Optimal Network Realizable Controllers for Networked Systems

Nicola Elia
Joint work with
Andalam, Satya Mohan Vamsi

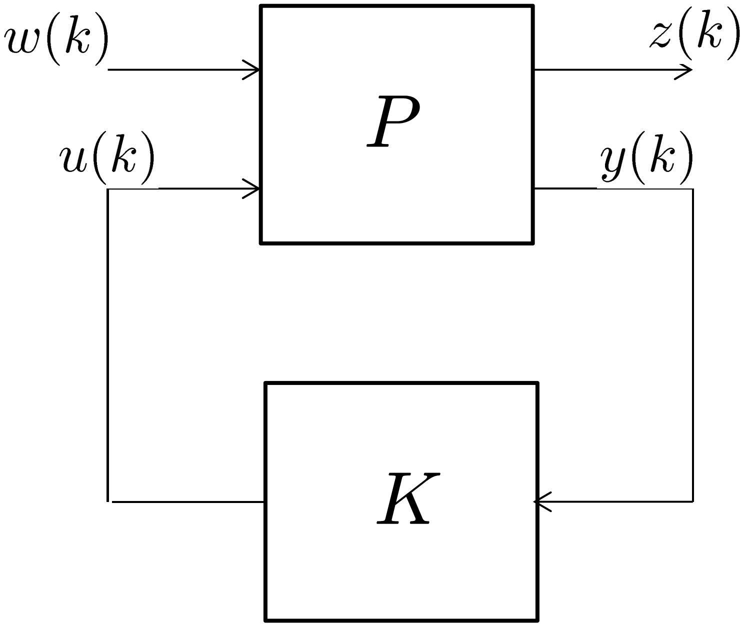
Dept. of Electrical and Computer Engineering
Iowa State University

Acknowledgments to: NSF

Outline

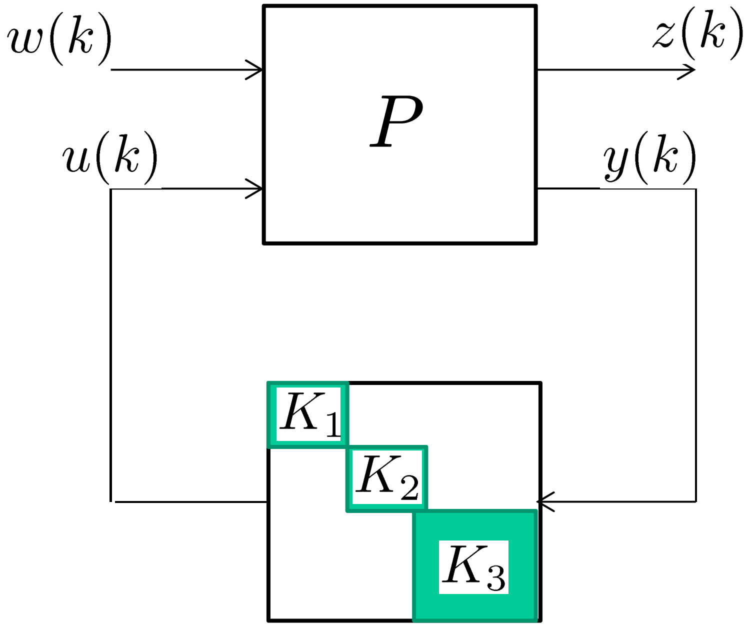
- Networked systems (Formulation)
- Representation (state-space or input-output)
- Networked systems vs structured systems
- Networked system realization
- Networked controller design for Networked systems
- All stabilizing (realizable) networked controllers
- Optimal networked controllers (realizable) on arbitrary networks
- Application: distributed optimization systems

Controller Design Problems



Classical centralized control problem

Controller Design Problems

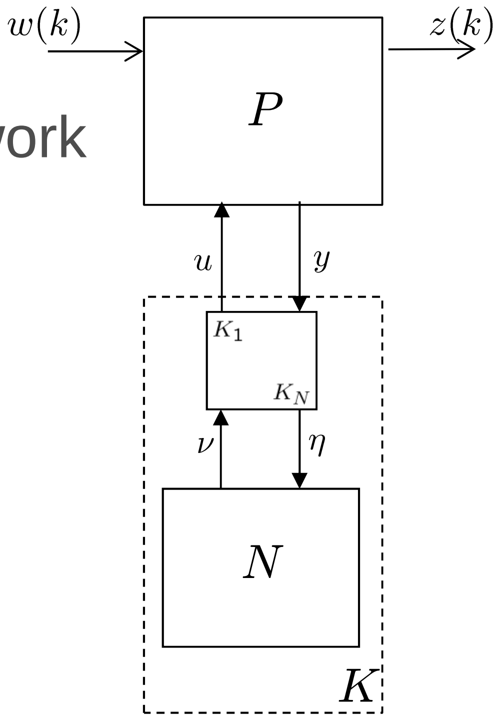


Decentralized control problem

Blondel and Tsitsiklis - Problem of finding a stabilizing decentralized static output feedback is NP-hard.

Controller Design Problems

- Distributed Controller
- Controllers communicate over network
- Still difficult for general plants



Searching over structures

- Looking for and identifying conveniently searchable structures, in the system state-space or input-output representation, has been the focus of most research in distributed control problems.
- Networked systems and structured systems have become so intertwined that they are often identified with each other.
- However it is not always clear when a structured state-space or transfer function, consistent with a given network, truly represents a networked system composed of subsystems interacting over a network.
- In contrast, it is easy to verify that both the state-space representation and the transfer function of networked systems inherit certain structures.

Some Previous Work on input-output models

- Voulgaris - For specific structural constraints (like lower block triangular and band structures), the problem was shown to be convex when the plant also follows the structural constraints.
- Hovd and Skogestad - Solved for symmetrically interconnected systems.
- Bamieh, Paganini and Dahleh - Solved for spatially-invariant systems.
- Gattami - Solved for dynamically coupled systems over arbitrary graphs. Modeled the plant in state-space but the controller in input-output.
- M. Rotkowitz and Sanjay Lall - Introduced quadratic invariance to address all the above cases (Even when plant and controller do not share same network constraints).

Focused on optimal searching over *structured transfer functions*
Provide controllers with transfer functions satisfying network constraints
Do not provide any explicit way to implement the controller
as a set of subsystems connected over the given network

Some Previous Work on state-space models

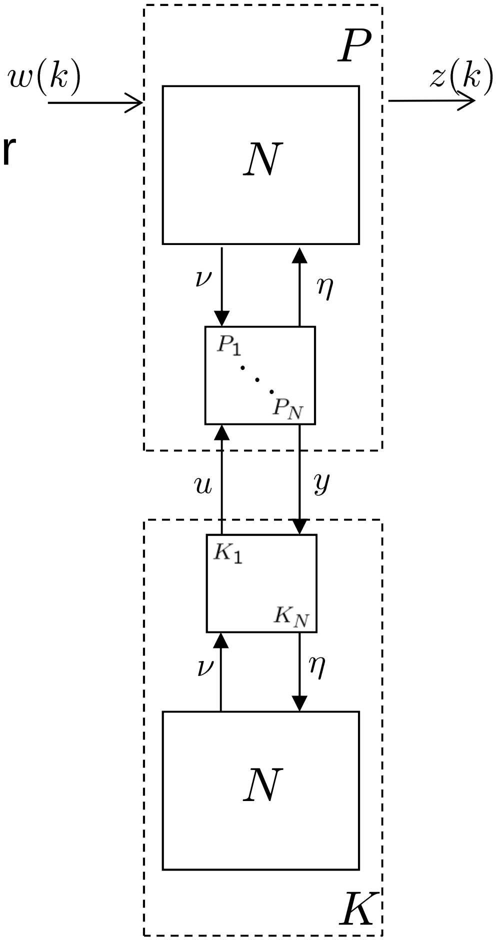
- C. Langbort, R. S. Chandra, and R. D'Andrea - Distributed controller design for systems interconnected over arbitrary graph.
- P. Massioni and M. Verhaegen - Distributed control for identical dynamically coupled decomposable systems.
- Parikshit and Parillo - Poset-causal systems over acyclic networks.
- Swigart and Sanjay Lall - Networked systems over acyclic networks.

Mostly, the state-space based approaches only have sufficiency conditions to obtain a distributed controller which lead to sub-optimal solution

