realizations on a given graph, then after proper ordering of the states and block partitioning of the matrices also the IMC controller will be a network realization on that graph.

Example: River Dams

$$\begin{bmatrix} x_1^+ \\ x_2^+ \\ x_3^+ \end{bmatrix} = \begin{bmatrix} 0.9 & 0 & 0 \\ 0.1 & 0.8 & 0 \\ 0 & 0.2 & 0.7 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} + \begin{bmatrix} -u_1 \\ u_1 - u_2 \\ u_2 - u_3 \end{bmatrix}$$

Let $\mathbf{Q}(z) = E(zI - A)^{-1}F$ be the desired map from reference to input:

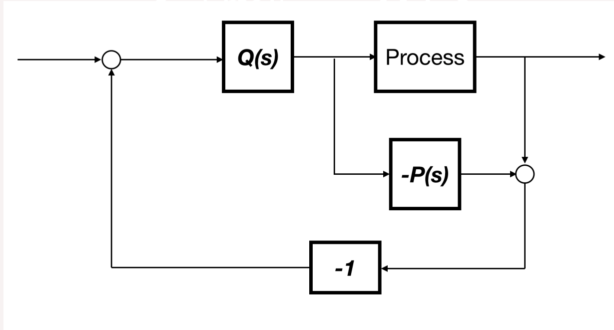
$$\left[\begin{array}{c|c} E & F \\ \hline G & H \end{array}\right] = \left[\begin{array}{ccc|ccc} E_{11} & 0 & 0 & F_1 & 0 & 0 \\ E_{21} & E_{22} & 0 & 0 & F_2 & 0 \\ 0 & E_{32} & E_{33} & 0 & 0 & F_3 \\ \hline G_1 & 0 & 0 & 0 & 0 & 0 \\ 0 & G_2 & 0 & 0 & 0 & 0 \\ 0 & 0 & G_3 & 0 & 0 & 0 \end{array}\right]$$

Example: River Dams

Then the controller $\mathbf{C} = \mathbf{Q}(I - \mathbf{P}\mathbf{Q})^{-1}$ has the realization

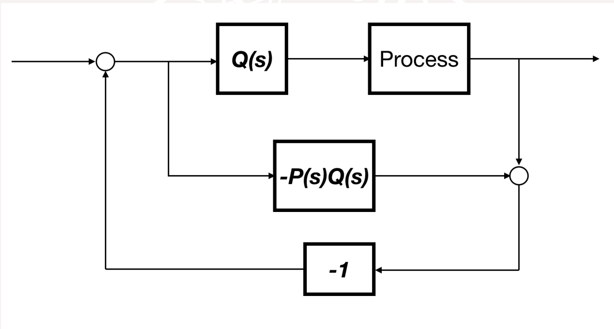
$$\begin{bmatrix} \hat{x}_1^+ \\ \xi_1^+ \\ \hat{x}_2^+ \\ \xi_2^+ \\ \hat{x}_3^+ \\ \xi_3^+ \end{bmatrix} = \left[\begin{array}{cc|cc|cc} 0.9 & -G_1 & 0 & 0 & 0 & 0 \\ F_1 & E_{11} & 0 & 0 & 0 & 0 \\ \hline 0.1 & G_1 & 0.8 & -G_2 & 0 & 0 \\ 0 & E_{21} & F_2 & E_{22} & 0 & 0 \\ \hline 0 & 0 & 0.2 & G_2 & 0.7 & -G_3 \\ 0 & 0 & 0 & E_{32} & F_3 & E_{33} \end{array}\right] \begin{bmatrix} \hat{x}_1 \\ \xi_1 \\ \hat{x}_2 \\ \xi_2 \\ \hat{x}_3 \\ \xi_3 \end{bmatrix} - \begin{bmatrix} 0 \\ e_1 \\ 0 \\ e_2 \\ 0 \\ e_3 \end{bmatrix}.$$

Internal Model Control (IMC)



What if $P(s)$ is unstable?

Internal Model Control (IMC)



Design stable $Q(s)$ to make also $Q(s)P(s)$ stable!

Network Realizability is a Closed Loop Convex Property!

Theorem 3
Consider $\mathbf{P}$ and $\mathbf{C}$ such that $\mathbf{P}$ is strictly proper and define the closed loop

$$\mathbf{H} = \begin{bmatrix} I & -\mathbf{P} \\ \mathbf{C} & I \end{bmatrix}^{-1} = \begin{bmatrix} (I+\mathbf{PC})^{-1} & \mathbf{P}(I+\mathbf{CP})^{-1} \\ -\mathbf{C}(I+\mathbf{PC})^{-1} & (I+\mathbf{CP})^{-1} \end{bmatrix}.$$

Then the following two statements are equivalent:

(i) Both $\mathbf{P}$ and $\mathbf{C}$ are network realizable on $\mathcal{G}$.
(ii) $\mathbf{H}$ is network realizable on $\mathcal{G}$.

