Identification of Networks of Dynamic Systems

Michel Verhaegen

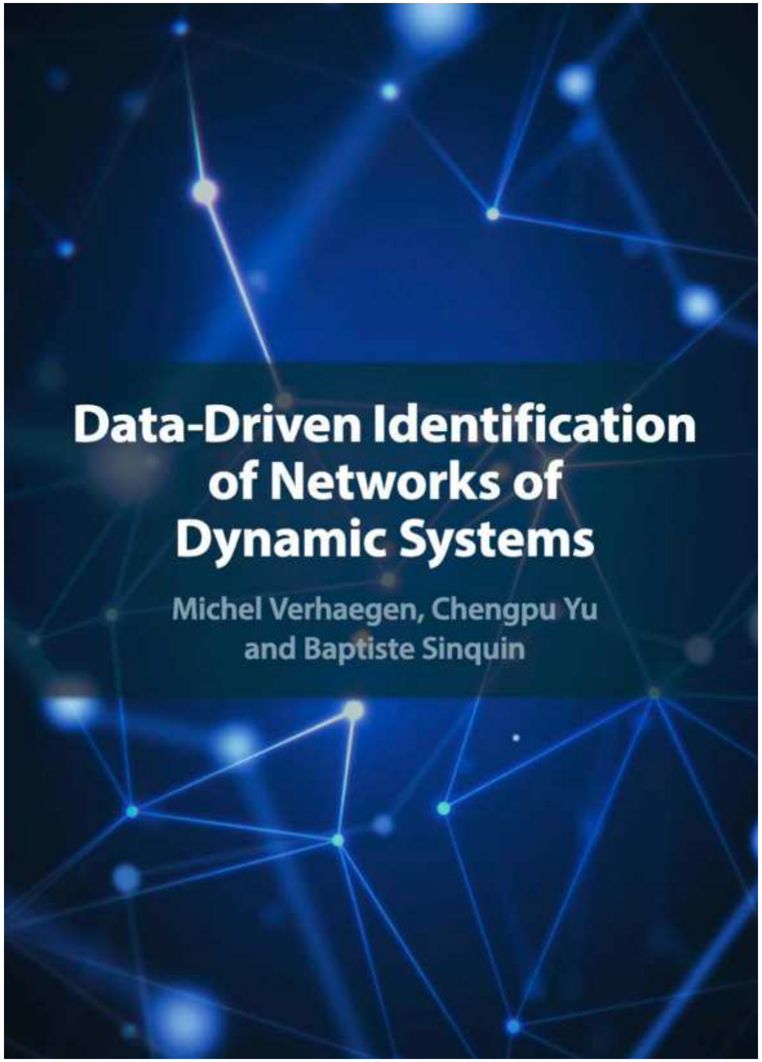
Delft Center for Systems and Control

Delft University of Technology

Content

1. **Three parts based on recent book**

Based on Cambridge University Press 2022



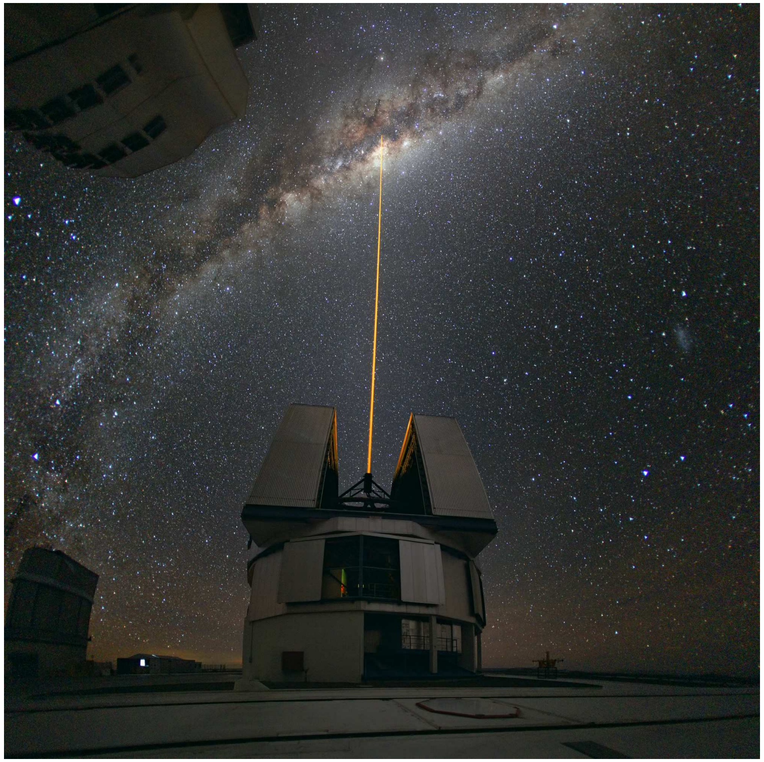
M.Verhaegen, Chengpu Yu and Baptiste Sinquin: “Data-Driven Identification of Networks of Dynamic Systems”

Content

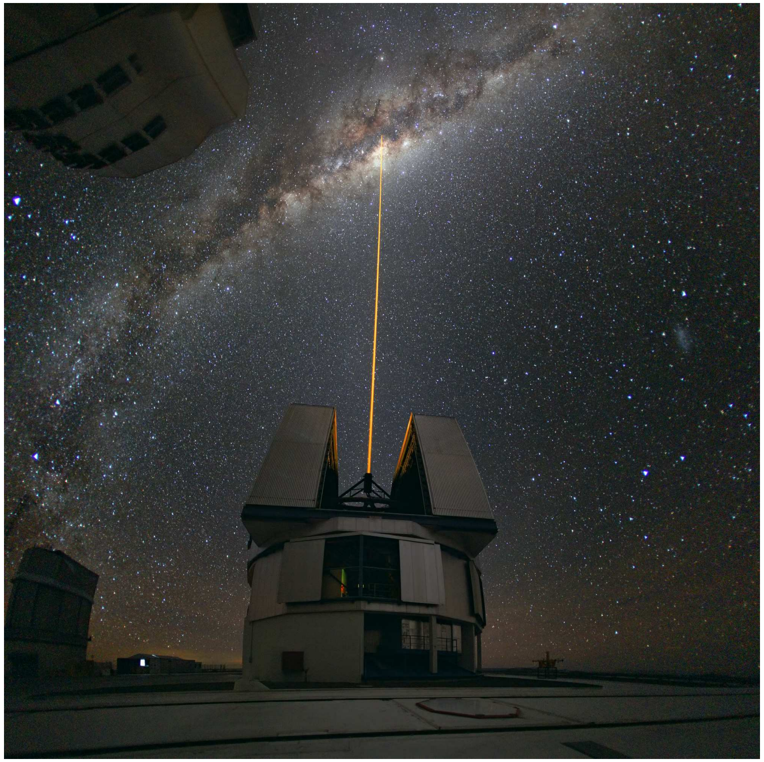
1. **Part 1:** Parametrization Large-Scale Dynamic Networks
2. **Part 2:** Identification methods
3. **Part 3:** Towards Control of Large-Scale Adaptive Optics Systems