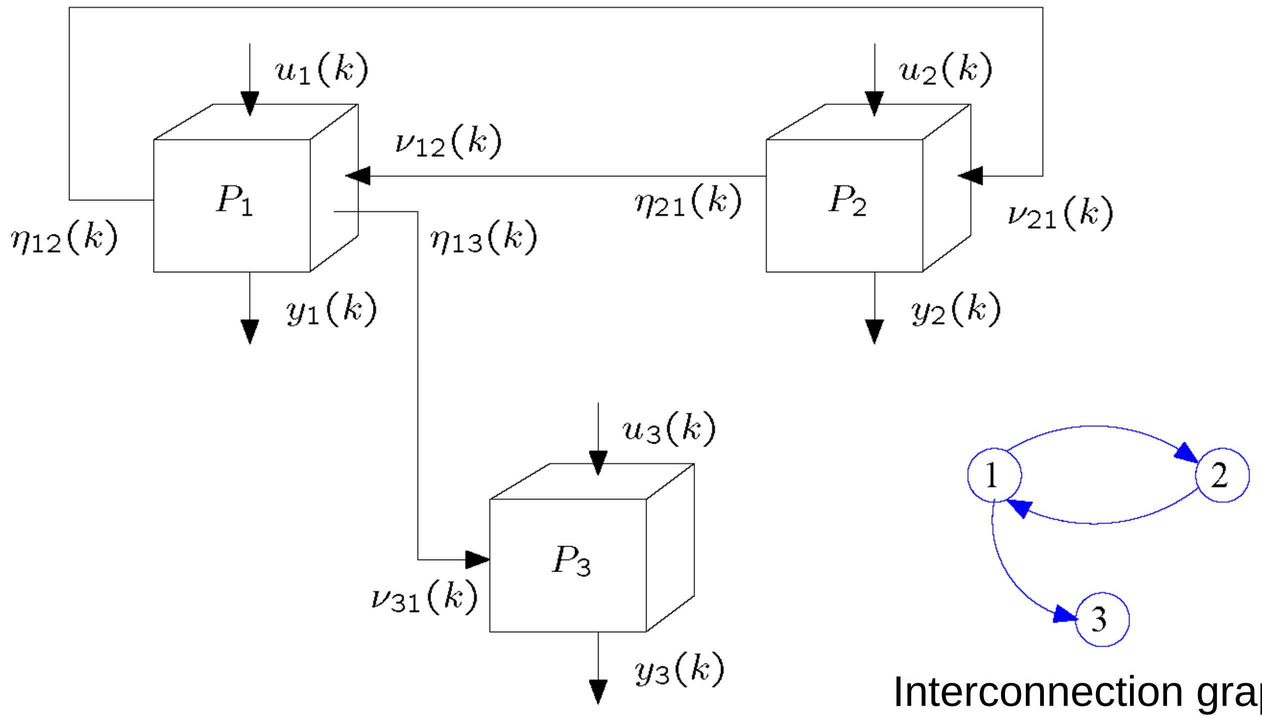
Controller Design Problem

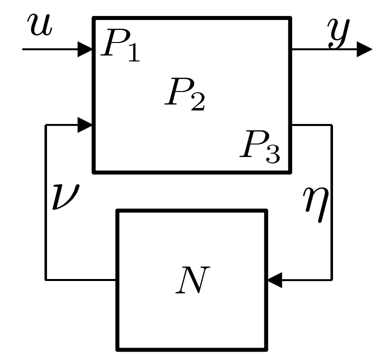
Networked distributed controller for Networked distributed plants

- Both plant and controller are distributed over the same network (This talk)



Networked Systems

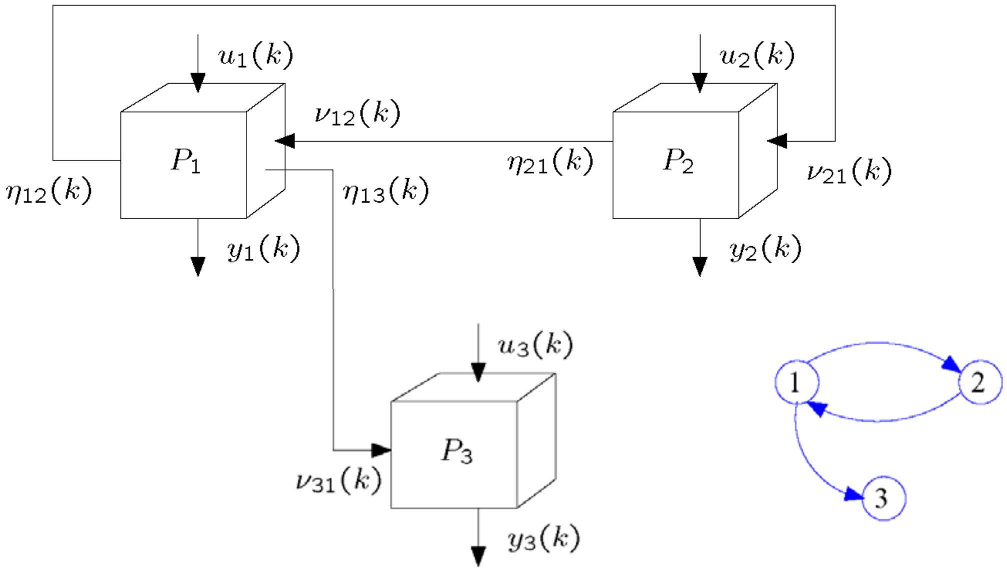


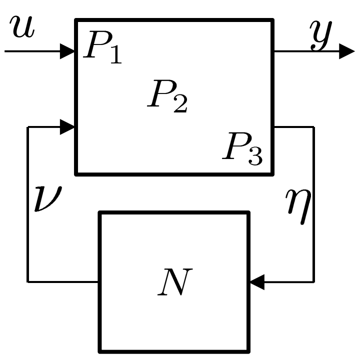


Interconnection graph

1. Sub-systems dynamics (DT-LTI)
2. Network interconnection graph
3. Messages/signals exchanged over the network
4. Local measurements and control inputs

Networked systems (without internal instability)

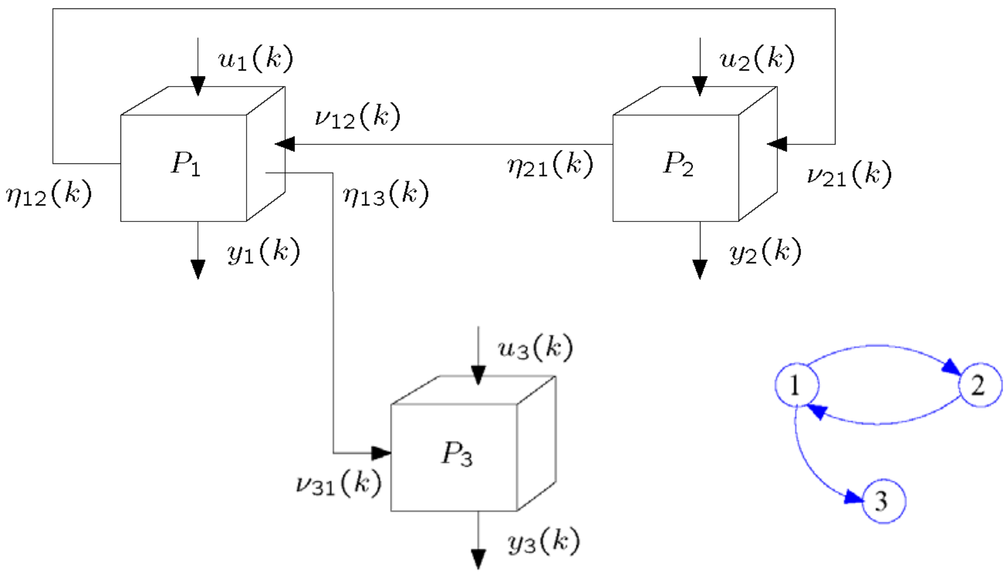


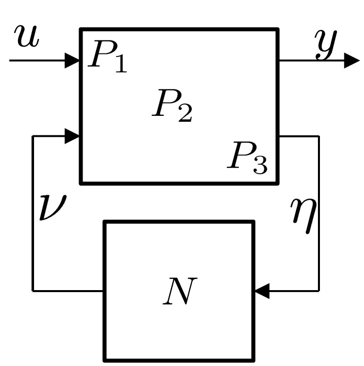


Assumtions: Feedback interconnection
1) Well-posed
2) Stabilizable from u and detectable from y

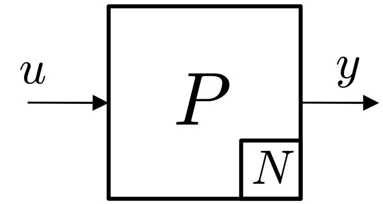
$\mathcal{N}^I(\mathcal{G}, \mathcal{P}_u, \mathcal{P}_y)$ denotes the set of all such *implementable* networked systems

Networked system representation





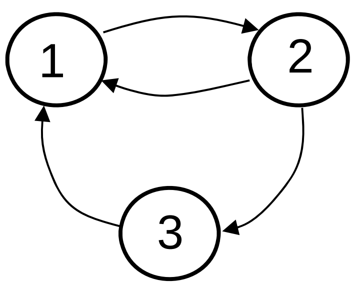
State-space or Input-Output



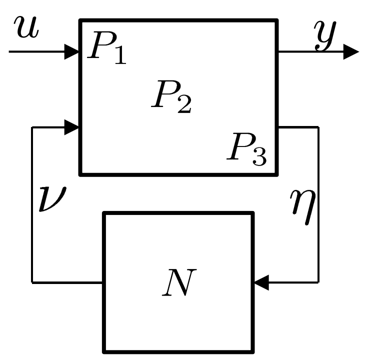
Network vs. Information Topology

Network topology: graph of physical interconnections among nodes
Information topology: graph of the information available to the nodes

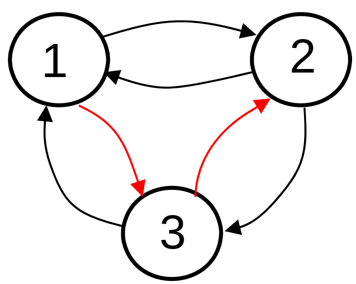
Mismatched if nodes on the networks are allowed to instantaneously relay information from their in-neighbors to their out-neighbors



Network topology



LFT interaction
Nodes & Network



Information topology
If nodes are instant relays

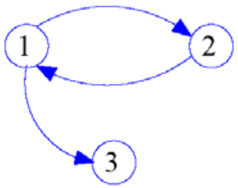
- Mismatch is a source of confusion in the literature
- SS or I/O structures of networked systems depends on the information topology

Strictly Causal Network interconnection

Sub-systems dynamics

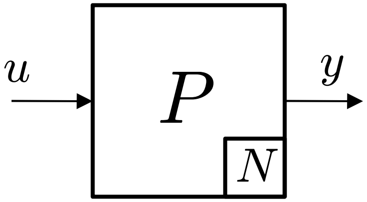
$$P_i = \begin{bmatrix} x_i(k+1) \\ y_i(k) \\ \eta_i(k) \end{bmatrix} = \begin{bmatrix} A^i & B_u^i & B_\nu^i \\ C_y^i & D_{yu}^i & D_{y\nu}^i \\ C_\eta^i & 0 & 0 \end{bmatrix} \begin{bmatrix} x_i(k) \\ u_i(k) \\ \nu_i(k) \end{bmatrix}$$

- We call the communication model a *strictly causal network interconnection* *if* the sub-systems can only pass their local state information to their neighbors at each discrete time step.
- Two nodes on a graph that are not directed neighbors cannot exchange their local information in the same time instant.
- The links are considered to be noiseless, delay-free and with no bandwidth constraints. $\nu_{ij}(k) = \eta_{ji}(k)$



Network topology corresponds to information topology !

