Input-and-state observability of structured network systems

Federica GARIN (INRIA Grenoble, France)

Lund LCCC seminar, June 7th, 2017

My current research interests

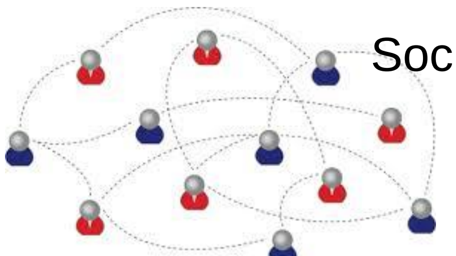
- **Privacy and security of cyber-physical systems:**
  - Input-and-state observability (this talk)
  - Counting nodes in anonymous networks
- **Urban traffic networks:**
  distributed optimization of traffic lights
- **Game theory (potential games):**
  distributed algorithms to find Nash Equilibrium

Outline

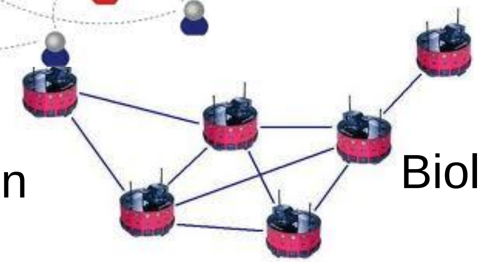
Part 1: Structural observability
(classical results)

Part 2: Structural input-and-state observability
(joint work with Alain Kibangou and Sebin Gracy)

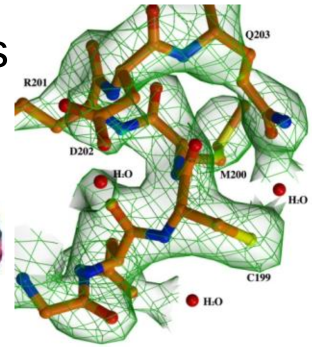
Network dynamical systems



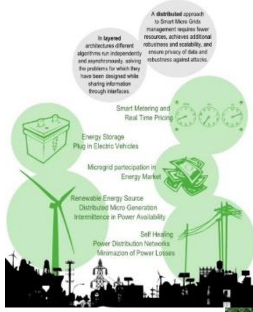
Social networks



Multi-robot coordination



Biological networks



Smart grids

Greenhouse irrigation

Intelligent transportation systems

Network dynamical systems – in this talk

Local states $x_i(k)$

Network state = vector collecting all local states

Local dynamics + interactions with some other states

➡ a (linear) system

$$\begin{cases} x(k+1) = Ax(k) + Bu(k) \\ y(k) = Cx(k) + Du(k) \end{cases}$$

Observability

By measuring only few local states (for some time), can we reconstruct the whole network state?

Classical algebraic conditions (1960-70's)

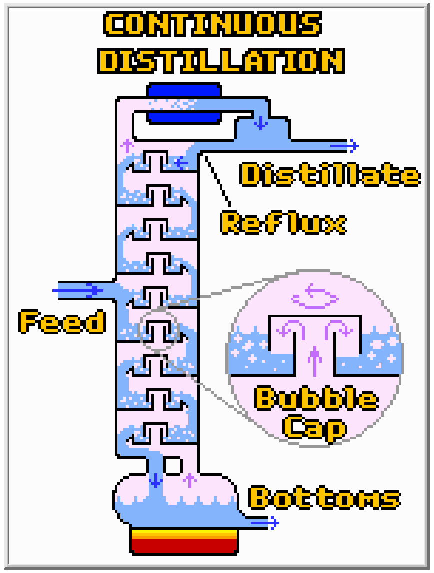
$$\begin{cases} x(k+1) = Ax(k) \\ y(k) = Cx(k) \end{cases}$$

is observable if and only if:

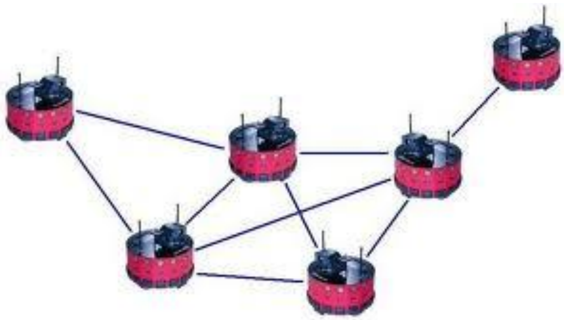
Kalman : $\begin{bmatrix} C \\ CA \\ \vdots \\ CA^{n-1} \end{bmatrix}$ has full column rank

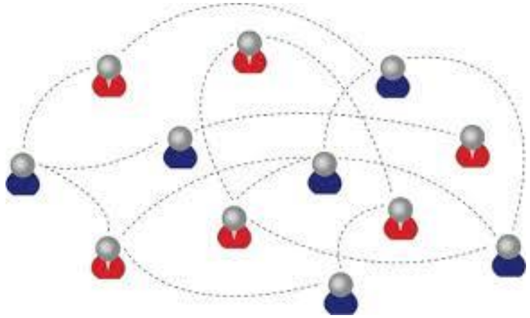
PBH: $\begin{bmatrix} zI - A \\ C \end{bmatrix}$ has full column rank $\forall z \in \mathbb{C}$

Graphical conditions (1980's + recent interest)



Not all states directly affect each other





Non-zero entries of system matrices $\leftrightarrow$ edges in network graph

Graphical conditions: structured systems (2)

- Seminal paper:

C.T. Lin, Structural controllability, IEEE Tr. Aut. Contr., 1974

- Works in the 70-80’s

See books by Murota (1987, 2000), Reinschke (1998), and survey paper by Dion, Commault, van der Woude (Automatica 2003)

- Recent revival in the context of **network systems**

A very popular paper (1400 citations):
Y. Y. Liu, J. J. Slotine and A. L. Barabasi,
Controllability of complex networks, Nature, 2011

Many recent works in the automatic control community
and in the complex networks community
(computer science, physics)

Structured systems – definition

$$\begin{cases} x(k+1) = Ax(k) \\ y(k) = Cx(k) \end{cases}$$

Non-zero entries of A, C are **free parameters**

$$A = \begin{bmatrix} 0 & 0 & 0 & \alpha_{14} & 0 & 0 \\ \alpha_{21} & 0 & 0 & 0 & \alpha_{25} & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & \alpha_{42} & 0 & 0 & 0 & 0 \\ 0 & 0 & \alpha_{53} & 0 & 0 & 0 \\ 0 & 0 & \alpha_{63} & 0 & 0 & 0 \end{bmatrix} \quad C = \begin{bmatrix} \gamma_{11} & 0 & 0 & 0 & 0 & 0 \\ 0 & \gamma_{22} & 0 & 0 & 0 & \gamma_{26} \\ 0 & 0 & 0 & 0 & \gamma_{35} & 0 \\ 0 & 0 & 0 & 0 & 0 & \gamma_{46} \end{bmatrix}$$

Generic results = true for almost all parameters

Almost all = except a proper subvariety of the param. space

If parameters are random, indep., continuous distribution:

Almost all = with prob. 1

Small detour: generic rank – examples

$$\begin{bmatrix} \mu_{11} & \mu_{12} \\ \mu_{21} & \mu_{22} \end{bmatrix}$$

has generic rank 2:
it is non-singular, except when

$$\mu_{11}\mu_{22} - \mu_{12}\mu_{21} = 0$$

$$\begin{bmatrix} \mu_{11} & 0 \\ \mu_{21} & \mu_{22} \end{bmatrix}$$

has generic rank 2;
moreover, it has rank 2
for all non-zero parameters

$$\begin{bmatrix} 0 & 0 \\ \mu_{21} & \mu_{22} \end{bmatrix}$$

has generic rank 1

Small detour: generic rank – characterization

Generic rank = size of maximum matching in bipartite graph

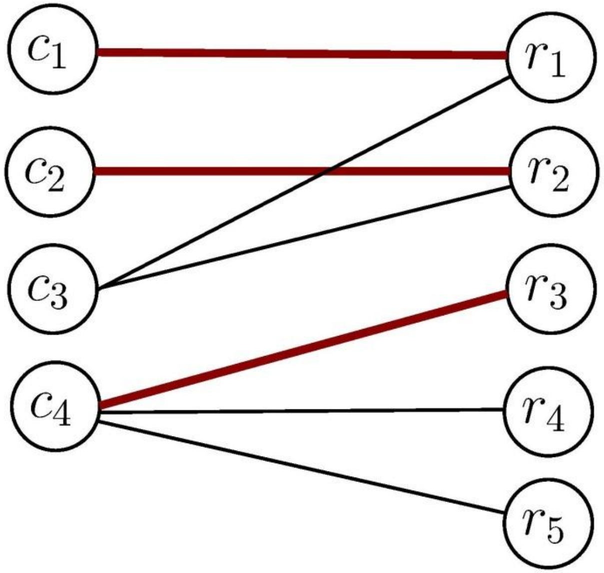
Bipartite graph

Left vertex set = columns

Right vertex set = rows

Edge $\{c_j, r_i\} \Leftrightarrow M_{ij} \neq 0$

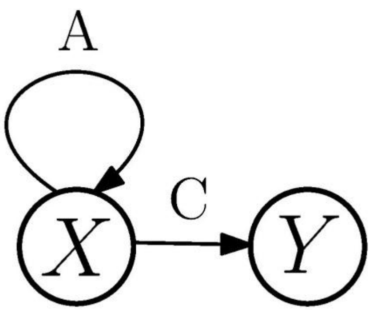
$$M = \begin{bmatrix} \mu_{11} & 0 & \mu_{13} & 0 \\ 0 & \mu_{22} & \mu_{23} & 0 \\ 0 & 0 & 0 & \mu_{34} \\ 0 & 0 & 0 & \mu_{44} \\ 0 & 0 & 0 & \mu_{54} \end{bmatrix}$$



generic rank = 3

Structured systems – digraph

Non-zero entries of A, C $\leftrightarrow$ edges in digraph

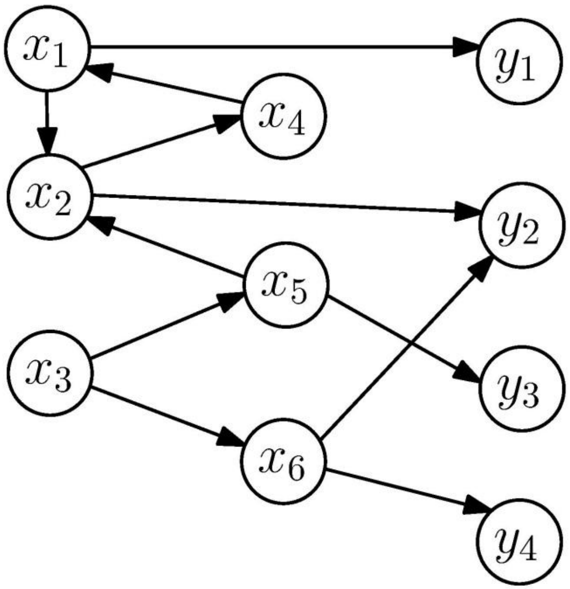


$$\boldsymbol{A}_{ij} \neq 0 \Leftrightarrow \text{edge } \boldsymbol{x}_j \rightarrow \boldsymbol{x}_i$$