Examples

An example of Large Scale Dynamic Systems



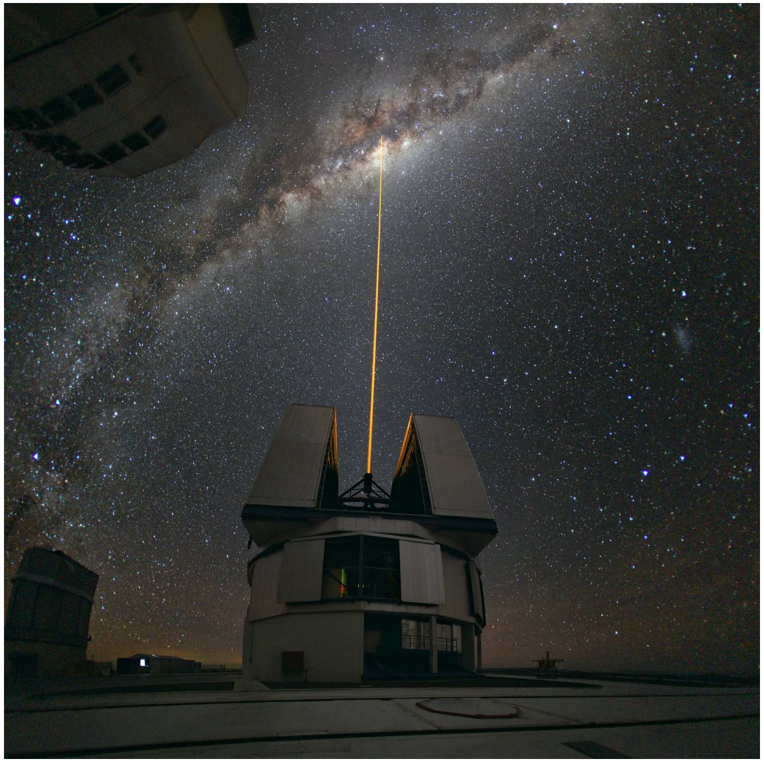
An example of Large Scale Dynamic Systems



ELT (XL-Telescope) :

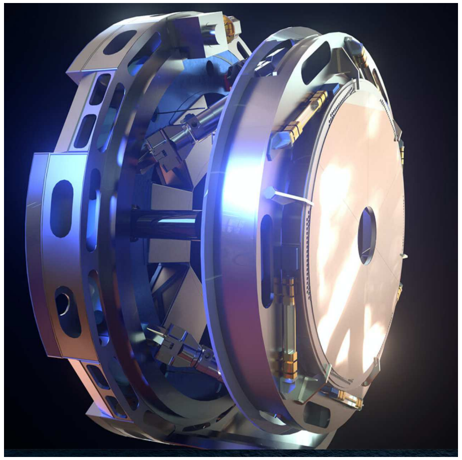
1. Primary Mirror: Segmented — 798 hexagonal segments (39m ⌀)

An example of Large Scale Dynamic Systems



ELT (XL-Telescope) :

1. Primary Mirror: Segmented — 798 hexagonal segments (39m ⌀)
2. M4 Adaptive Mirror: 8000 actuators (2.4 m ⌀).



3. $\cdots$

Many More Examples of Large Scale Networks of Dynamic Sy

1. Active Boundary Layer Control
2. Formation Flying of satellites
3. Cellular Network dynamics in diseases
4. $\cdots$

Part 1: Parametrization Large-Scale Dynamic Networks

Content on Parametrization

1. Transfer Function models
2. Structured State Space models
3. etc.

Part 1a: Transfer Function Models

Content for Parametrizing Transfer Funct

1. Use of Graphs to define sparse (transfer function) matrices.
2. **Kronecker-Based VAR (Quarks) models**
3. Tensor VAR models
4. etc.

Need for Structural Model Parametrization

Given a MIMO system modeled as:

$$y(k) = G(q)u(k) + H(q)e(k)$$

with $y(k) \in \mathbb{R}^p, u(k) \in \mathbb{R}^m$ resp. the output and input signals of the system, and $e(k) \in \mathbb{R}^p$ a (temporally) white noise signal and

- $G(z)$ is a $p \times m$ rational function matrix (strictly) proper
- $H(z)$ is a $p \times p$ rational function matrix (proper)

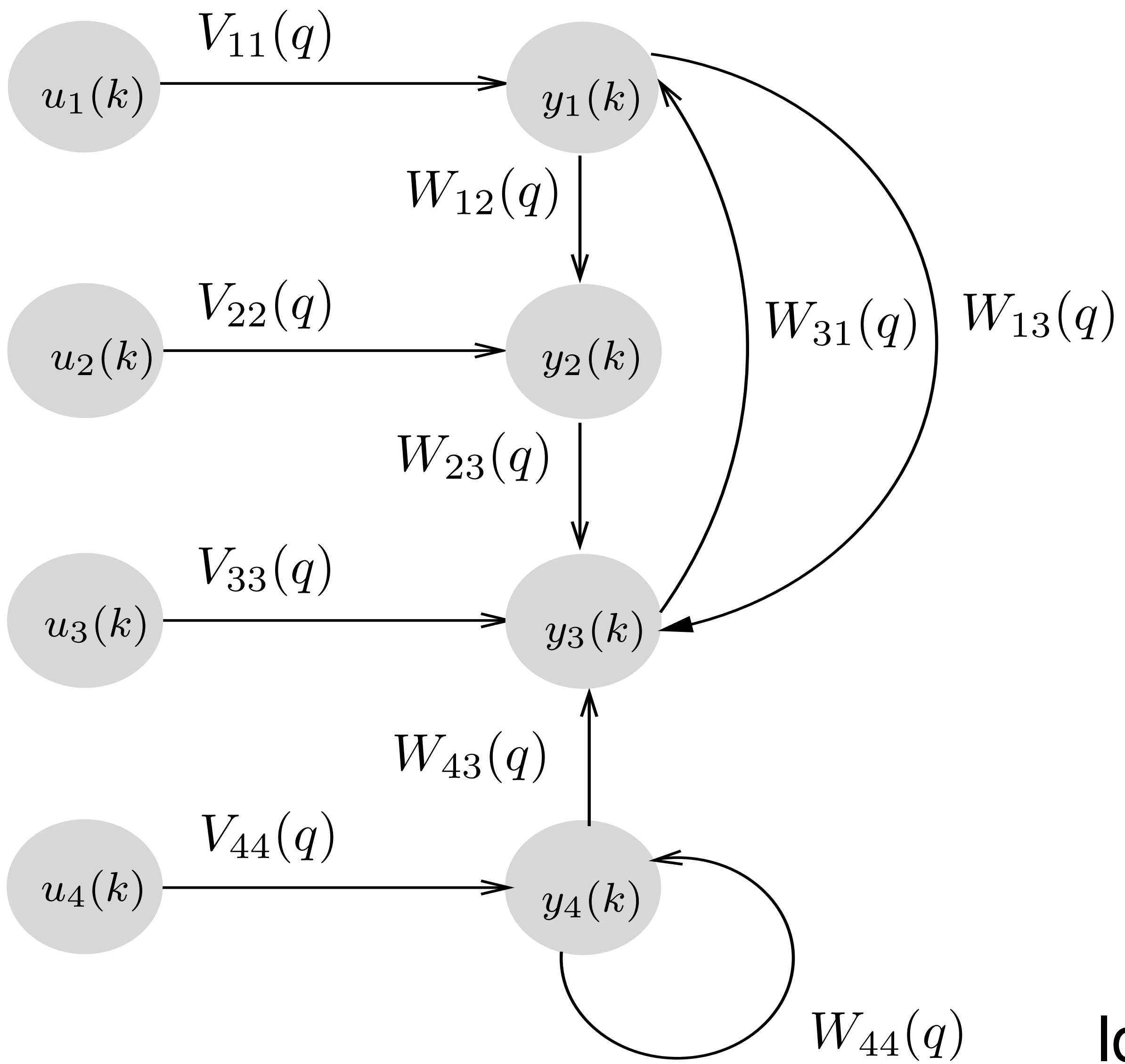
What if p, m is of $O(10^4)$?