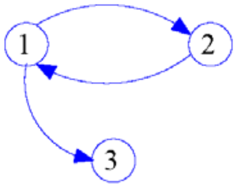
Structured State-Space representation



State-space structure

$$P = ss(A, B_u, C_y, D_{yu})$$

$\mathcal{G}$



$$A = \begin{bmatrix} A_{11} & A_{12} & 0 \\ A_{21} & A_{22} & 0 \\ A_{31} & 0 & A_{33} \end{bmatrix}$$

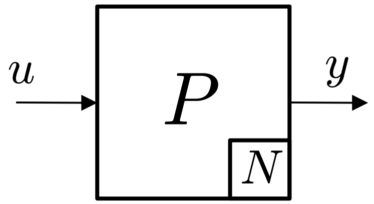
A and C_y are structured according to the network B_u and D_{yu} are block diagonal

- *Sparsity* – no directed path from node 3 to node 1 and 2 & from node 2 to node 3

$\mathcal{S}(\mathcal{G}, \mathcal{P}_u, \mathcal{P}_y)$ denotes the set of all SS structured according to the graph $\mathcal{G}$ and given input output partitions.

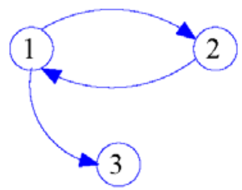
$\mathcal{S}^S(\mathcal{G}, \mathcal{P}_u, \mathcal{P}_y)$ are the stable ones

Structured Input-Output representation



Transfer Function structure

$$P(z) = \begin{bmatrix} H_{11}(z) & z^{-1}H_{12}(z) & 0 \\ z^{-1}H_{21}(z) & H_{22}(z) & 0 \\ z^{-1}H_{31}(z) & z^{-2}H_{32}(z) & H_{33}(z) \end{bmatrix}$$



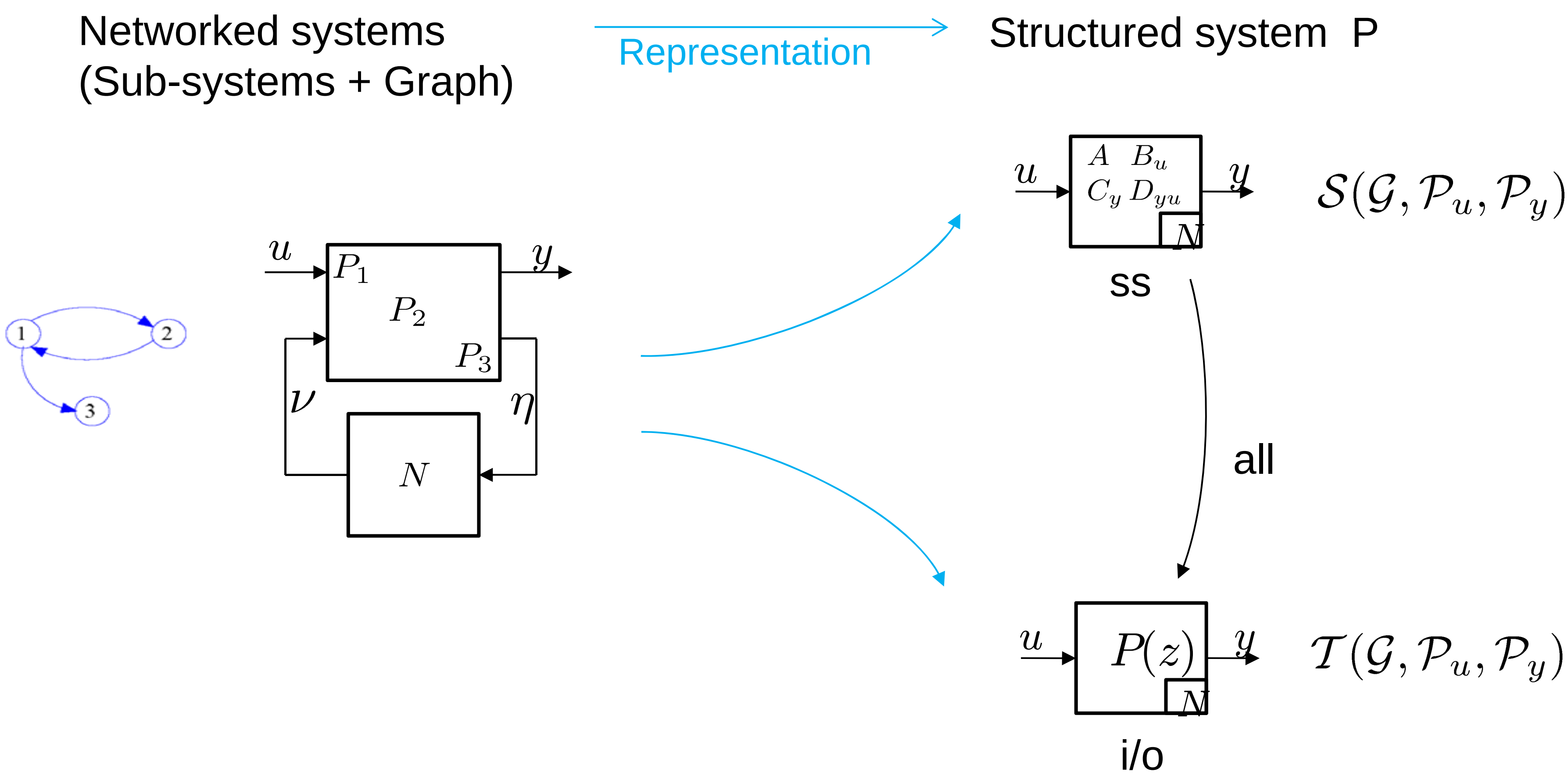
$H_{ij}(z)$ real proper rational transfer function matrices

- *Delay* - shortest path from node 2 to node 3 is equal to 2
- *Sparsity* – no directed path from node 3 to node 1 and node 2.

$\mathcal{T}(\mathcal{G}, \mathcal{P}_u, \mathcal{P}_y)$ denotes the set of all transfer functions matrices structured according to the graph $\mathcal{G}$ and given input and output partitions.

$\mathcal{T}^S(\mathcal{G}, \mathcal{P}_u, \mathcal{P}_y)$ are the stable ones

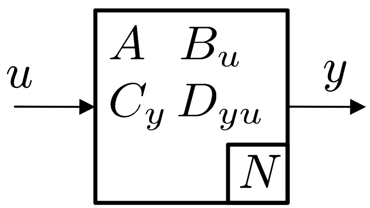
Structured systems



- Structured systems may be efficiently searched over
- Many result based on searching and designing over I/O structures

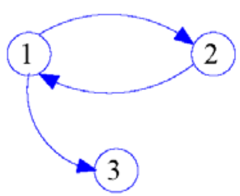
Networked system implementation

Given

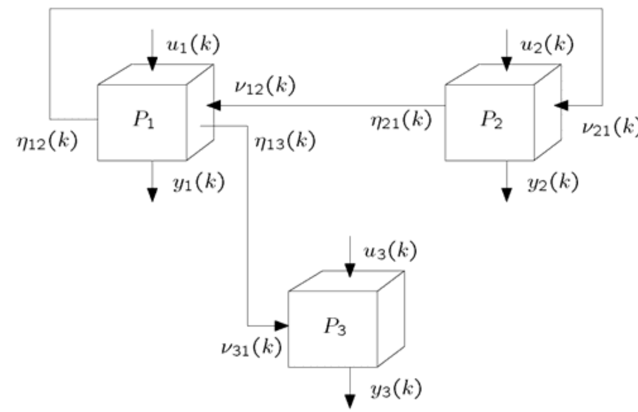


$\mathcal{S}(\mathcal{G},\mathcal{P}_u,\mathcal{P}_y)$

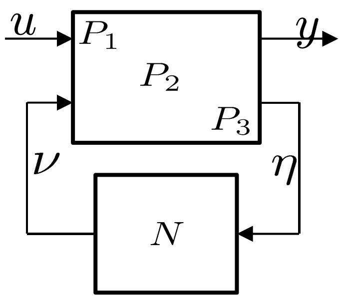
Stab. & Detec.



Find P_i



Stab. & Detec.

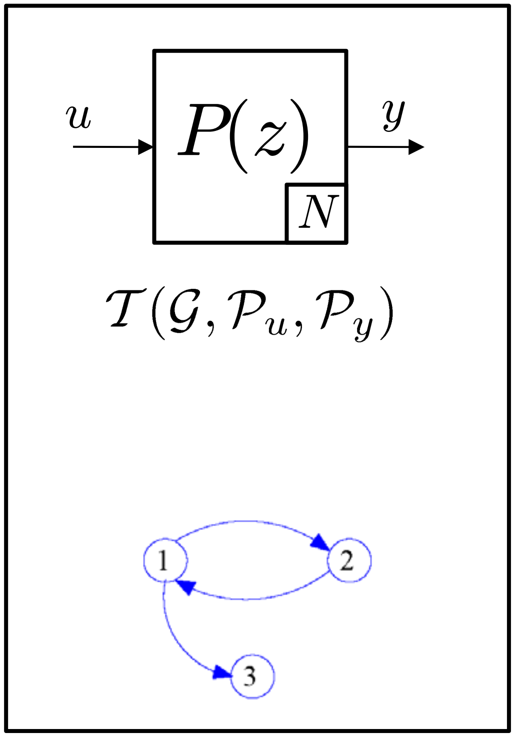


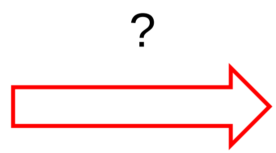
Networked implementability

$P \in \mathcal{S}(\mathcal{G},\mathcal{P}_u,\mathcal{P}_y)$ stab.&detec. is network-implementable if $P \in \mathcal{N}^I(\mathcal{G},\mathcal{P}_u,\mathcal{P}_y)$