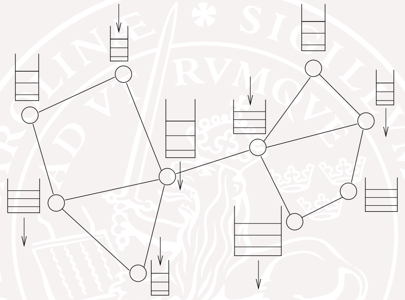
The following two statements are also equivalent:

(iii) Both $\mathbf{PC}$ and $\mathbf{C}$ are network realizable on $\mathcal{G}$.
(iv) Both $(I+\mathbf{PC})^{-1}$ and $\mathbf{C}(I+\mathbf{PC})^{-1}$ are network realizable on $\mathcal{G}$.

Outline

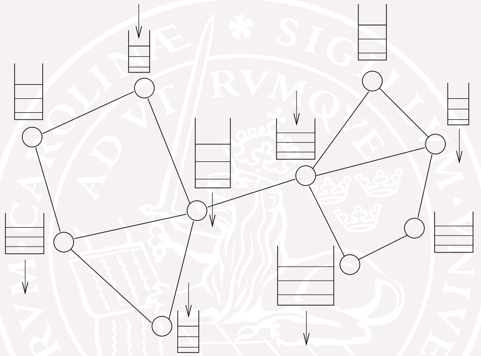
- Network realizable control
  - Don't look for sparse transfer functions
  - Connection to Internal Model Control
- **Scalable H_∞ optimal synthesis**
  - H_∞ optimal static controllers
  - H_∞ optimal dynamic controllers
- Scalable H_2 optimal synthesis
  - a transportation example
- Concluding remarks

Dynamic Buffer Networks



- Producers, consumers and storages
- Examples: water, power, traffic, data
- Discrete/continuous, stochastic/deterministic
- Multiple commodities, human interaction

Scalable Synthesis for Dynamic Buffer Networks



A common approach is to just focus on stationary optimality or equilibrium conditions and ignore dynamics. ("No arbitrage")

When is this justified for a control system?

H_∞ Optimal Static Control on Networks

Problem:
Given a graph $(\mathcal{V}, \mathcal{E})$ and

$$\dot{x}_i = a_i x_i + \sum_{(i,j)\in\mathcal{E}} (u_{ij} - u_{ji}) + w_i \qquad i \in \mathcal{V}$$

find control law $u = Kx$ that minimizes the H_∞ norm of the map from w to (x, u).

Solution:
An optimal control law when $a_i < 0$ is given by

$$u_{ij} = x_i/a_i - x_j/a_j \qquad (i,j) \in \mathcal{E}.$$

[Lidström/Rantzer, ACC2016]

H_∞ Optimal Static Control on Networks

Problem:

Given a graph $(\mathcal{V}, \mathcal{E})$ *and*

$$\dot{x}_i = a_i x_i + \sum_{(i,j)\in\mathcal{E}} (u_{ij} - u_{ji}) + w_i \qquad i \in \mathcal{V}$$

find control law $u = Kx$ *that minimizes the* H_∞ *norm of the map from* w *to* (x, u).

Solution:

An optimal control law when $a_i < 0$ is given by

$$u_{ij} = x_i/a_i - x_j/a_j \qquad (i, j) \in \mathcal{E}.$$

[Lidström/Rantzer, ACC2016]

Example

$$\dot{x} = \underbrace{-\operatorname{diag}(1,3,2)}_{A}\, x + \underbrace{\begin{bmatrix} -1 & 0 & 0 \\ 1 & 1 & -1 \\ 0 & 0 & 1 \end{bmatrix}}_{B} u + w$$

Find controller $u = Kx$ that minimizes the H_∞ gain from w to (x, u). The Riccati solution gives

$$K_1 = \begin{bmatrix} 0.93 & -0.11 & 0.00 \\ -0.05 & -0.17 & -0.01 \\ 0.04 & 0.16 & -0.26 \end{bmatrix}$$

Our theorem gives another optimal solution:

$$K_2 = \begin{bmatrix} 1 & -\frac{1}{3} & 0 \\ 0 & -\frac{1}{3} & 0 \\ 0 & \frac{1}{3} & -\frac{1}{2} \end{bmatrix}$$

Anders Rantzer | Towards a Theory of Scalable Control

Structure Preserving Static Feedback

Problem

Consider the system $\dot{x} = Ax + Bu + w$ with A symmetric and Hurwitz. Find a state feedback controller $u = Kx$ that minimizes the H_∞ norm of the map from w to (x, u) in the closed loop system $\dot{x} = (A + BK)x + w$.