Example of a sparse DNF

$$\begin{bmatrix} y_1(k) \\ y_2(k) \\ y_3(k) \\ y_4(k) \end{bmatrix} = \begin{bmatrix} 0 & W_{12}(q) & W_{13}(q) & 0 \\ 0 & 0 & W_{23}(q) & 0 \\ W_{31}(q) & 0 & 0 & 0 \\ 0 & 0 & W_{43}(q) & W_{44}(q) \end{bmatrix} \begin{bmatrix} y_1(k) \\ y_2(k) \\ y_3(k) \\ y_4(k) \end{bmatrix}$$

$$+ \begin{bmatrix} V_{11}(q) & 0 & 0 & 0 \\ 0 & V_{22}(q) & 0 & 0 \\ 0 & 0 & V_{33}(q) & 0 \\ 0 & 0 & 0 & V_{44}(q) \end{bmatrix} \begin{bmatrix} u_1(k) \\ u_2(k) \\ u_3(k) \\ u_4(k) \end{bmatrix}$$

Example of a sparse DNF (Graph representation)



Identifiability question?

Sparse VAR models

Consider the 2nd-order VAR model:

$$y(k) = A_1 y(k-1 + A_2 y(k-2) + e(k) \quad e(k) \frown (0, \Sigma_e) \;\; K_e = \Sigma_e^{-1} \;\; y(k) \in \mathbb{R}^p$$

Then the Granger Causality graph is the mixed graph (V_{GC}, A_{GC}, E_{GC}), with A_{GC} representing the directed edges and E_{GC} the undirected ones, is defined as follows:

$$\begin{aligned} V_{GC} \quad &= \quad \{1, \cdots, p\} \\ (i,j) \notin A_{GC} \quad &\Leftrightarrow \quad (A_\tau)_{ji} = 0 \;\; \forall \tau \in \{1, \cdots, 2\} \\ (i,j) \notin E_{GC} \quad &\Leftrightarrow \quad K_{ij} = 0 \end{aligned}$$

Generalized VAR(X) models

The generalized VAR model:

$$A_0^G y(k) = A_1^G y(k-1) + A_2^G y(k-2) + e_0(k) \quad e(k) \frown (0, \Sigma_0) \;\; \Sigma_0 \text{ diagonal}$$

with A_0^G lower triangular with unit entries on the diagonal.
The relation with the original VAR(X) model is that:

$$\Sigma_e = (A_0^G)^{-1} \Sigma_0 (A_0^G)^{-T}$$

maybe full, while its inverse is sparse. This introduces sparsity in A_0^G and probably also in the other parameter matrices.

The class of Sum-of-Kronecker Product Matrices

Definition: The class of sum-of-Kronecker product matrices is defined as:

$$\mathcal{K}_{2,r} = \{M \in \mathbb{R}^{m_1 m_2 \times n_1 n_2} \ |M = \sum_{i=1}^{r} M_{i,2} \otimes M_{i,1}\}$$

for $\text{for} \ M_{i,1} \in \mathbb{R}^{m_1 \times n_1}, M_{i,2} \in \mathbb{R}^{m_2 \times n_2}$. The matrices $M_{i,1}$ and $M_{i,2}$ will be called the factor matrices.

The class of Sum-of-Kronecker Product Matrices

Definition: The class of sum-of-Kronecker product matrices is defined as:

$$\mathcal{K}_{2,r} = \{M \in \mathbb{R}^{m_1 m_2 \times n_1 n_2} \ |M = \sum_{i=1}^{r} M_{i,2} \otimes M_{i,1}\}$$

for $\text{for} \;\; M_{i,1} \in \mathbb{R}^{m_1 \times n_1}, M_{i,2} \in \mathbb{R}^{m_2 \times n_2}$. The matrices $M_{i,1}$ and $M_{i,2}$ will be called the factor matrices.

Definition: [Kronecker rank] A matrix within $\mathcal{K}_{2,r}$ has Kronecker rank r when the following two matrices with r columns, which each column a vectorization of the matrices $M_{i,1}$ and $M_{i,2}$ respectively denoted as,

$$\begin{bmatrix} \cdots & \text{vec}\Big(M_{i,1}\Big) & \cdots \end{bmatrix} \quad \begin{bmatrix} \cdots & \text{vec}\Big(M_{i,2}\Big) & \cdots \end{bmatrix}$$

have full column rank r.

Advantages of the class $\mathcal{K}_{2,r}$

Let $x \in \mathbb{R}^{N^2}$. Then, the orders of magnitude of the computational complexity for matrix-vector multiplication, matrix-matrix multiplication and matrix inversion are as follows:

	$A, B \in \mathbb{R}^{N^2 \times N^2}$	$A, B \in \mathcal{K}_{2,r}$
Ax	$\mathcal{O}(N^4)$	$\mathcal{O}(rN^3)$
AB	$\mathcal{O}(N^6)$	$\mathcal{O}(r^2N^3)$
A^{-1} (case: Kronecker rank of A is 1)	$\mathcal{O}(N^6)$	$\mathcal{O}(N^3)$

The complexity obtained with the Kronecker-product parametrization considers the operations required for forming the factor matrices only.