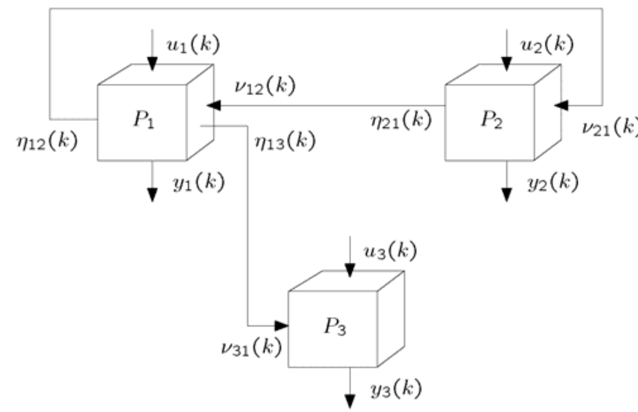
Networked system realization

Given

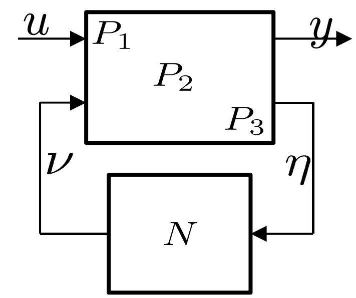




Find P_i



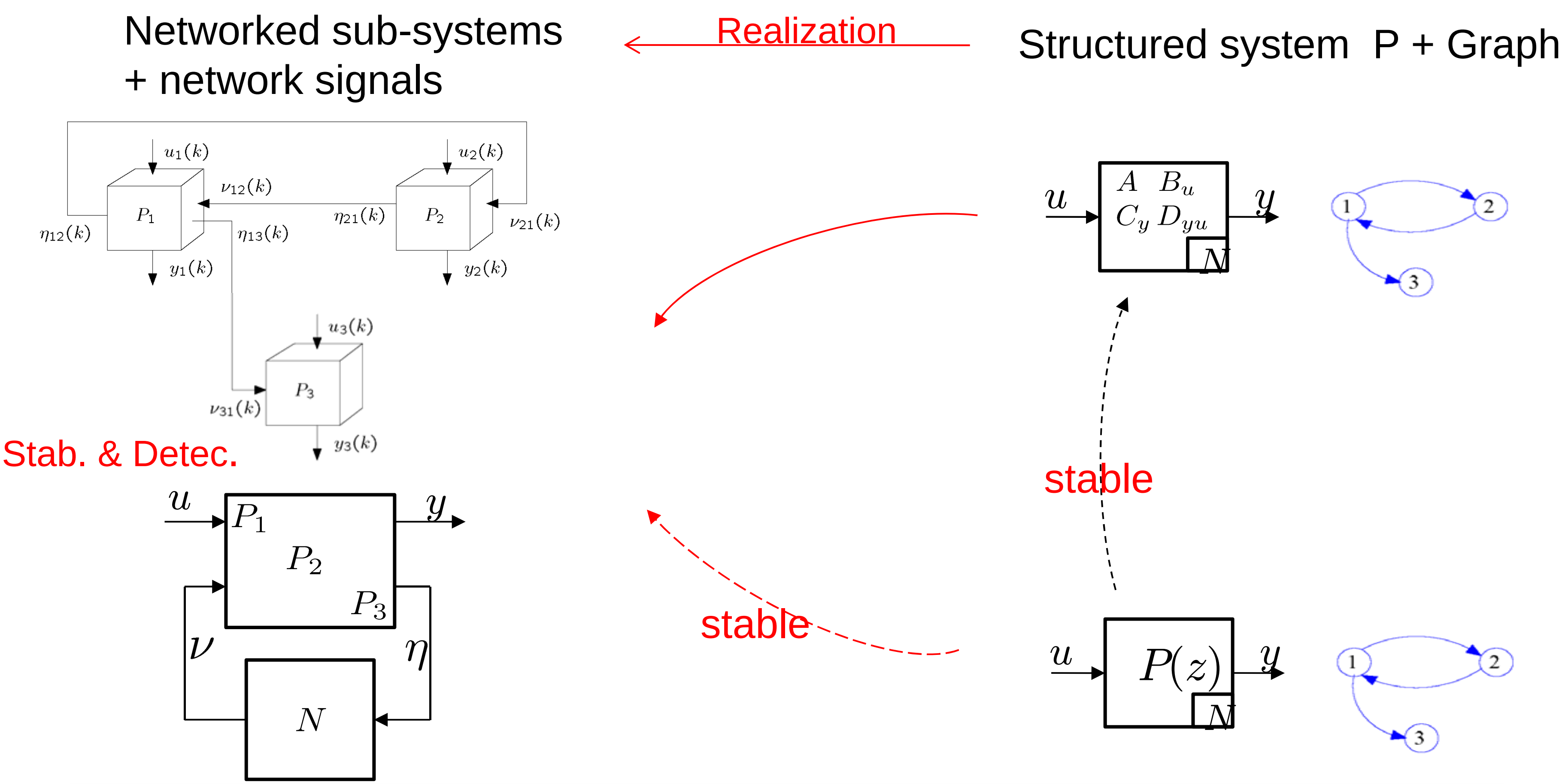
Stab. & Detec.



Networked realizability

$P(z) \in \mathcal{T}(\mathcal{G}, \mathcal{P}_u, \mathcal{P}_y)$ is network-reliazable, if can find $\tilde{P} \in \mathcal{N}^I(\mathcal{G}, \mathcal{P}_u, \mathcal{P}_y)$ with $P(z) = \tilde{P}(z)$

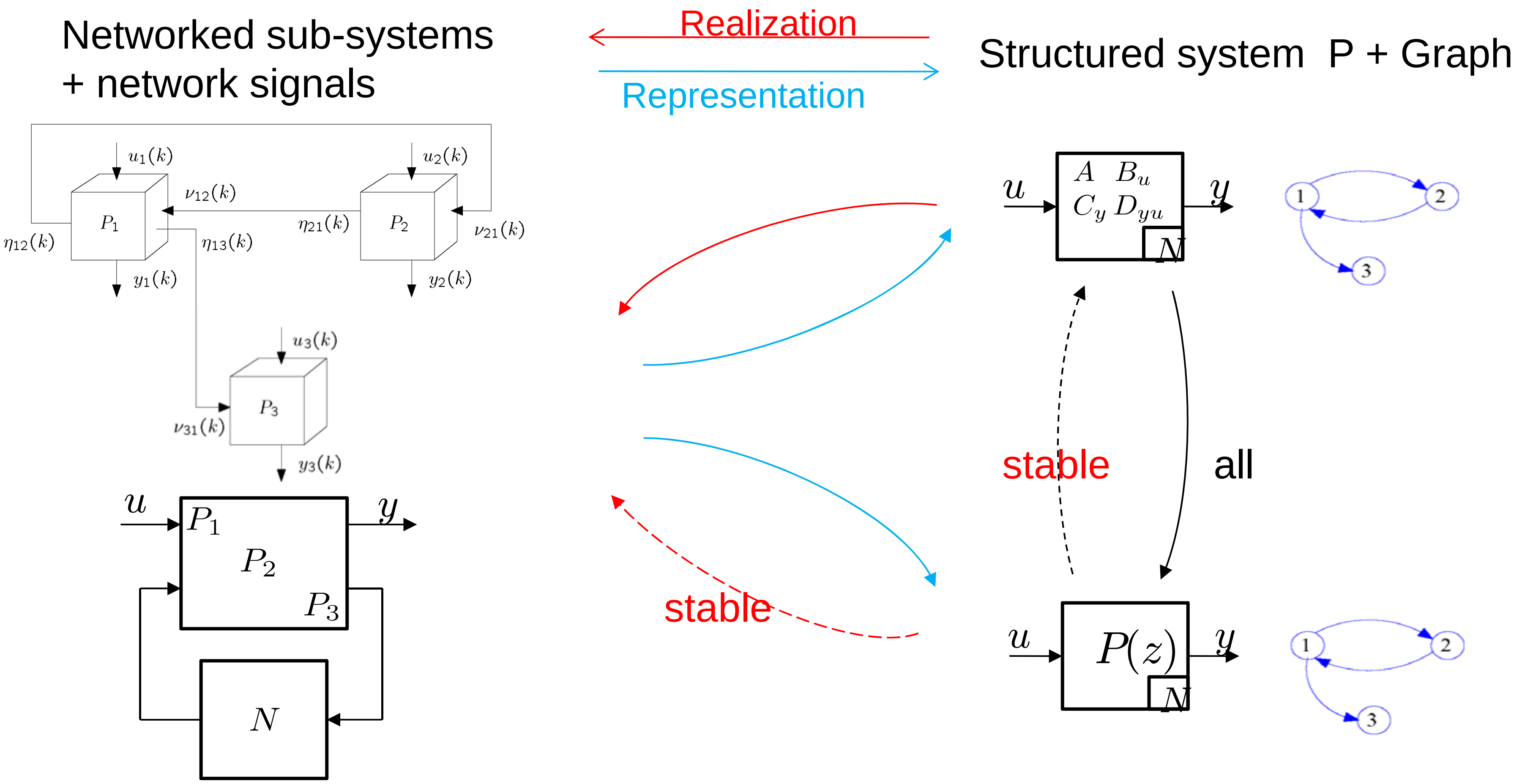
Realizability over networks



Main results

- State-space structured according to the network is always implementable
- I/O with sparsity&delays according to the network is realizable if stable

Realizability over networks



Stab. & Detec.

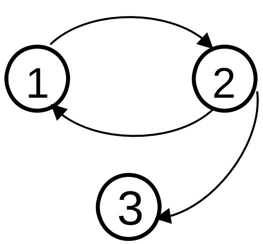
Network realizability for unstable I/O maps?

- Is not known how to realize the networked system when $P(z)$ (with sparsity & delay structure according to the network) is unstable.

- Realizability of systems over networks has been overlooked.