$$\boldsymbol{C}_{ij} \neq 0 \Leftrightarrow \text{edge } \boldsymbol{x}_j \rightarrow \boldsymbol{y}_i$$

$$\boldsymbol{A} = \begin{bmatrix} 0 & 0 & 0 & \alpha_{14} & 0 & 0 \\ \alpha_{21} & 0 & 0 & 0 & \alpha_{25} & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & \alpha_{42} & 0 & 0 & 0 & 0 \\ 0 & 0 & \alpha_{53} & 0 & 0 & 0 \\ 0 & 0 & \alpha_{63} & 0 & 0 & 0 \end{bmatrix}$$

$$\boldsymbol{C} = \begin{bmatrix} \gamma_{11} & 0 & 0 & 0 & 0 & 0 \\ 0 & \gamma_{22} & 0 & 0 & 0 & \gamma_{26} \\ 0 & 0 & 0 & 0 & \gamma_{35} & 0 \\ 0 & 0 & 0 & 0 & 0 & \gamma_{46} \end{bmatrix}$$



Observability of structured systems (1)

Proposition

[R.W. Shields, J.B. Pearson, Structural controllability of multi-input linear systems, IEEE Tr. Aut. Contr., 1976]

If there exists one choice of free parameters for which (A, C) is observable, then (A, C) is generically observable.

I.e., for a given digraph,
either the system is observable for almost all parameters,
or it can't be observable, for any parameter choice.

Same for controllability, but not for all properties,
e.g., **not for stability**

Observability of structured systems (2)

Theorem

[C.T. Lin, Structural controllability, IEEE Tr. Aut. Contr., 1974 +
K. Murota, Systems analysis by graphs and matroids, 1987]

(A, C) is generically observable iff

i) Digraph is output-connected
(from every state vertex there is a path to an output vertex)

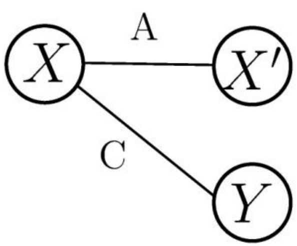
ii) Rank condition:

$$\begin{bmatrix} A \\ C \end{bmatrix}$$ generically has full column rank

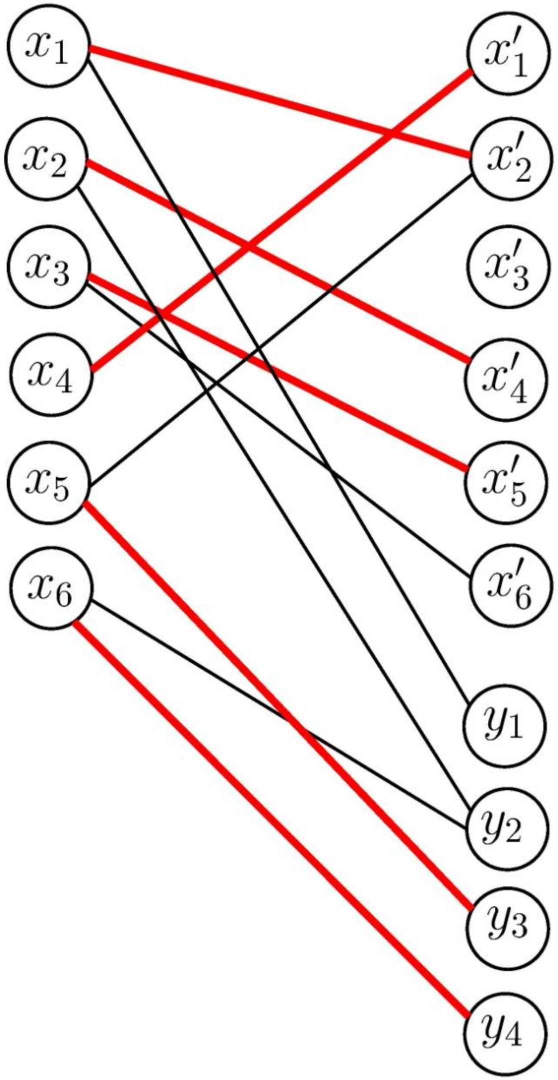
Equivalent versions of the rank condition (1)

$\begin{bmatrix} A \\ C \end{bmatrix}$ generically has full column rank iff

Bipartite graph



has a matching of size #X

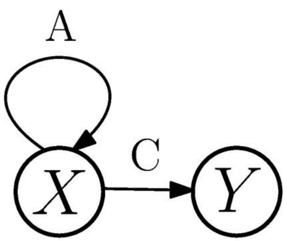


Remark

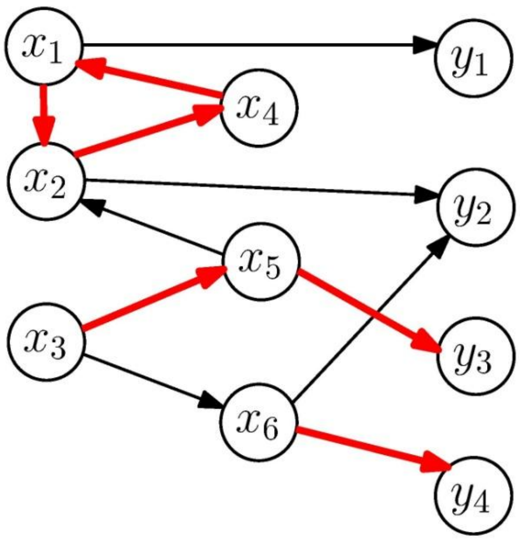
If A has non-zero diagonal, rank condition is always true!