The Quarks (Kronecker-based ARX) Model

The Quarks model is defined as:

$$y(k) = \sum_{i=1}^{n} \Big(\sum_{j=1}^{r_i} M_{i,j,2} \otimes M_{i,j,1} \Big) y(k-i)$$

Assume that $y(k) = \mathrm{vec}(Y(k))$ for $Y(k) \in \mathbb{R}^{pN \times N}$ and using the property of the vec-operator that vec$(XYZ) = (Z^T \otimes X)\mathrm{vec}(Y)$, we can write the Quarks model as:

$$Y(k) = \sum_{i=1}^{n} \sum_{j=1}^{r_i} \Big(M_{i,j,1} Y(k-i) M_{i,j,2}^T \Big)$$

The Quarks (Kronecker-based ARX) Model

The Quarks model is defined as:

$$y(k) = \sum_{i=1}^{n} \Big(\sum_{j=1}^{r_i} M_{i,j,2} \otimes M_{i,j,1} \Big) y(k-i)$$

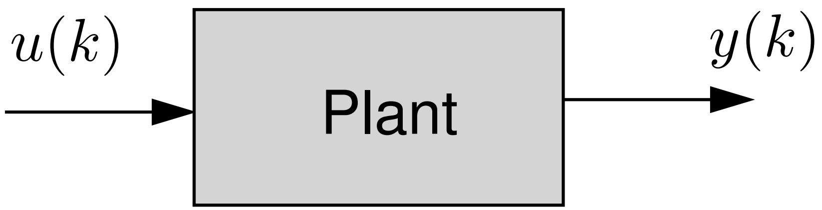
Assume that $y(k) = \mathrm{vec}(Y(k))$ for $Y(k) \in \mathbb{R}^{pN \times N}$ and using the property of the vec-operator that vec$(XYZ) = (Z^T \otimes X)\mathrm{vec}(Y)$, we can write the Quarks model as:

$$Y(k) = \sum_{i=1}^{n} \sum_{j=1}^{r_i} \Big(M_{i,j,1} Y(k-i) M_{i,j,2}^T \Big)$$

Remark: This can be generalized further by parametrizing the coefficient matrices with Tensor calculus. See [Data-Driven Identification of Networks of Dynamic Systems, Ch. 3].

Part 1b: Parametrization State Space models

State Space Models

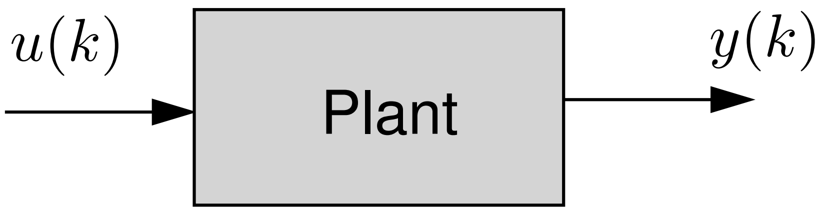


An LTI state space model is given as,:

$$
\begin{aligned}
x(k+1) &= Ax(k) + Bu(k) \\
y(k) &= Cx(k) + v(k)
\end{aligned}
$$

for $x(k) \in \mathbb{R}^n$, $y(k) \in \mathbb{R}^p, u(k) \in \mathbb{R}^m$ with n, m, p of $O(10^4)$, **full parametrization**?

State Space Models



An LTI state space model is given as,:

$$
\begin{aligned}
x(k+1) &= Ax(k) + Bu(k) \\
y(k) &= Cx(k) + v(k)
\end{aligned}
$$

for $x(k) \in \mathbb{R}^n$, $y(k) \in \mathbb{R}^p, u(k) \in \mathbb{R}^m$ with n, m, p of $O(10^4)$, **full parametrization**?

Or spatial structure imposed by "special" structure on the system matrices A, B, C?

Parametrization of SSM

1. Sparsely State space models.
   - A priori parametrized
   - Decomposable systems
   - Systems with Block-Tridiagonal system matrices
2. Data-sparse parametrized State Space models.
   - 1D: Sequentially Semi-Seperable system matrices
   - Multi-dimensional State Space Models

Block-Triangular state space model

This is state space model of the form:

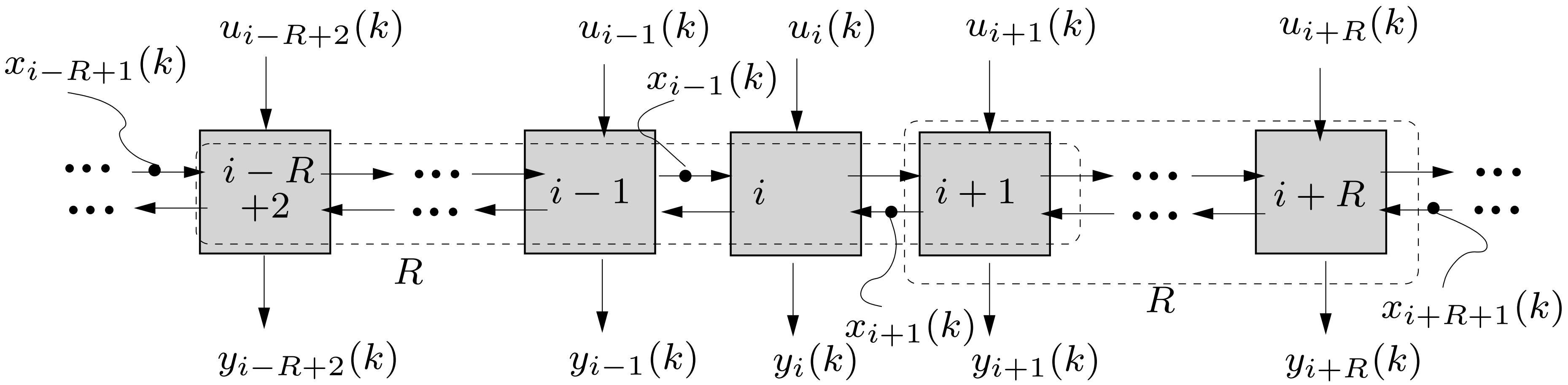
$$\begin{cases} x(k+1) & = \mathcal{A}x(k) + \mathcal{B}u(k) + \mathcal{H}w(k) \\ y(k) & = \mathcal{C}x(k) + \mathcal{D}u(k) + \mathcal{G}w(k) \end{cases}$$

With the system matrices Block-Triangular. Such as,

$$\mathcal{A} = \begin{bmatrix} A_1 & A_{1,r} & & \\ A_{2,\ell} & A_2 & \ddots & \\ & \ddots & \ddots & A_{N-1,r} \\ & & A_{N,\ell} & A_N \end{bmatrix}$$