- Optimal structuerd controllers can be unstable. How to build the, as interconnected systems without introducing internal instability?

$$K(z) = \begin{bmatrix} -\frac{z-2}{(z-0.8625)(z-4.637)} & \frac{2(z-0.5)}{(z-0.8625)(z-4.637)} & 0 \\ \frac{z-5}{(z-0.8625)(z-4.637)} & \frac{(z-3.5)}{(z-0.8625)(z-4.637)} & 0 \\ \frac{z-0.5}{(z-0.8625)(z-4.637)(z-1.5)} & -\frac{z-0.5}{(z-0.8625)(z-4.637)} & -\frac{1}{z-1.5} \end{bmatrix}$$

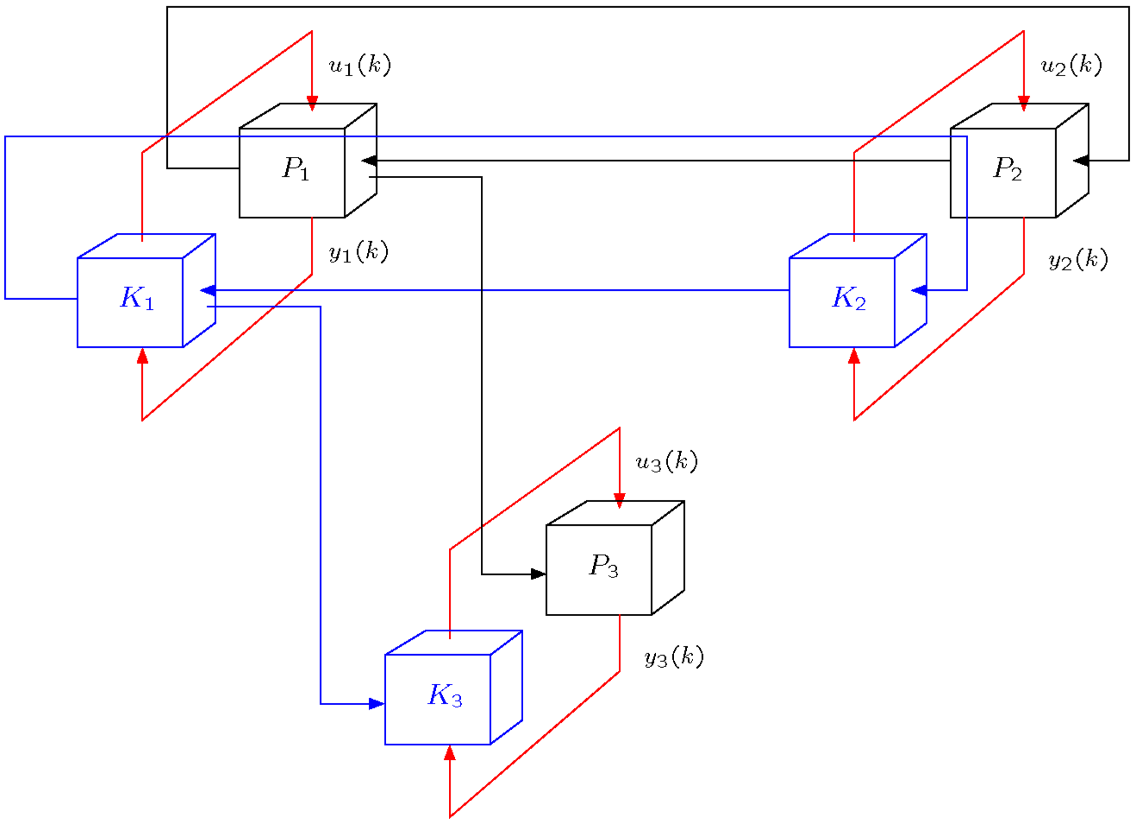


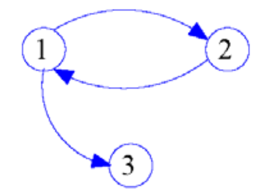
Only two unstable poles one at 1.5 the other at 4.637 !

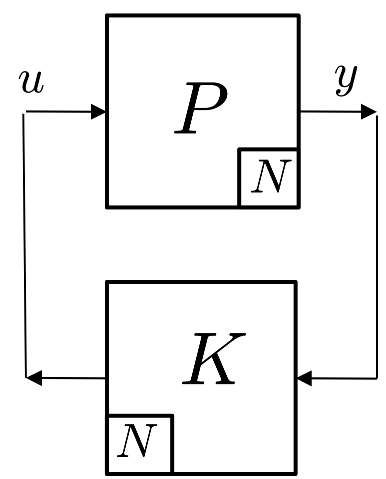
Realizing single TF leads to replication of unstable poles

Networked controller design problem

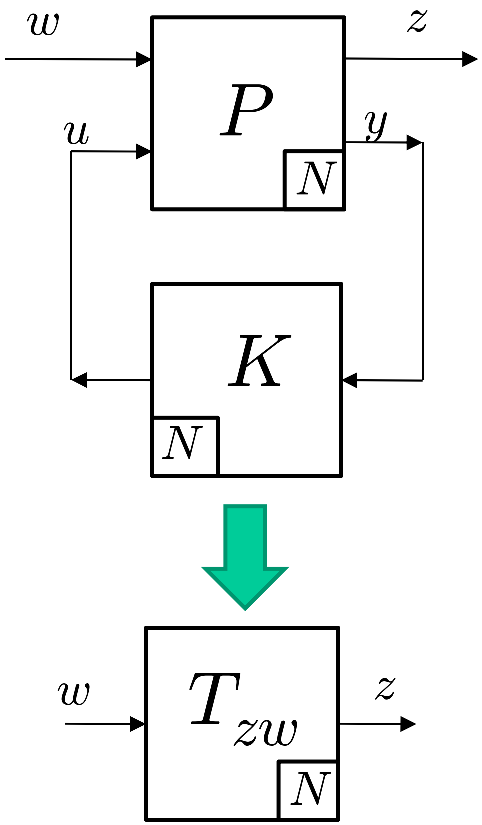
Given a networked plant P on a network N, design a stabilizing networked controller on the same network.







Networked systems in feedback



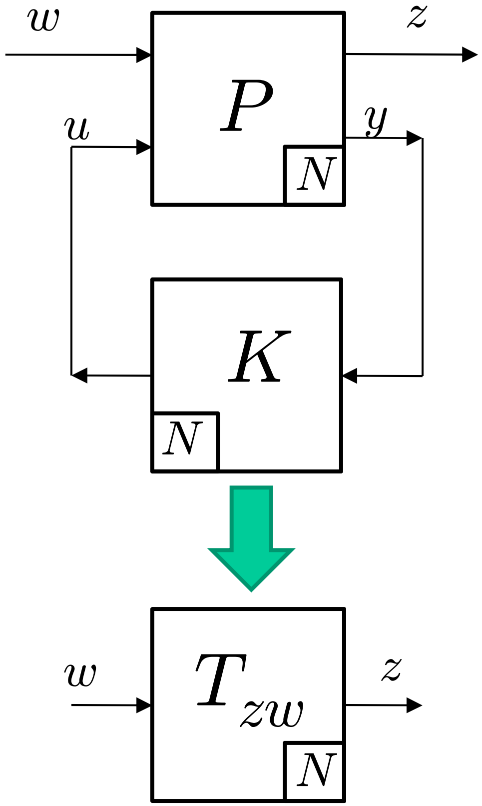
Assume $P_i = \begin{bmatrix} A^i & B_w^i & B_u^i & B_\nu^i \\ C_z^i & D_{zw}^i & D_{zu}^i & D_{z\nu}^i \\ C_y^i & D_{yw}^i & 0 & D_{y\nu}^i \\ C_\eta^i & 0 & 0 & 0 \end{bmatrix}$

So that $P = \begin{bmatrix} A & B_w & B_u \\ C_z & D_{zw} & D_{zu} \\ C_y & D_{yw} & 0 \end{bmatrix}$

with A, C_y, C_z, D_{zw} structured according to the network and B_u, B_w, D_{zu}, D_{yw} block-diagonal

The closed loop system , T_{zw} is network realizable for any stabilizing networked K .

Closed Loop System



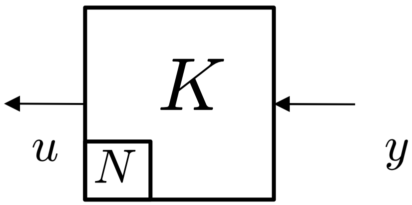
$$P = \begin{bmatrix} A & B_w & B_u \\ C_z & D_{zw} & D_{zu} \\ C_y & D_{yw} & 0 \end{bmatrix}$$

$$K = \begin{bmatrix} A_K & B_K \\ C_K & D_K \end{bmatrix}$$

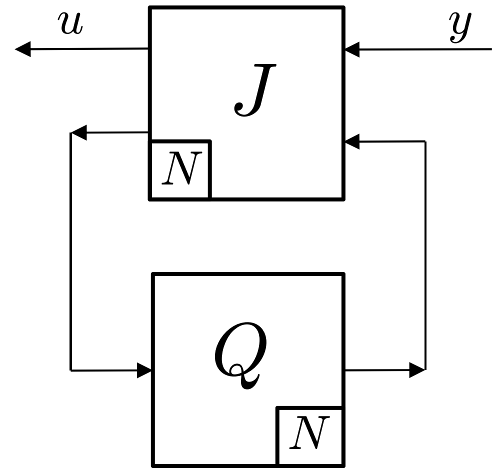
$$T_{zw} = \left[\begin{array}{cc|c} A + B_u D_K C_y & B_u C_K & B_w + B_u D_K D_{yw} \\ B_K C_y & A_K & B_K D_{yw} \\ \hline C_z + D_{zu} D_K C_y & D_{zu} C_K & D_{zw} + D_{zu} D_K D_{yw} \end{array}\right]$$

All stabilizing network realizable controllers

Main Theorem: All network realizable stabilizing controllers, K, for P have the form:



=



$$J = \left[\begin{array}{c|cc} A + B_u F + L C_y & -L & -B_u \\ \hline F & 0 & -I \\ -C_y & I & 0 \end{array}\right]$$

where Q is any stable network realizable system,

if F, is structured and L is block diagonal with A+BF and A+LC Schur stable.

(Sufficient condition to find F and L based on structured LMIs)

F=L=0 if plant is stable.