Equivalent versions of the rank condition (2)

$\begin{bmatrix} A \\ C \end{bmatrix}$ generically has full column rank iff

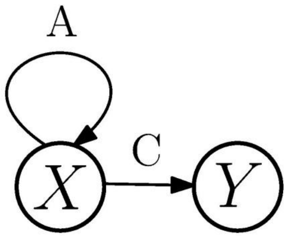
In digraph 

state vertices X are spanned by a collection of disjoint cycles and paths to output

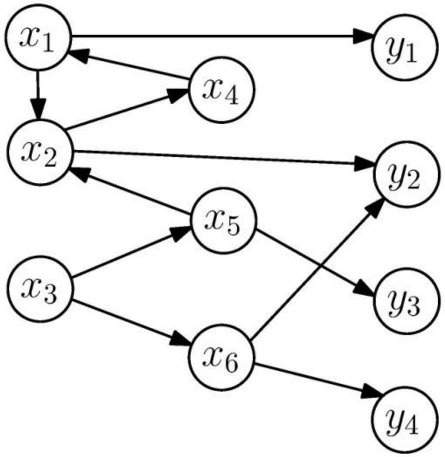


Equivalent versions of the rank condition (3)

$\begin{bmatrix} \boldsymbol{A} \\ \boldsymbol{C} \end{bmatrix}$ generically has full column rank iff

Digraph 

has no contraction:
for all set of state vertices S ⊆ X,
its set of out-neighbors E(S) has #E(S) ≥ #S



For example

$\boldsymbol{S} = \{\boldsymbol{x}_1, \boldsymbol{x}_2\}$

$\boldsymbol{E}(\boldsymbol{S}) = \{\boldsymbol{x}_2, \boldsymbol{x}_4, \boldsymbol{y}_1, \boldsymbol{y}_2\}$

Other classical results on observability...

- Structural observability = generically observable
  (for almost all parameters)
  Strong structural observability = for all non-zero parameters
  Characterizations of strong structural observability
  with uniquely restricted matchings, or zero-forcing sets

- LTV systems with constant graph:
  same characterization as corresponding LTI system

- LTV systems with varying graph:
  a characterization of structural observability
  with “dynamic graph”

Structural input-and-state observability

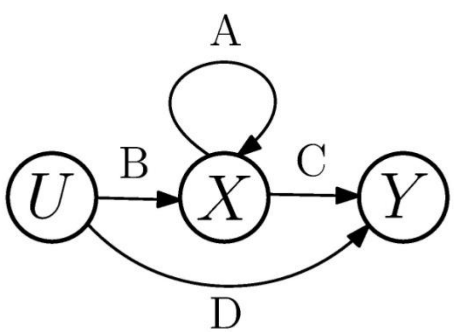
On-going work, with Sebin Gracy and Alain Kibangou

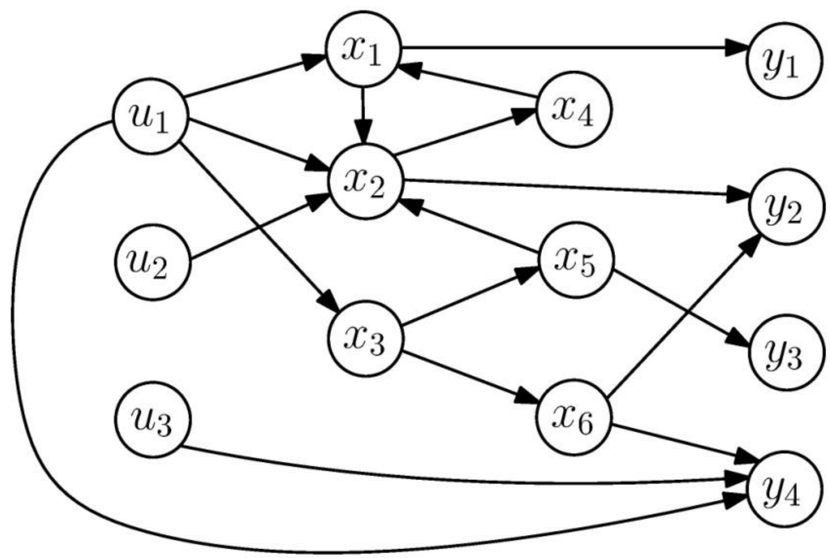
Motivation: cyber-physical security

What if an attacker injects an input in the system?