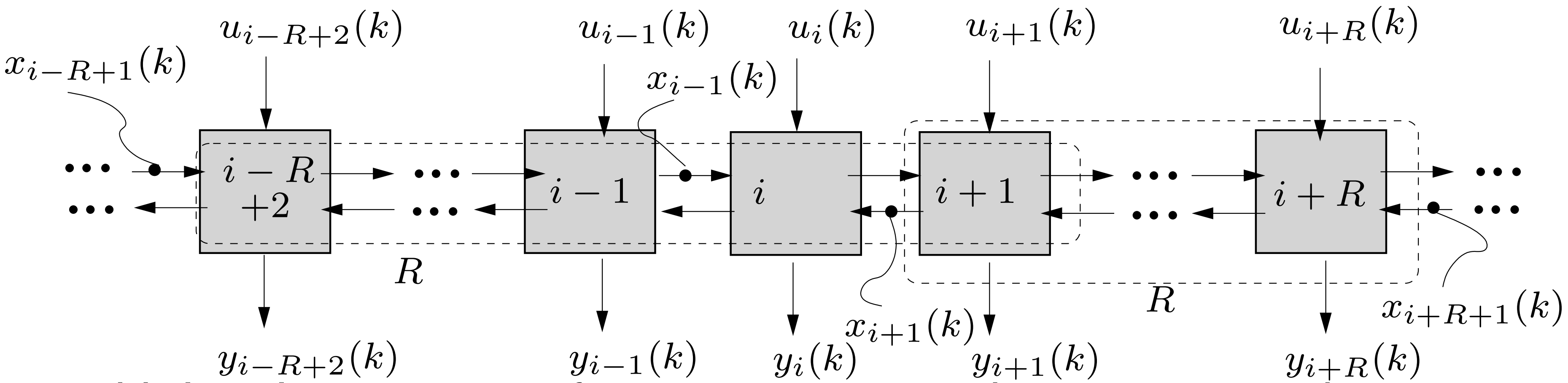
Example: 1D Heterogeneous DNS

Consider the network of LTI systems:



Example: 1D Heterogeneous DNS

Consider the network of LTI systems:



with local systems Σ_i for $i = 1, \cdots, N$ to be represented as,

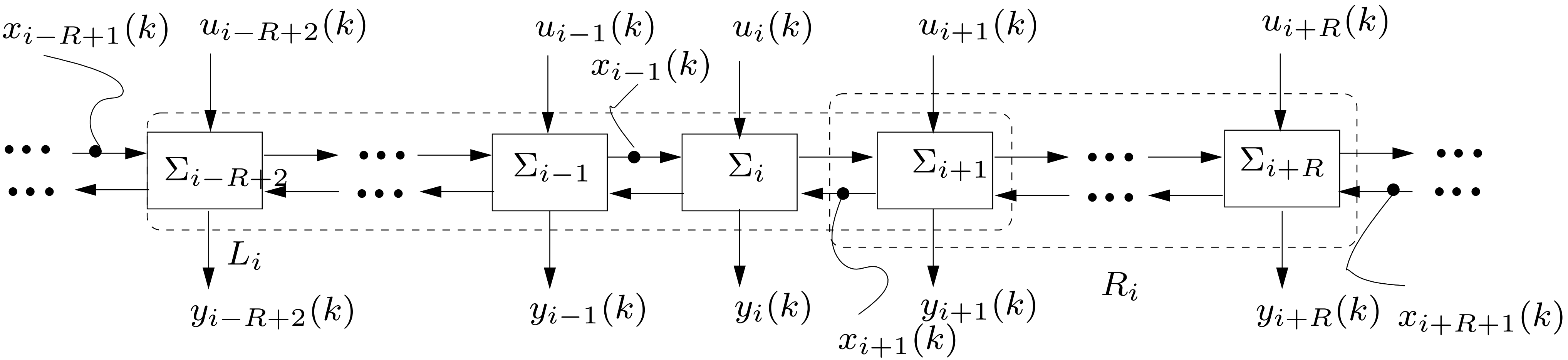
$$\begin{aligned}
\Sigma_1 \quad : \quad x_1(k+1) &= A_1 x_1(k) + A_{1,r} x_2(k) + B_1 u_1(k) \\
y_1(k) &= C_1 x_1(k) + e_1(k) \\
\Sigma_i \quad : \quad x_i(k+1) &= A_i x_i(k) + A_{i,\ell} x_{i-1}(k) + A_{1,r} x_{i+1}(k) + B_i u_i(k) \\
y_i(k) &= C_i x_i(k) + e_i(k) \\
\Sigma_N \quad : \quad x_N(k+1) &= A_N x_1(k) + A_{N,\ell} x_{N-1}(k) + B_N u_N(k) \\
y_N(k) &= C_N x_N(k) + e_N(k)
\end{aligned}$$

where $x_i(k) \in \mathbb{R}^{n_i}, u_i(k) \in \mathbb{R}^{m_i}, y_i(k) \in \mathbb{R}^{p_i}$ and $e_i(k)$ is a zero-mean white noise sequence with given covariance matrix.

Part 2: Identification of Large-Scale Dynamic Networks

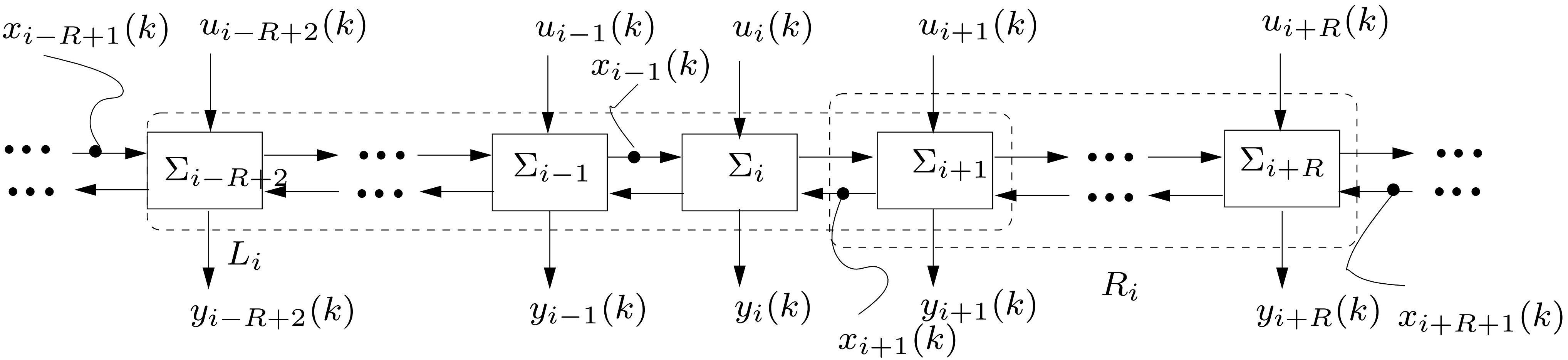
One Identification Problem

Consider a 1D-network of N dynamic systems and we zoom in on the local system Σ_i and its LOCAL neigborhood!



One Identification Problem

Consider a 1D-network of N dynamic systems and we zoom in on the local system Σ_i and its LOCAL neigborhood!