Theorem

A solution is given by $u = K_* x$ where $K_* = B^T A^{-1}$. The minimal value of the norm is $\sqrt{\|(A^2 + BB^T)^{-1}\|}$.

Proof idea

$K_* = B^T A^{-1}$ minimizes the static gain. Other frequencies are better off.

Anders Rantzer | Towards a Theory of Scalable Control

Structure Preserving Static Feedback

Problem

Consider the system $\dot{x} = Ax + Bu + w$ with A symmetric and Hurwitz. Find a state feedback controller $u = Kx$ that minimizes the H_∞ norm of the map from w to (x, u) in the closed loop system $\dot{x} = (A + BK)x + w$.

Theorem

A solution is given by $u = K_* x$ where $K_* = B^T A^{-1}$. The minimal value of the norm is $\sqrt{\|(A^2 + BB^T)^{-1}\|}$.

Proof idea

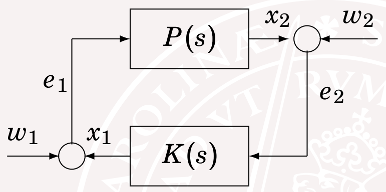
$K_* = B^T A^{-1}$ minimizes the static gain. Other frequencies are better off.

Anders Rantzer | Towards a Theory of Scalable Control

Outline

- Network realizable control
  - Don't look for sparse transfer functions
  - Connection to Internal Model Control
- Scalable H_∞ optimal synthesis
  - H_∞ optimal static controllers
  - **H_∞ optimal dynamic controllers**
- Scalable H_2 optimal synthesis
  - a transportation example
- Concluding remarks

Frequency Weighted Specifications



$$e_1 = Ke_2 + w_1$$
$$e_2 = Pe_1 + w_2$$

Disturbance rejection:
The transfer functions $(I+PK)^{-1}$ and $(I+PK)^{-1}P$ should be small for low frequencies. ("Integral action")

Measurement errors:
The transfer functions $K(I+PK)^{-1}$ and $PK(I+PK)^{-1}$ should be small for high frequencies.

Structure Preserving H_∞ Control

Theorem
Let $P(s) = (sI - A)^{-1}B$ with A symmetric negative definite. The problem

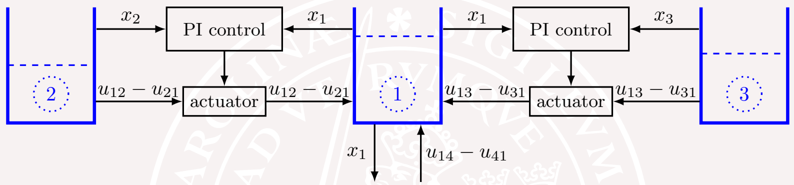
$$\begin{aligned} &\text{Minimize} && \|(I+KP)^{-1}K\|_\infty \\ &\text{subject to} && \|\tfrac{1}{s}P(I+KP)^{-1}\|_\infty \leq \tau \\ & && K \text{ stabilizing} \end{aligned}$$

is solved by

$$\widehat{K}(s) = \frac{\|(A^{-1}B)^\dagger\|}{\tau}\left(B^TA^{-2} - \frac{1}{s}B^TA^{-1}\right).$$

provided that $\tau \geq \sqrt{\|B^TA^{-4}B\|}$.

Optimal Network Control with Edge Integrators



Given a graph $(\mathcal{V}, \mathcal{E})$, let $P(s)$ be the transfer matrix from u to x given by $\dot{x}_i = a_i x_i + \sum_{(i,j)\in\mathcal{E}}(u_{ij} - u_{ji})$, $i \in \mathcal{V}$ with $a_i < 0$. Then $\widehat{K}(s)$ is a separate PI controller for each graph edge:

$$\begin{cases} \dot{z}_{ij} = k(x_i/a_i - x_j/a_j) \\ u_{ij} = z_{ij} - x_i/a_i^2 + x_j/a_j^2 \end{cases}$$

(Works if the graph is a tree!)