Other motivation: input can represent a fault

$$\begin{cases} \boldsymbol{x(k+1) = Ax(k) + Bu(k)} \\ \boldsymbol{y(k) = Cx(k) + Du(k)} \end{cases}$$





Input-and-state observability (ISO) – definition

$$\begin{cases} x(k+1) = Ax(k) + Bu(k) \\ y(k) = Cx(k) + Du(k) \end{cases}$$

- Strong observability: despite presence of unknown input u, can reconstruct initial state x(0) from outputs y(0), …, y(n)
- Delay-L left invertibility: can reconstruct input u(0) from x(0), y(0), …, y(L)
- Left invertibility (delay-L left inv. for some L ≤ n): can reconstruct input u(0) from x(0), y(0), …, y(n)
- **Input-and-state observability (ISO)** (strong obs + left inv): **can reconstruct x(0), u(0) from y(0), …, y(n)**
- **Delay-1 ISO** (ISO + delay-1 left inv.): **can reconstruct x(0), u(0), …, u(n-1) from y(0), …, y(n)**

ISO – algebraic characterization (classical)

- PBH-like test: ISO iff

$$\begin{bmatrix} A - zI & B \\ C & D \end{bmatrix}$$ has full column rank $\forall z \in \mathbb{C}$

- Delay-1 left inv. iff

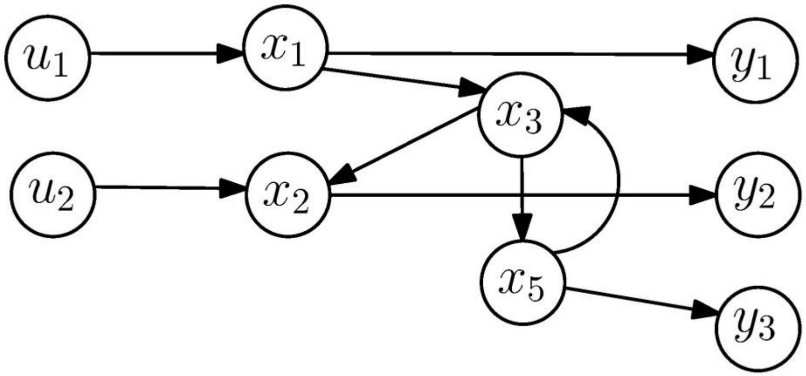
$$\operatorname{rank} \begin{bmatrix} D & 0 \\ CB & D \end{bmatrix} = \#U + \operatorname{rank} D$$

The two together give delay-1 ISO

Delay-1 ISO as observability of a subsystem

Assumption on matrices B, C, D:

- Each input acts on a single state
  (columns of B have a single non-zero element,
  input vertices have out-degree 1);
- Each output measures a single state
  (rows of C have a single non-zero element,
  output vertices have in-degree 1);
- D = 0 (no edge from U to Y).



Delay-1 ISO as observability of a subsystem

Under our assumption on B, C, D

Necessary condition for delay-1 ISO:

All attacked sates (i.e., affected by an input) **are measured**

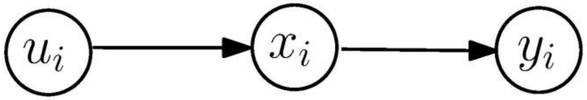
Proof: from characterization of delay-1 left inv. (in case D = 0)
CB full column rank

Delay-1 ISO as observability of a subsystem (2)

Under assumption on B, C, D + all attacked states are observed

System decomposition

Relabel vertices to put **attacked states first**:

for i = 1, …, #U, 

$$B = \begin{bmatrix} I \\ 0 \end{bmatrix} \quad C = \begin{bmatrix} I & 0 \\ 0 & \tilde{C} \end{bmatrix} \qquad x = \begin{bmatrix} x_a \\ \tilde{x} \end{bmatrix} \; y = \begin{bmatrix} y_a \\ \tilde{y} \end{bmatrix}$$

$$\begin{cases} x_a(k+1) = A_{aa} x_a(k) + A_{a\sim} \tilde{x}(k) + u(k) \\ y_a(k) = x_a(k) \end{cases}$$

$$\begin{cases} \tilde{x}(k+1) = \tilde{A} \tilde{x}(k) + A_{\sim a} x_a(k) \\ \tilde{y}(k) = \tilde{C} \tilde{x}(k) \end{cases}$$

Delay-1 ISO as observability of a subsystem (3)

Theorem

Under our assumption on B, C, D,

Delay-1 ISO iff

- All attacked states are measured
- Subsystem $(\tilde{A}, \tilde{C})$ is observable
  (subsystem without inputs, attacked states and corresponding outputs)

Proof: from PBH-like characterization

Same result also for LTV (constant B, C), more tricky proof

Delay-1 ISO as observability of a subsystem (4)

We can characterize generic delay-1 ISO using known characterization of structural observability