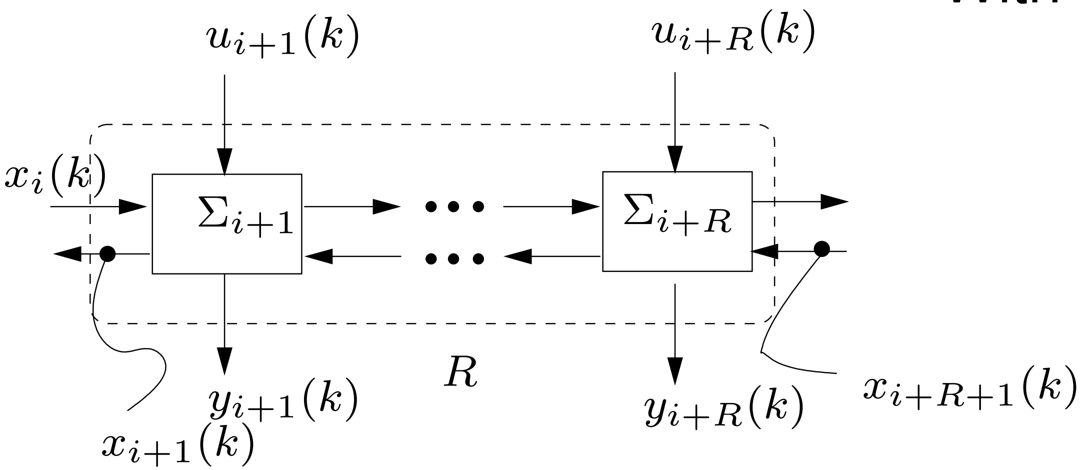
Identification problem of Identifying Local System Σ_i

Given: local input-output data of systems Σ_j for $j = i - R + 2, \cdots, i + R$ with $R \ll N$

Determine: the system matrices A_i, B_i, C_i of system Σ_i

Strategy to solve the "one" Identification Problem

- Identify the state sequence $x^R(k)$ of the lifted system of R-systems to the right of Σ_i.

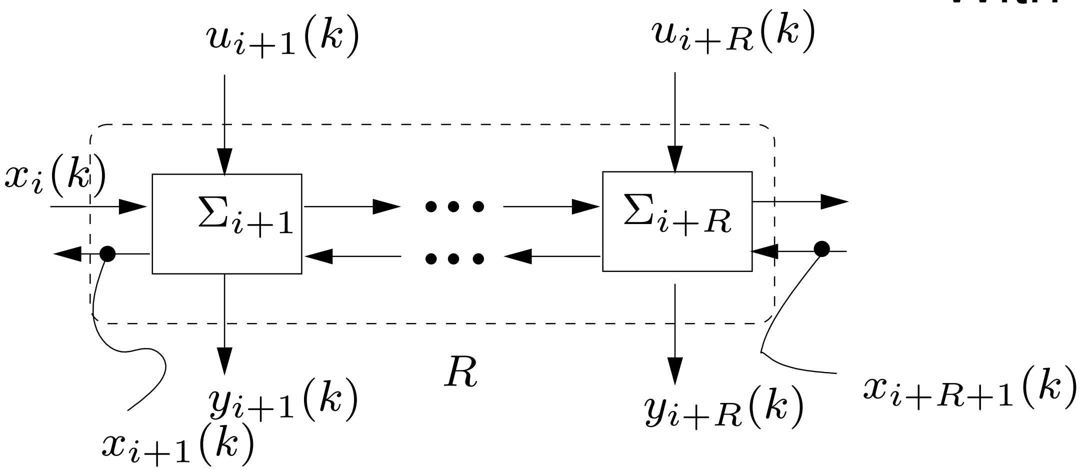


With that lifted system denoted as:

$$
\begin{aligned}
x^R(k+1) &= A^R x^R(k) + B^R u^R(k) \\
&\quad + F^R \begin{bmatrix} x_i(k) \\ x_{i+R+1}(k) \end{bmatrix} \\
y^R(k) &= C^R x^R(k)
\end{aligned}
$$

Strategy to solve the "one" Identification Problem

- Identify the state sequence $x^R(k)$ of the lifted system of R-systems to the right of Σ_i.



With that lifted system denoted as:

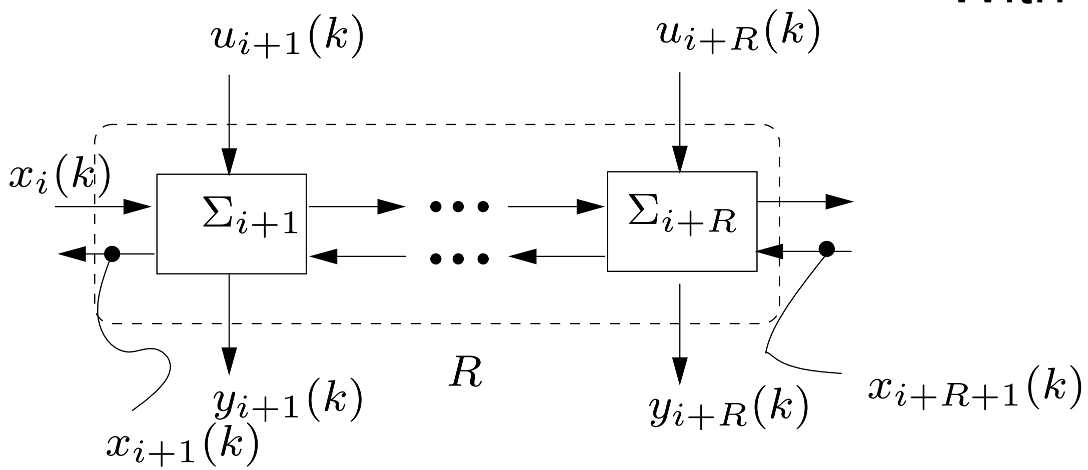
$$
\begin{aligned}
x^R(k+1) &= A^R x^R(k) + B^R u^R(k) \\
&\quad + F^R \begin{bmatrix} x_i(k) \\ x_{i+R+1}(k) \end{bmatrix} \\
y^R(k) &= C^R x^R(k)
\end{aligned}
$$

- Likewise of the R systems to the left of Σ_{i+2}.

TU Delft

Strategy to solve the "one" Identification Problem

- Identify the state sequence $x^R(k)$ of the lifted system of R-systems to the right of Σ_i.