Conditions for F and L

Based on relaxed LMI condition for stability [De Olivera et.al]

A Schur stable iff can find $M = M',\, M > 0,\, G \in \mathbb{R}^{n\times n}$ satisfying

$$\begin{bmatrix} M & AG \\ G'A' & G + G' - M \end{bmatrix} > 0$$

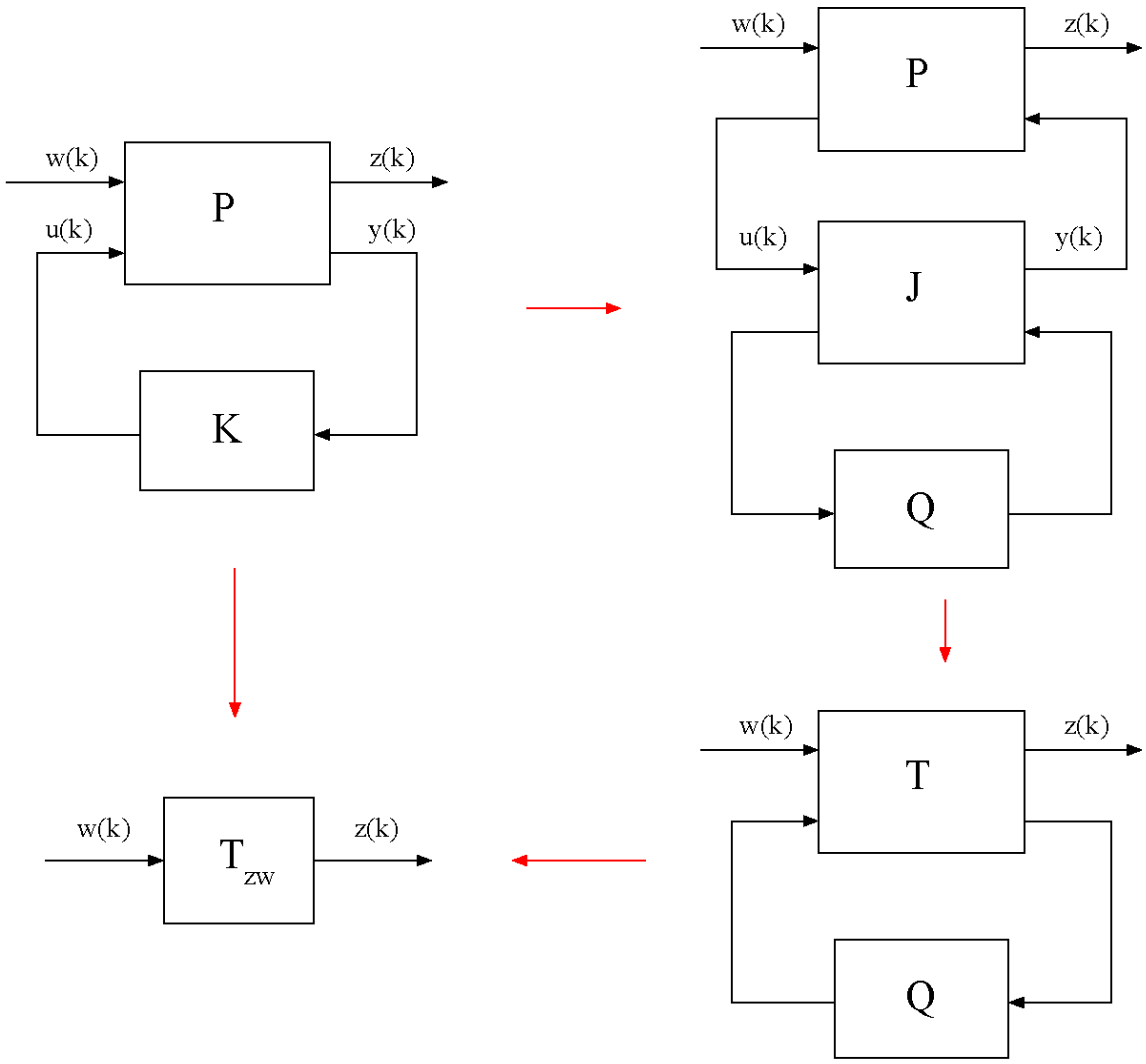
There exist F structured according to G such that $A+B_uF$ is Schur stable if can find $M = M',\, M > 0,\, G \in \mathbb{R}^{n\times n},\, R$ structured according to $\mathcal{G}$

$$\begin{bmatrix} M & AG + B_uR \\ (AG + B_uR)' & G + G' - M \end{bmatrix} > 0 \qquad F = RG^{-1}$$

There exist F structured according to G such that $A+LC_y$ is Schur stable if can find $M = M',\, M > 0,\, G \in \mathbb{R}^{n\times n},\, R$ block-diag $\mathcal{G}$

$$\begin{bmatrix} M & A'G + C_y'R \\ G'A + R'C_y & G + G' - M \end{bmatrix} > 0 \qquad L = (RG^{-1})'$$

Closed loop maps with parametrization



We can search over I/O Q stable with the network sparsity & delay structure

Optimal solution for networked H_2 problem

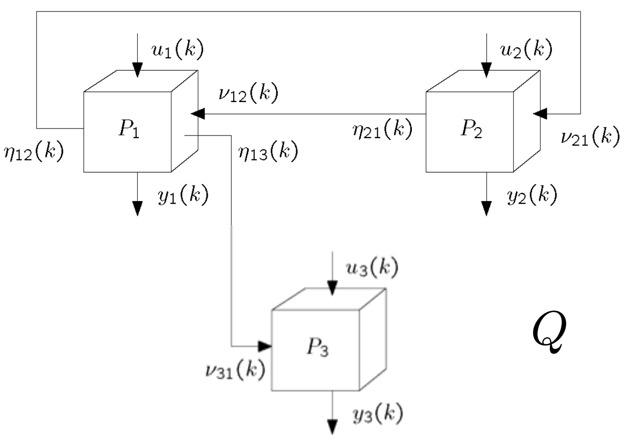
min $\|T_{zw}(K)\|_2$ = min $\|T_{11} + T_{12}QT_{21}\|_2$

$T_{zw}(K)$ internally stable

K network realizable

Q stable

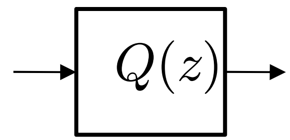
Q with network delay & sparsity structure

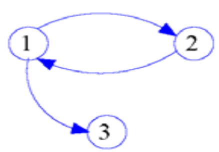


Q

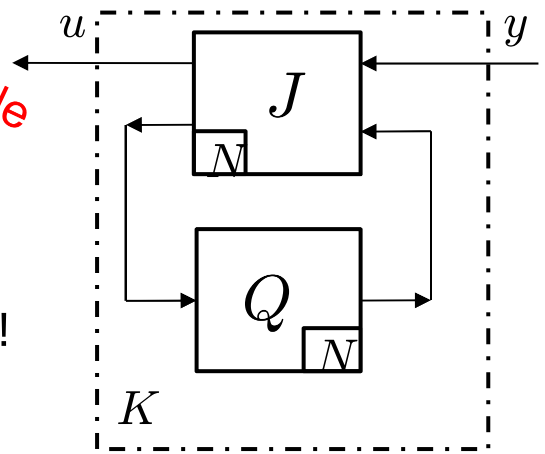
Realizable

stable





Implementable controller



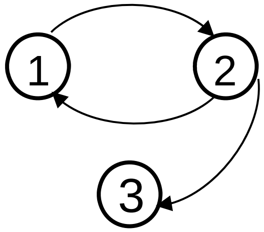
Optimal networked controller can be unstable!

Summary of approach

- Given a networked plant P in SS over a graph $\mathcal{G}$ with P_{22} stab&detec.
- Find gain F structured according $\mathcal{G}$ and L block-diag so that A+BF and A+LC are Schur stable.
- Compute networked J and networked T = LFT (P,J) in SS
- Convert to frequency domain $T(z)$
- Solve for optimal $Q(z)$ $$\min_{Q(z)\in\mathcal{T}^S(\mathcal{G},\mathcal{P}_y,\mathcal{P}_u)} \|T_{11}+T_{12}QT_{21}\|_2^2$$
- Realize $Q(z)$ as a networked system
- Obtain network-implementable K=LFT(J,Q)

Example

$$P_1\colon \left[\begin{array}{c} x_1(k+1) \\ \hline z_1(k) \\ \hline y_1(k) \\ \hline \eta_{12}(k) \\ \hline \eta_{13}(k) \end{array}\right] = \left[\begin{array}{cc|cc|c|cc} -0.6 & 0.3 & 0.05 & 0 & 0.05 & 1 & 0 \\ 0.5 & 0.2 & 0.2 & 0 & 0.2 & 0 & 1 \\ \hline -1.3 & 0.5 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ \hline -1.3 & 0.5 & 0 & 1 & 0 & 0 & 0 \\ \hline 1.4 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0.5 & 0.8 & 0 & 0 & 0 & 0 & 0 \\ \hline 1.1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0.9 & 0 & 0 & 0 & 0 & 0 \end{array}\right] \left[\begin{array}{c} x_1(k) \\ \hline w_1(k) \\ \hline u_1(k) \\ \hline \nu_1(k) \end{array}\right]$$



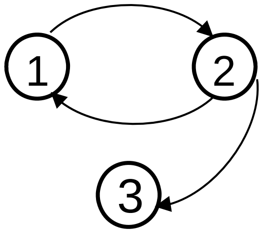
$$P_2\colon \left[\begin{array}{c} x_2(k+1) \\ \hline z_2(k) \\ \hline y_2(k) \\ \hline \eta_{21}(k) \end{array}\right] = \left[\begin{array}{cc|cc|c|cc} 0.1 & 0.8 & 0.1 & 0 & 0.1 & 1 & 0 \\ 0.3 & -0.5 & 0.1 & 0 & 0.1 & 0 & 1 \\ \hline 0.4 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ \hline 0.4 & 0 & 0 & 1 & 0 & 0 & 0 \\ \hline 1.2 & -0.7 & 0 & 0 & 0 & 0 & 0 \\ 0 & -1 & 0 & 0 & 0 & 0 & 0 \end{array}\right] \left[\begin{array}{c} x_2(k) \\ \hline w_2(k) \\ \hline u_2(k) \\ \hline \nu_2(k) \end{array}\right]$$

$$P_3\colon \left[\begin{array}{c} x_3(k+1) \\ \hline z_3(k) \\ \hline y_3(k) \end{array}\right] = \left[\begin{array}{cc|cc|c|cc} 0.2 & 0 & 0.4 & 0 & 0.4 & 1 & 0 \\ 0.3 & 0.4 & 0 & 0 & 0 & 0 & 1 \\ \hline 0 & 0.6 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ \hline 0 & 0.6 & 0 & 1 & 0 & 0 & 0 \end{array}\right] \left[\begin{array}{c} x_3(k) \\ \hline w_3(k) \\ \hline u_3(k) \\ \hline \nu_3(k) \end{array}\right]$$