Corollary

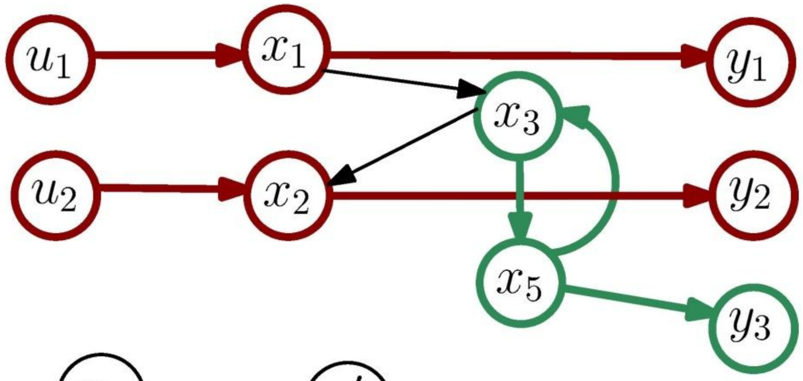
Under our assumption on B, C, D,

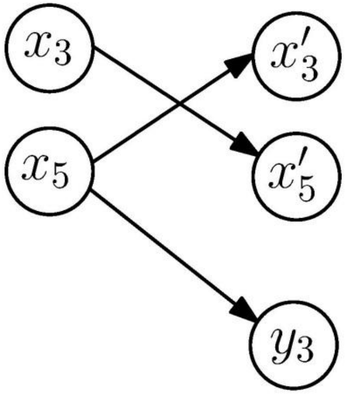
Generically delay-1 ISO iff

- All attacked states are measured,
- Subsystem $(\tilde{A}, \tilde{C})$

a) Bipartite graph has a matching of size #X-#U

b) Digraph is output-connected





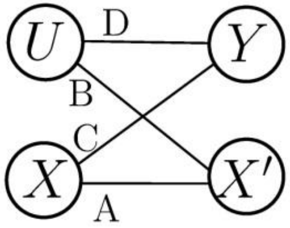
And more: strongly-structural (for all non-zero param), LTV

Structural ISO (no assumptions on B, C, D)

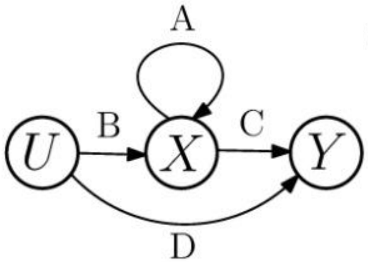
Proposition If there exists one choice of free parameters s.t. (A, B, C, D) is ISO, then (A, B, C, D) is generically ISO.

Theorem [Based on Boukhobza et al, State and input observability for structured linear systems: A graph-theoretic approach, Automatica, 2007]

Generically ISO iff

a) Bipartite graph 

has a matching of size #U+#X

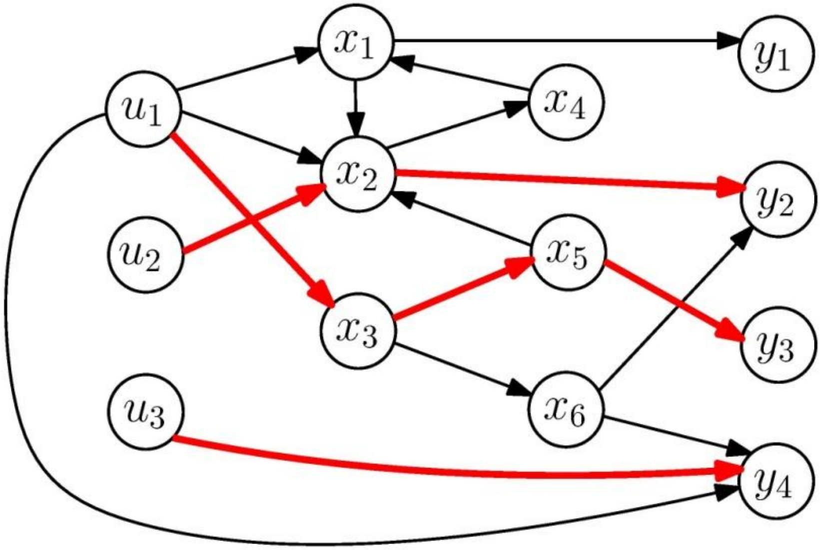
b) In 

from every non-essential state vertex there is a path to an output vertex, with no essential vertex in the path

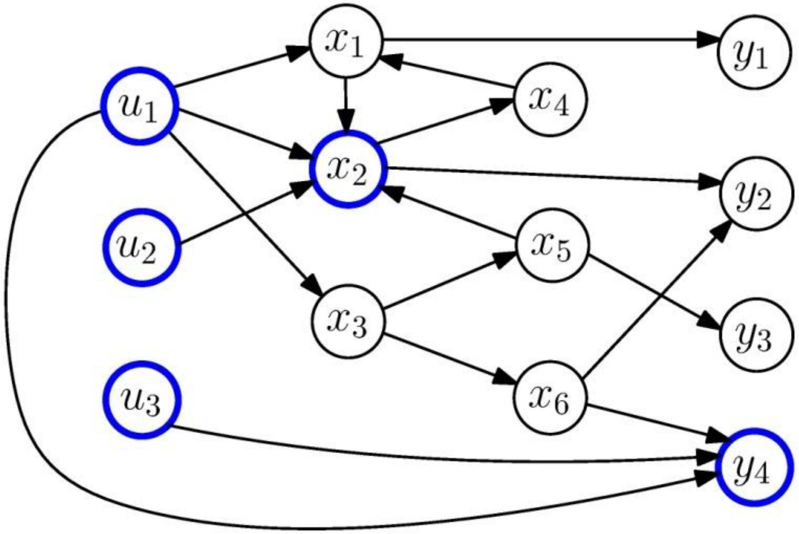
Essential vertices

Linking from U to Y =
set of vertex-disjoint paths
from U to Y

Size of a linking = # paths



Essential vertices
= vertices present in all
maximum linkings
= union of all minimum
vertex separators