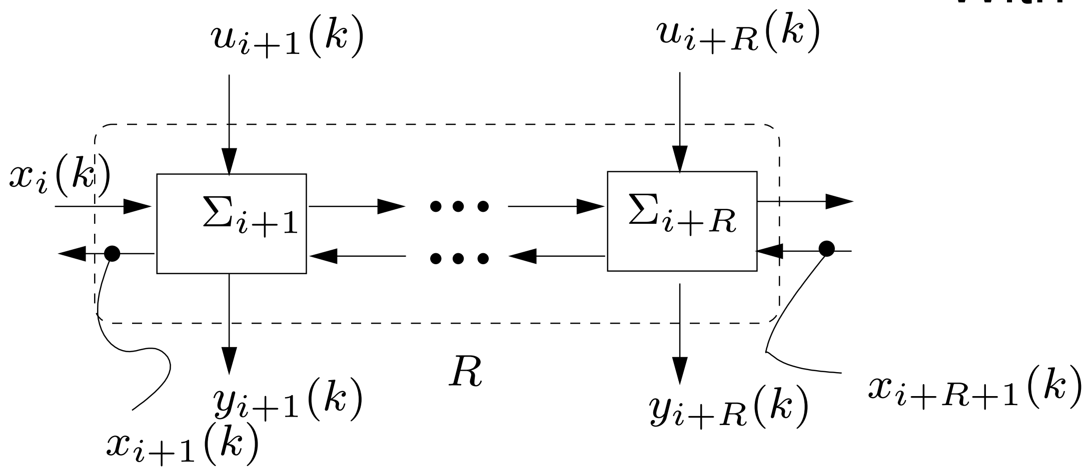
With that lifted system denoted as:

$$\begin{aligned} x^R(k+1) &= A^R x^R(k) + B^R u^R(k) \\ &\quad + F^R \begin{bmatrix} x_i(k) \\ x_{i+R+1}(k) \end{bmatrix} \\ y^R(k) &= C^R x^R(k) \end{aligned}$$

- Likewise of the R systems to the left of Σ_{i+2}.
- $x_{i+1}(k) = x^R(k) \bigcap x^L(k)$

Strategy to solve the "one" Identification Problem

- Identify the state sequence $x^R(k)$ of the lifted system of R-systems to the right of Σ_i.



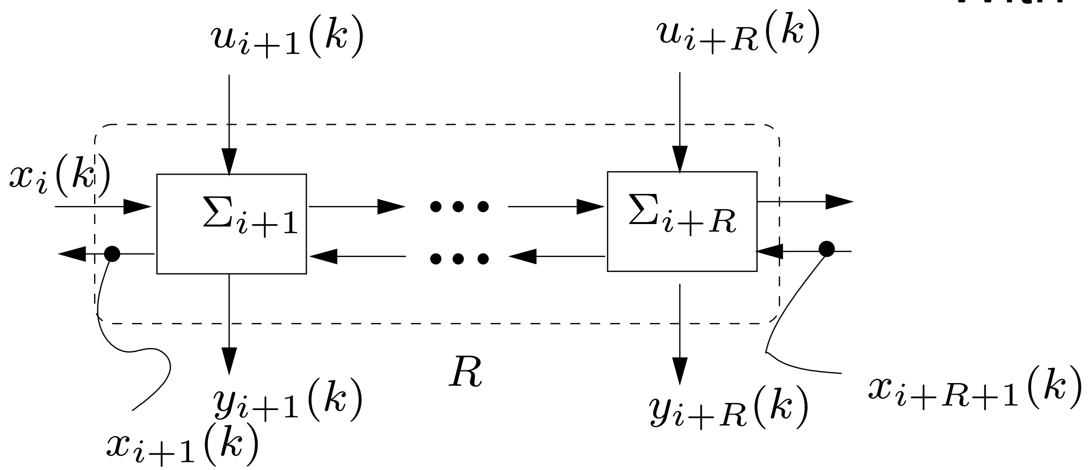
With that lifted system denoted as:

$$\begin{aligned} x^R(k+1) &= A^R x^R(k) + B^R u^R(k) \\ &\quad + F^R \begin{bmatrix} x_i(k) \\ x_{i+R+1}(k) \end{bmatrix} \\ y^R(k) &= C^R x^R(k) \end{aligned}$$

- Likewise of the R systems to the left of Σ_{i+2}.
- $x_{i+1}(k) = x^R(k) \bigcap x^L(k)$
- Likewise determine $x_{i-1}(k)$

Strategy to solve the "one" Identification Problem

- Identify the state sequence $x^R(k)$ of the lifted system of R-systems to the right of Σ_i.



With that lifted system denoted as:

$$\begin{aligned} x^R(k+1) &= A^R x^R(k) + B^R u^R(k) \\ &\quad + F^R \begin{bmatrix} x_i(k) \\ x_{i+R+1}(k) \end{bmatrix} \\ y^R(k) &= C^R x^R(k) \end{aligned}$$

- Likewise of the R systems to the left of Σ_{i+2}.
- $x_{i+1}(k) = x^R(k) \bigcap x^L(k)$
- Likewise determine $x_{i-1}(k)$
- Determine the system matrices A_i, B_i, C_i.

Solution to the first step

Lemma 1: When the lifted system

$$\begin{aligned} x^R(k+1) &= A^R x^R(k) + B^R u^R(k) + F^R \begin{bmatrix} x_i(k) \\ x_{i+R+1}(k) \end{bmatrix} \\ y^R(k) &= C^R x^R(k) \end{aligned}$$

is strongly observable, i.e. the compound matrix $\begin{bmatrix} C^R \\ C^R A^R \\ \vdots \\ C^R (A^R)^{s-1} \end{bmatrix}, \begin{bmatrix} 0 & & \\ C^R F^R & \cdots & 0 \\ \vdots & & \ddots \\ C^R (A^R)^{s-2} F^R & & & C^R F^R \end{bmatrix}$ has full rank, then the *ordinary* intersection algorithm of Moonen et. al can be used to find $x^R(k)$.

Illustration

Consider a homogeneous 1D network of 40 systems given by the following systemm matrices:

$$A_i = \begin{bmatrix} 0.2728 & -0.2068 \\ 0.1068 & 0.2728 \end{bmatrix}, A_{i,i-1} = \begin{bmatrix} -0.1195 & -0.3565 \\ 0.0874 & -0.1048 \end{bmatrix}$$

$$A_{i,i+1} = \begin{bmatrix} 0.0699 & -0.4278 \\ 0.3842 & 0.1135 \end{bmatrix}, B_i = \begin{bmatrix} 0.3870 \\ -1.2705 \end{bmatrix}$$

$$C_i = \begin{bmatrix} -0.9075 & -1.3651 \end{bmatrix} \text{ for } i = 1, \cdots, 40.$$

The system input in the simulation is generated randomly following the standard Gaussian distribution.