Optimal Network Control with Node Integrators

Given a graph $(\mathcal{V}, \mathcal{E})$, let the plant be given by

$$\dot{x}_i = a_i x_i + b_i u_i + \sum_{(i,j)\in\mathcal{E}} (u_{ij} - u_{ji}) \qquad i \in \mathcal{V}$$

with $a_i < 0$. Then $\widehat{K}(s)$ is the map from x to u given by

$$\begin{cases} \dot{z}_i = x_i \\ u_{ij} = z_i/a_i - x_i/a_i^2 - z_j/a_j + x_j/a_j^2 \\ u_i = b_i(z_i/a_i - x_i/a_i^2) \end{cases}$$

(There is a problem if all b_i are zero. Why?)

Limitations due to Graph Structure

Let $A = -I$ while B is an oriented incidence matrix, e.g.

$$B = \begin{bmatrix} 1 & 0 \\ -1 & 1 \\ 0 & -1 \end{bmatrix}.$$

If the graph is a tree, B has full column rank and

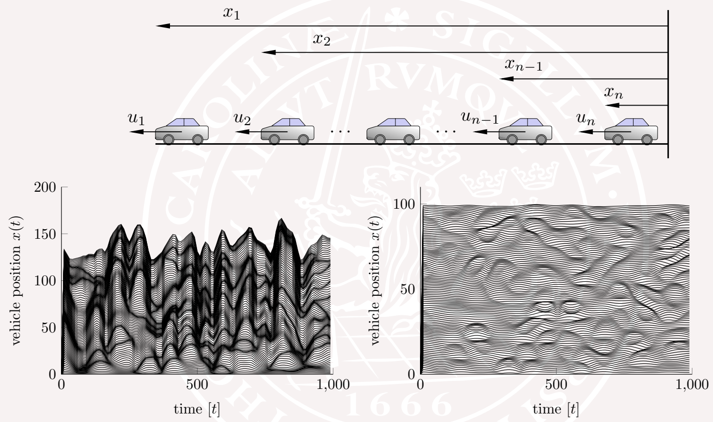
$$\|F_{\widehat{K}}\|_\infty = \sqrt{\|(B^T B)^{-1}\|} = \frac{1}{\sqrt{\lambda_2}},$$

where λ_2 is the algebraic connectivity of the graph. In two extreme cases, a star graph and a one-dimensional path, λ_2 is

$$1 \quad \text{and} \quad \left(2\sin\frac{\pi}{2n}\right)^2$$

respectively.

Relative versus absolute measurements



Centralized information does not help, as long as only local relative measurements are available!

Outline

- Network realizable control
  - Don't look for sparse transfer functions
  - Connection to Internal Model Control
- Scalable H_∞ optimal synthesis
  - H_∞ optimal static controllers
  - H_∞ optimal dynamic controllers
- **Scalable H_2 optimal synthesis**
  - a transportation example
- Concluding remarks

$\mathbf{H}_2$ Optimal Control

1. Optimal state feedback gains are dense.
2. They are unique.

Bad for large scale problems!

$\mathbf{H}_2$ Optimal Control

1. Optimal state feedback gains are dense.
2. They are unique.

Bad for large scale problems!

Why do we want sparse feedback matrices?

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

Why do we want sparse feedback matrices?

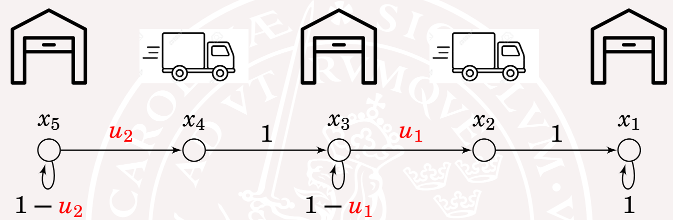
$$u = \begin{bmatrix} & * & & & & * & & & & * & & * & & * & \\ * & * & & & & & & & & & * & * & * & * & \\ & & & & & & & & & & * & * & & * & \\ & & & * & & & & * & & & & & & * & * \\ * & * & & * & & * & & * & & * & * & * & & & * \\ & & & & & * & * & & & * & & & & * & * \\ * & * & * & & & & & * & & * &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

Sparsity $\Longrightarrow$ Distributed computation of u.

Why do we want sparse feedback matrices?

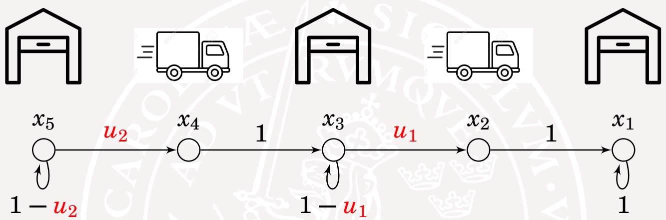
Sparsity $\not\Longleftarrow$ Distributed computation of u.