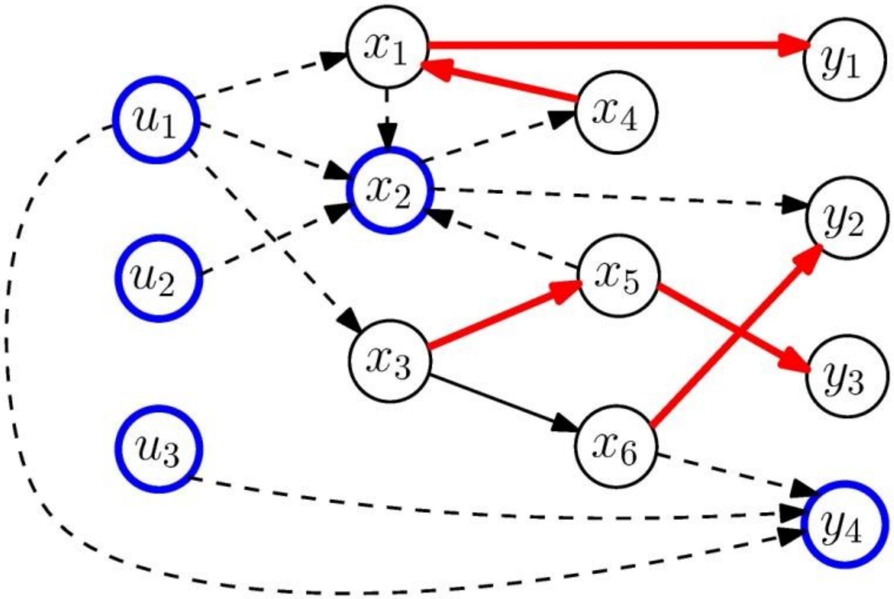


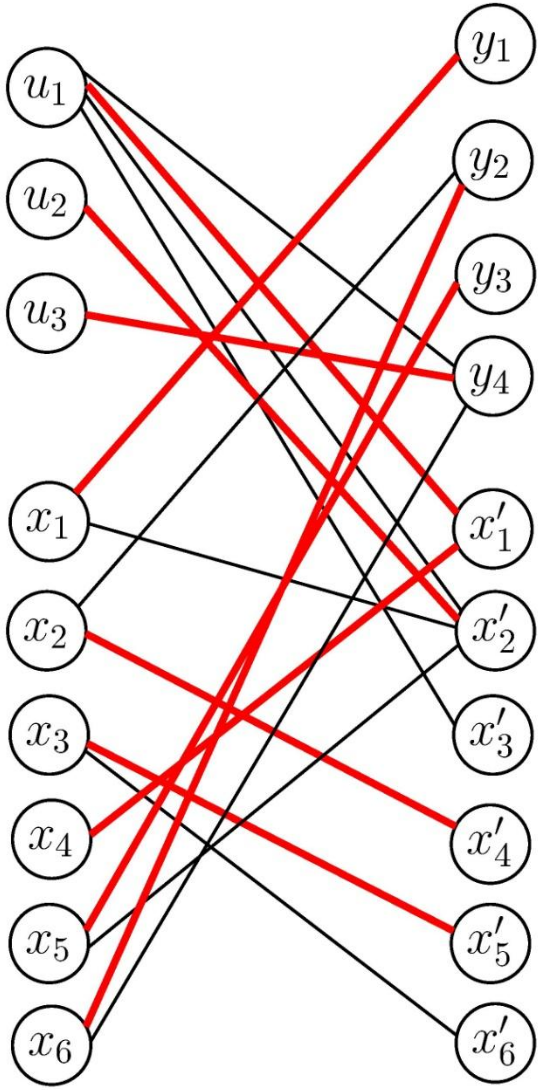
Remark: under a), size of max-linking = # U

Structural ISO – Example

a) Bipartite graph
has a matching of size #U+#X

b) In digraph, from every non-essential state vertex there is a path to an output vertex, with no essential vertex in the path





Structural delay-1 left invertibility

Proposition

If D= 0, if there exists one choice of free parameters for which (A, B, C, D) is delay-1 left inv,
then (A, B, C, D) is generically ISO delay-1 left inv.

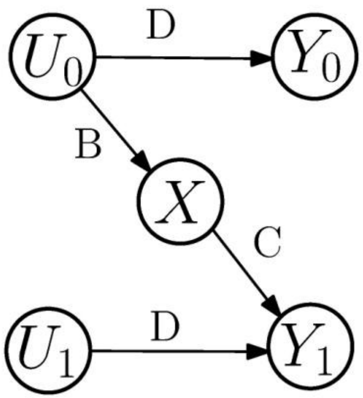
I.e., when D=0, for a given digraph,
either the system is delay-1 left inv for almost all parameters,
or it can't be delay-1 left inv, for any parameter choice.

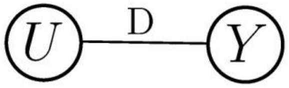
For general D, for a given digraph,
either the system is delay-1 left inv for almost all parameters,
or it is not delay-1 left inv for almost all parameters
(but there might be few parameters for which it is)

Structural delay-1 left invertibility (2)

Theorem

Generically delay-1 left inv. iff
Exists linking of size #U + r from $\boldsymbol{U}_0 \cup \boldsymbol{U}_1$ to $\boldsymbol{Y}_0 \cup \boldsymbol{Y}_1$
in:

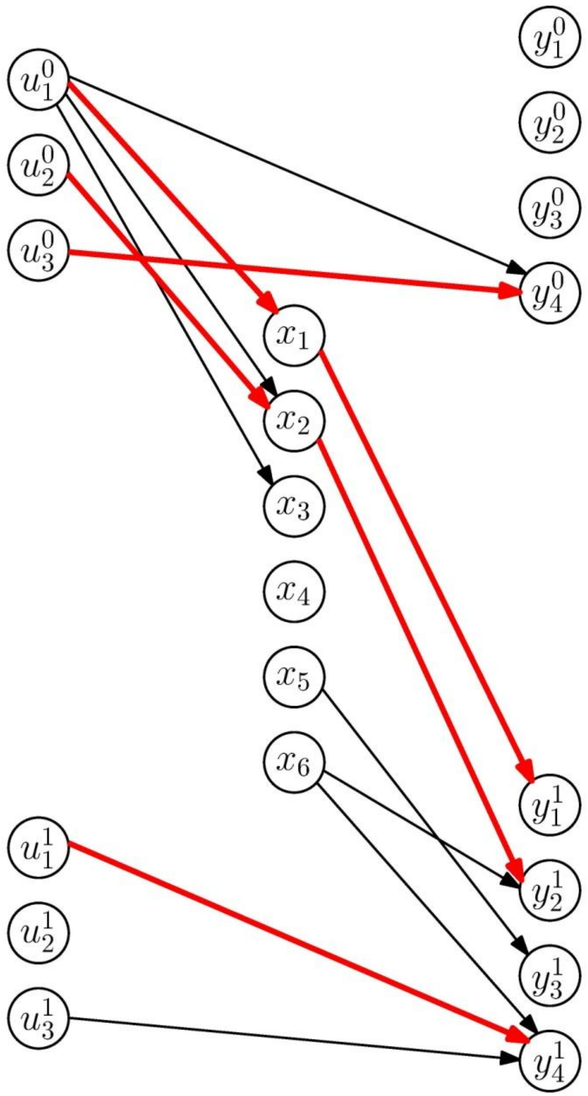


r = generic rank (D)
= size of max matching in 

Structural delay-1 left invertibility – Example

r = generic rank (D) = 1

$$D = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \\ \delta_{41} & 0 & \delta_{43} \end{bmatrix}$$



Generically delay-1 left inv:
Exists linking
from $U_0 \cup U_1$ to $Y_0 \cup Y_1$
of size #U + r = 4

Structural delay-1 left invertibility – Example 2

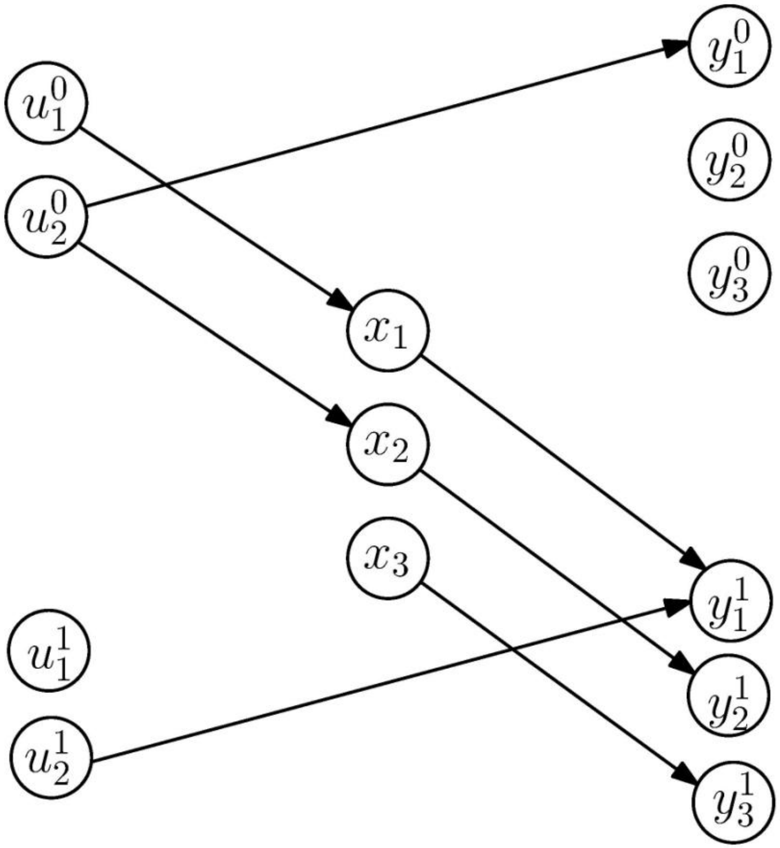
$$B = \begin{bmatrix} \beta_1 & 0 \\ 0 & \beta_2 \\ 0 & 0 \end{bmatrix} \quad C = \begin{bmatrix} \gamma_1 & 0 & 0 \\ 0 & \gamma_2 & 0 \\ 0 & 0 & \gamma_3 \end{bmatrix} \quad D = \begin{bmatrix} 0 & \delta \\ 0 & 0 \\ 0 & 0 \end{bmatrix}$$

r = generic rank (D) = 1

Not generically delay-1 left inv:
size of max linking = 2 < #U + r

But if $\delta = 0$ and
$\beta_1, \beta_2, \gamma_1, \gamma_2 \neq 0$
it is delay-1 left inv.

$$CB = \begin{bmatrix} \beta_1\gamma_1 & 0 \\ 0 & \beta_2\gamma_2 \\ 0 & 0 \end{bmatrix}$$



Conclusion

This talk

- Structural systems: generic results, depending only on zero pattern, true for almost all paramenters
- Classical characterization of structural observability
- Recent results on structural ISO (with delay 1)

Current work on structural ISO

- LTV
- Strong structural (for all non-zero parameters)
- Delay-L left inv.

Future work

- Other notions related to attack detection
- Distributed algorithms for ISO or other defense from attacks