Illustration (Ct'd)

We considered $s = 10$ and $R = 7$ and $N_t = 2000$. Further 200 Monte Carlo trials are made. The estimated poles of the system for $i = 20$ are:

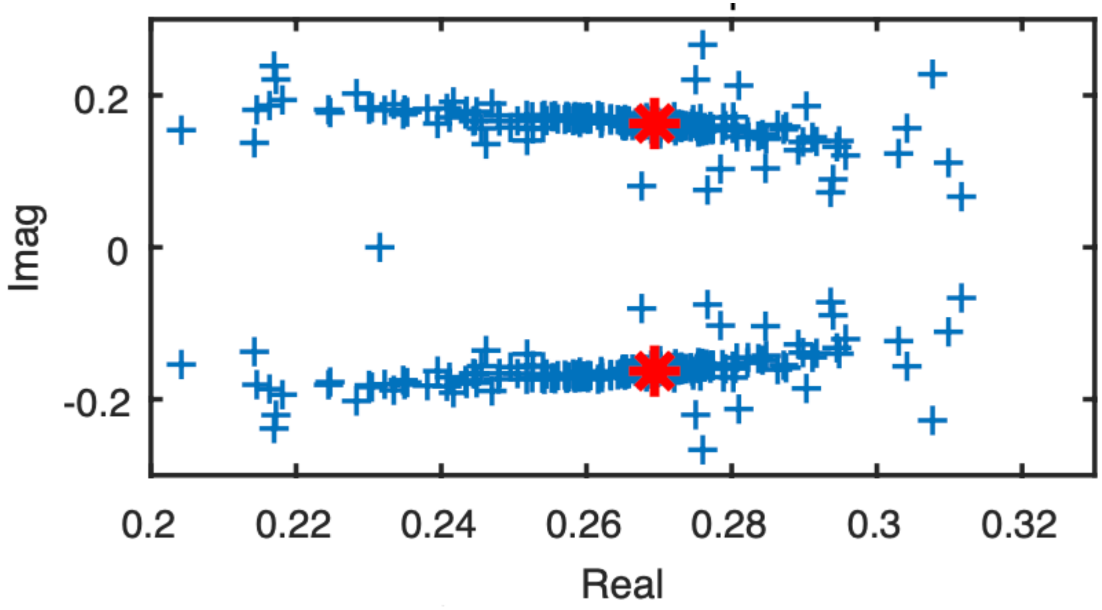
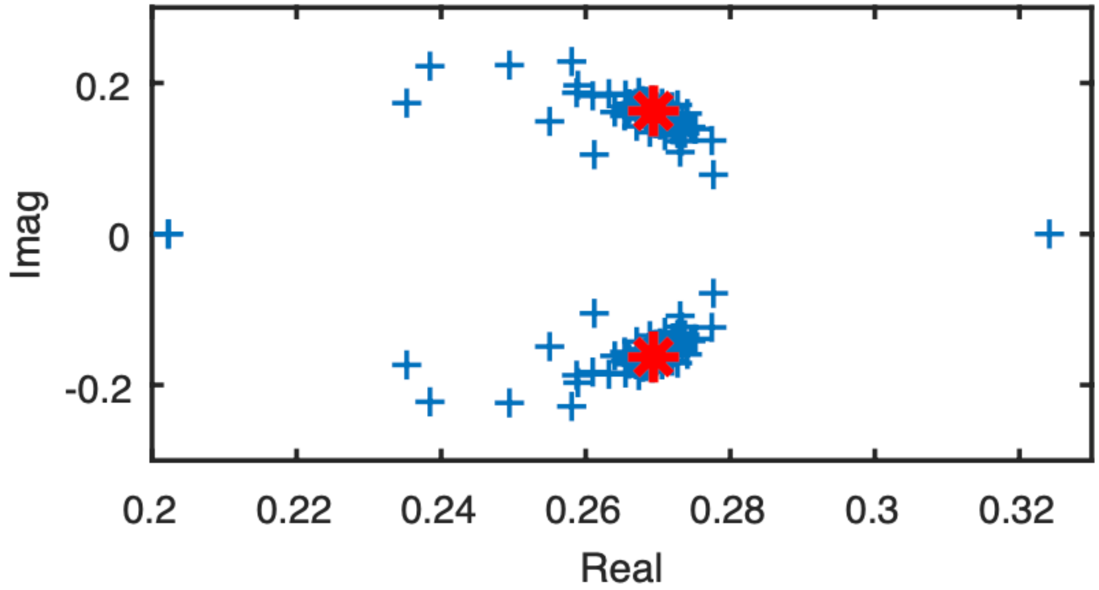


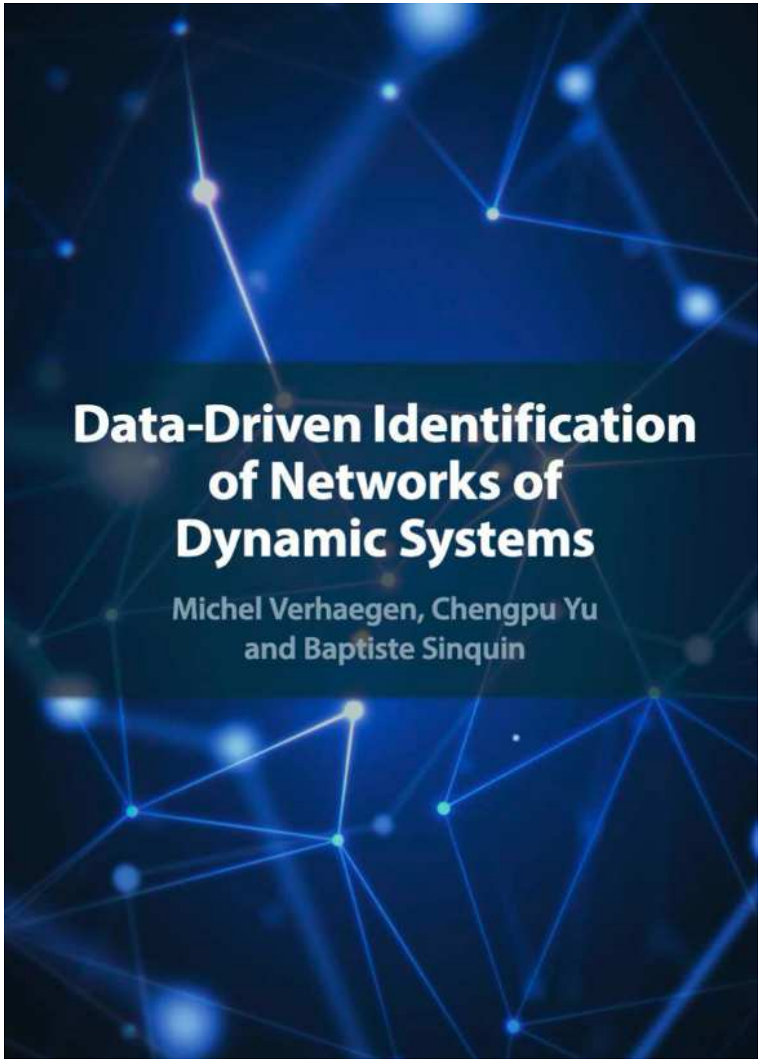
Illustration (Ct'd)

We considered $s = 10$ and $R = 7$ and $N_t = 8000$. Further 200 Monte Carlo trials are made. The estimated poles of the system for $i = 20$ are:



Part 3: Applications

The book is readily available



M.Verhaegen, Chengpu Yu and Baptiste Sinquin: "Data-Driven Identification of Networks of Dynamic Systems"

Acknowledgement
EU: ERC-Advanced Grant iCON 716669