A Curious Example



European Control Conference, Limassol, Cyprus 2018
M. Heyden, R. Pates, A. Rantzer
A Structured Linear Quadratic Controller for Transportation Problems

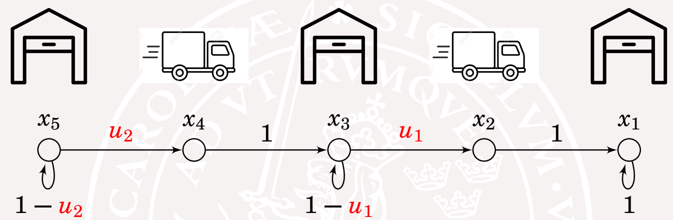
A Curious Example



$$\min_{u[0],u[1],\dots} \sum_{k=0}^{\infty} a^{2k}\left(x_1[k]^2 + x_3[k]^2 + x_5[k]^2\right)$$

$$\text{s.t. } x[k+1] = \begin{bmatrix} 1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix} x[k] + \begin{bmatrix} 0 & 0 \\ 1 & 0 \\ -1 & 0 \\ 0 & 1 \\ 0 & -1 \end{bmatrix} u[k],\ x[0] \in \mathbb{R}^5.$$

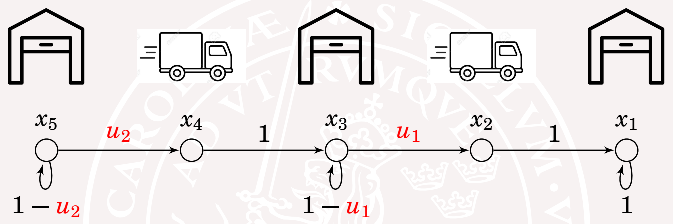
A Curious Example



Solve:

$$a^2A^TXA - X - aA^TXB\underbrace{a\left(B^TXB + R\right)^{-1}B^TXA}_{K_{\text{opt}}} + Q = 0.$$

A Curious Example



There is a pattern...

$$K_{\text{opt}} = \begin{bmatrix} * & * & * & * & 0 & 0 & 0 & 0 & 0 \\ * & * & * & * & * & * & 0 & 0 & 0 \\ * & * & * & * & * & * & * & * & 0 \\ * & * & * & * & * & * & * & * & * \end{bmatrix}$$

Why do we want sparse feedback matrices?

$$K_{\text{opt}} = \begin{bmatrix} a^2+1 & 0 \\ 0 & a^4+a^2+1 \end{bmatrix}^{-1} \begin{bmatrix} -a^3 & -a^3 & a & a & 0 \\ -a^5 & -a^5 & -a^5 & -a^5 & a\left(a^2+1\right) \end{bmatrix}$$

$$= \begin{bmatrix} p_1(a) & 0 \\ -a^2 & \dfrac{p_2(a)}{p_1(a)} \end{bmatrix}^{-1} \begin{bmatrix} -a^3 & -a^3 & a & a & 0 \\ 0 & 0 & -a^3 & -a^3 & a \end{bmatrix}$$

Why do we want sparse feedback matrices?

This pattern continues

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

Distributed feedback law:

$$u_i = -\frac{p_i(a)}{p_{i-1}(a)} \left(a^{-3}x_{2i-1} + a^{-3}x_{2i} - ax_{2i+1} - ax_{2i+2} + a^2 u_{i-1}\right).$$

No sparse feedback matrix!!!

Why do we want sparse feedback matrices?

This pattern continues

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

Distributed feedback law:

$$u_i = -\frac{p_i(a)}{p_{i-1}(a)} \left(a^{-3}x_{2i-1} + a^{-3}x_{2i} - ax_{2i+1} - ax_{2i+2} + a^2 u_{i-1}\right).$$

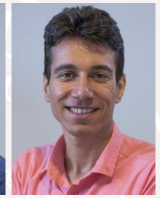
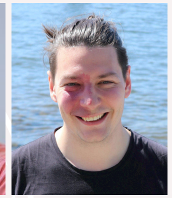
No sparse feedback matrix!!!

Conclusions

- Network realizable control
  - Don't look for sparse transfer functions
  - Connection to Internal Model Control
- Scalable H_∞ optimal synthesis
  - H_∞ optimal static controllers
  - H_∞ optimal dynamic controllers
- Scalable H_2 optimal synthesis
  - a transportation example
- Concluding remarks

Thanks



Carolina Bergeling Richard Pates Martin Heyden Hamed Sadeghi Rijad Alisic

Anders Rantzer Towards a Theory of Scalable Control