Example (cont.)

$$A = \begin{bmatrix} 1.5 & -2 & 0 \\ -1 & 4 & 0 \\ 0 & 1 & 1.5 \end{bmatrix}, \quad C_z = \begin{bmatrix} -1 & 2 & 0 \\ 1 & 1 & 0 \\ 0 & -1 & -1 \end{bmatrix}, \quad C_y = \begin{bmatrix} -1 & 2 & 0 \\ 1 & 1 & 0 \\ 0 & -1 & -1 \end{bmatrix}$$

Centralized optimal controller $K_c = \begin{bmatrix} 1.1667 & -0.3333 & 0 \\ -1.6667 & -0.6667 & 0 \\ 0.1667 & 0.1667 & 1.5000 \end{bmatrix}.$



- The networked plant P is unstable.
- Need a networked stabilizing controller to start

$$F = \begin{bmatrix} -1.0351 & 2.0702 & \boxed{0} \\ 1.9185 & -3.8371 & \boxed{0} \\ \boxed{0} & -1.1356 & -1.1356 \end{bmatrix}$$

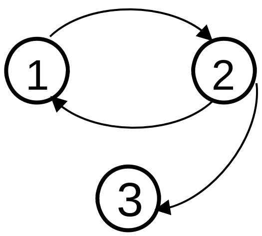
$$L = \begin{bmatrix} 1.0140 & 0 & 0 \\ 0 & -4.1139 & 0 \\ 0 & 0 & 1.0027 \end{bmatrix}$$

Higher controller order to compensate for lack of communication

- Centralized optimal cost 3
- Centralized optimal controller order 0
- Distributed optimal cost 8.8822
- Optimal distributed controller is unstable
- Sub-controllers order =[4,5,4]

The optimal distributed controllers

$$K_1 = \left[\begin{array}{cccc|c|ccccc}
1.3153 & 0.3761 & 0.0162 & 0.0003 & 0.8504 & 0 & 1 & 0 & 0 & 0 \\
-8.7760 & -0.5699 & -0.1741 & -0.0029 & -8.7760 & 0 & 0 & 1 & 0 & 0 \\
0.5454 & 2.0099 & 0.5923 & 0.0106 & 0.5454 & 0 & 0 & 0 & 1 & 0 \\
0.0097 & -0.4995 & 0.9859 & -0.0224 & 0.0097 & 0 & 0 & 0 & 0 & 1 \\
\hline
0.8292 & 0.3761 & 0.0162 & 0.0003 & 1.8643 & 1 & 0 & 0 & 0 & 0 \\
\hline
3.2751 & 0.6224 & -0.0089 & -0.0003 & 0 & 0 & 0 & 0 & 0 & 0 \\
-1.8388 & 0.6224 & -0.0089 & -0.0003 & 0 & 0 & 0 & 0 & 0 & 0 \\
-8.7927 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0.0658 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0.0103 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0.0741 & -0.0076 & 0.0009 & 0 & 0 & 0 & 0 & 0 & 0
\end{array}\right]$$



$$K_2 = \left[\begin{array}{ccccc|c|cccccc}
-2.5944 & 1.0699 & -0.0249 & -0.0118 & 0 & 2.7573 & 0 & 1 & 0 & 0 & 0 & 0 \\
-8.7927 & -0.6650 & -0.2102 & -0.1303 & 0 & 8.7927 & 0 & 0 & 1 & 0 & 0 & 0 \\
0.0658 & 1.9547 & 0.0814 & 0.0802 & 0 & -0.0658 & 0 & 0 & 0 & 1 & 0 & 0 \\
0.0103 & 0.5032 & 1.0182 & 0.5836 & 0 & -0.0103 & 0 & 0 & 0 & 0 & 1 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\
\hline
-2.4806 & 1.0699 & -0.0249 & -0.0118 & 0 & -1.3565 & 1 & 0 & 0 & 0 & 0 & 0 \\
\hline
-1.6585 & 0.6675 & 0.0302 & 0.0207 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
-1.6306 & 0.6675 & 0.0302 & 0.0207 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
17.5520 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
-0.0194 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
-1.091 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
\hline
0.2219 & -0.0909 & -0.0714 & 0.0129 & -1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0.2192 & -0.0909 & -0.0714 & 0.0129 & -1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
-8.6785 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0
\end{array}\right]$$

$$K_3 = \left[\begin{array}{cccc|c|ccc}
0.7192 & 0.0689 & 0.0018 & -0.0366 & 0.3548 & 0 & 1 & 0 \\
-8.6785 & -0.6965 & 0.3506 & -0.0258 & -8.6785 & 0 & 0 & 1 \\
0 & -1.4659 & 0.7539 & -0.0579 & 0 & 0 & 0 & 0 \\
0 & 0 & 0.2499 & -0.0575 & 0 & 0 & 0 & 0 \\
\hline
0.2219 & 0.0689 & 0.0018 & -0.0366 & 1.3575 & 1 & 0 & 0
\end{array}\right]$$

Distributed Least Squares over Noisy Channels

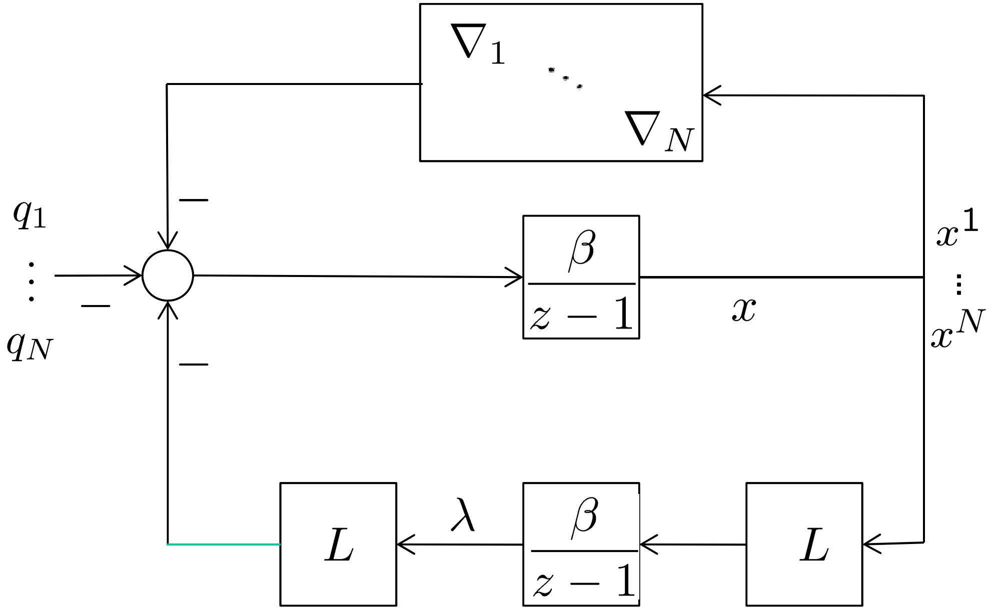
N sensors want to collectively learn $x \in \mathbb{R}^n$, (location of a target)

Each sensor has inaccurate incomplete (scalar) measurements

$$y_i = a_i^T x + v_i, \;\; v_i \sim N(0,1)$$

Problem: distributedly find the optimal ML estimate x^*

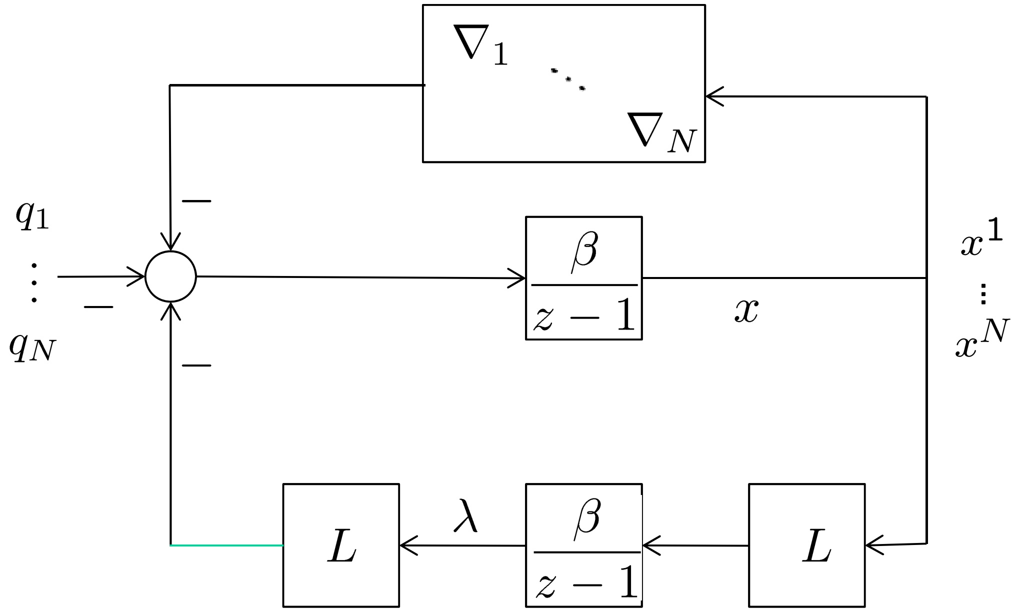
Solution : $$x^* = \arg\min_x \sum_{i=1}^{N} (a_i^T x - y_i)^2 = \arg\min_x \|Ax - y\|_2^2$$



$$\nabla_i = 2 a_i a_i^T$$
$$q_i = -2 y_i a_i$$

Least squares system

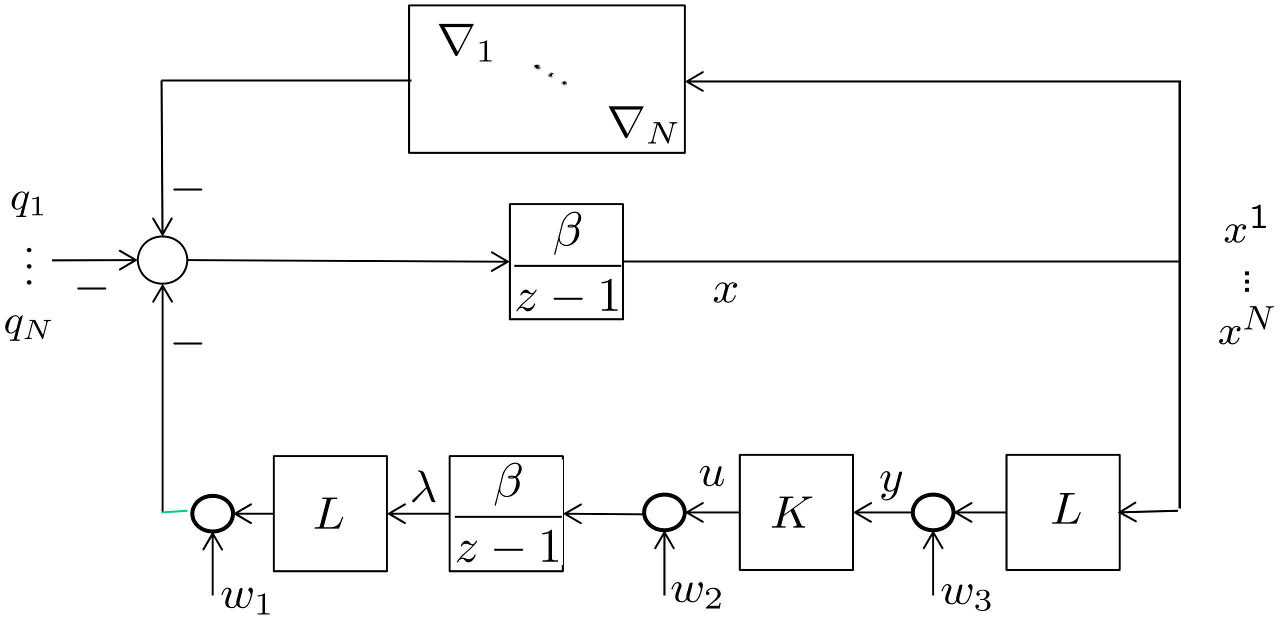
- Dynamical system converging to solution of least squares
- Sampling time β
- L=L' graph laplacian



$$\nabla_i = 2a_i a_i^T$$
$$q_i = -2y_i a_i$$

- β cannot be too large for stability

Advanced networked controller



Design a networked controller to improve stability margins and noise rejection

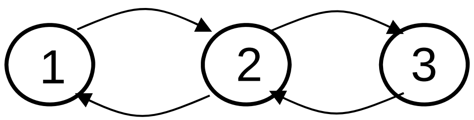
Special Case

$$\arg \min_{x_i = x_j, \forall i,j} \sum_{i=1}^{N} \frac{1}{2}(x_i - q_i)^2.$$

$$x_i \in \mathbb{R}, \text{ and } q_i \in \mathbb{R}.$$

Graph Laplacian

$$L = \begin{bmatrix} -1 & 1 & 0 \\ 1 & -2 & 1 \\ 0 & 1 & -1 \end{bmatrix}$$



Optimal solution, average: $x^* = \dfrac{q_1 + q_2 + q_3}{3}$

Networked Plant

Generalized Networked Plant

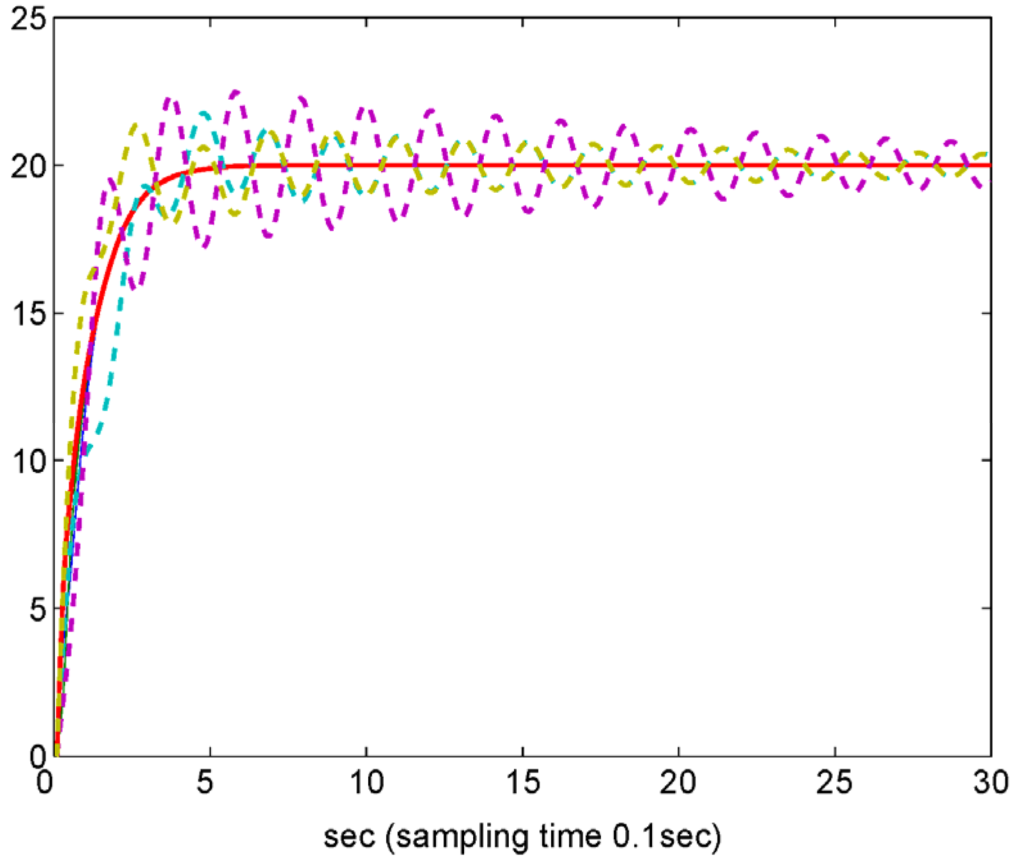
$$
\begin{aligned}
x(k+1) &= x(k) - \beta \nabla x(k) - \beta L \lambda(k) - \beta q + \beta w_1(k) \\
\lambda(k+1) &= \lambda(k) + \beta u(k) + w_2(k) \\
z_1(k) &= x(k) \\
z_2(k) &= \alpha u(k) \\
y(k) &= L x(k) + w_3(k)
\end{aligned}
$$

$$
P = \left[\begin{array}{cc|ccc|c}
I - \beta\nabla & -\beta L & -\beta\sigma_1^2 I & 0 & 0 & 0 \\
0 & I & 0 & -\beta\sigma_2^2 I & 0 & -\beta I \\
\hline
I & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & \alpha I \\
\hline
L & 0 & 0 & 0 & \sigma_3^2 I & 0
\end{array}\right],
$$

$$
P_{22} = \left[\begin{array}{cc|c}
I - \beta\nabla & -\beta L & 0 \\
0 & I & -\beta I \\
\hline
L & 0 & 0
\end{array}\right],
$$

Result β =0.1 sec

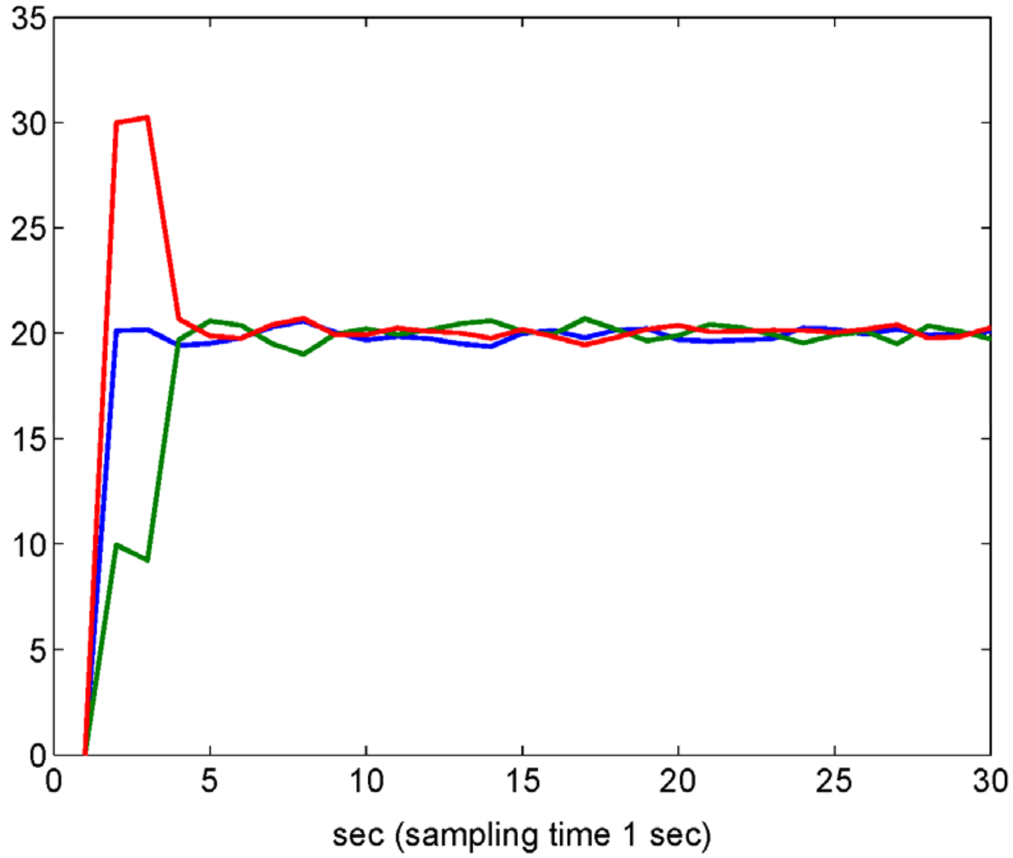
- Controller order [8,11,8]



- Lightly damped without controller

Result β=1 sec

- Controller order [6,8,6]



- Unstable without control for such sampling time.

Conclusions

- Networked controller design for Networked systems
- Defined a large class of networked systems
- Characterized networked system implementation and realization
- Derived All stabilizing (realizable) networked controllers
- Optimal networked controllers (realizable) on arbitrary networks
- Application to networked computing systems (in progress)