Full module network identification

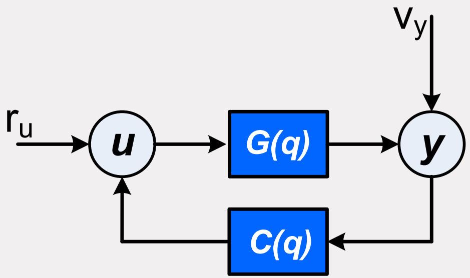
Paul Van den Hof

Doctoral School Lyon, France, 11-12 April 2024

www.sysdynet.eu
www.pvandenhof.nl
p.m.j.vandenhof@tue.nl

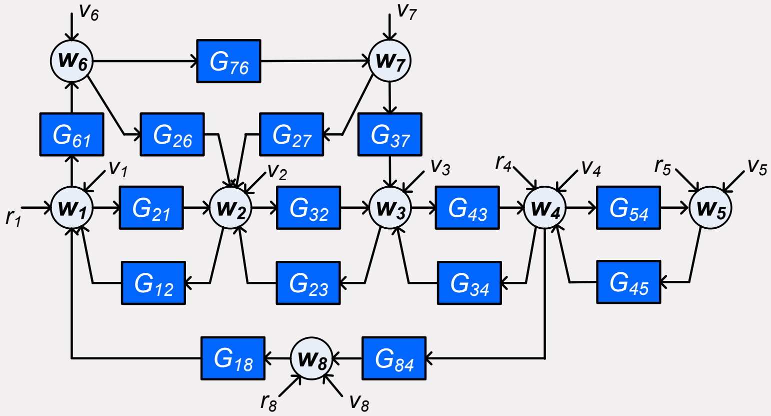
European Research Council

Network identification

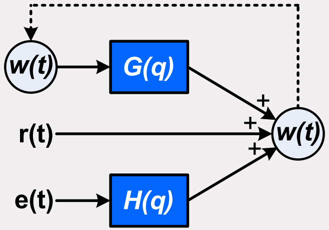


From classical models…

…to dynamic network models



Nodes: input *and* output



Network identification

$$w(t) = G(q)w(t) + R(q)r(t) + H(q)e(t)$$

Derivation of the predictor - In case of full rank disturbances ($H(q)$ square and invertible)*:

$$e(t) = H(q)^{-1}[(I - G(q))w(t) - R(q)r(t)]$$

$$\begin{aligned} w(t) &= G(q)w(t) + R(q)r(t) + (H(q) - I)e(t) + e(t) \\ &= G(q)w(t) + R(q)r(t) + (I - H(q)^{-1})[(I - G(q))w(t) - R(q)r(t)] + e(t) \\ &= [I - H(q)^{-1}(I - G(q))]w(t) + [H(q)^{-1}R(q)]r(t) + e(t) \end{aligned}$$

$$\begin{aligned} \hat{w}(t|t-1) = \bar{E}\{w(t) \mid w^{t-1}, r^t\} &= [I - H(q)^{-1}(I - G(q))]w(t) + [H(q)^{-1}R(q)]r(t) \\ &= [\underbrace{I - H(q)^{-1}}_{\text{``output''}} + \underbrace{H(q)^{-1}G(q)}_{\text{``input''}}]w(t) + H(q)^{-1}R(q)r(t) \end{aligned}$$

* For simplicity we assume G strictly proper

Network identification

$$w(t) = G(q)w(t) + R(q)r(t) + H(q)e(t)$$

Predictor model:

$$\hat{w}(t|t-1;\theta) = [I - H(q,\theta)^{-1} + H(q,\theta)^{-1}G(q,\theta)]w(t) + H(q,\theta)^{-1}R(q,\theta)r(t)$$

This leads to a prediction error:

$$\varepsilon(t,\theta) = H(q,\theta)^{-1}[(I - G(q,\theta))w(t) - R(q,\theta)r(t)]$$

and a prediction error estimator:

$$\hat{\theta}_N = \arg\min_{\theta} \frac{1}{N}\sum_{t=0}^{N-1} \varepsilon(t,\theta)^T Q \varepsilon(t,\theta) \qquad Q > 0$$

Network identification

This is a consistent estimator, under the following conditions:

- System is in the model set, $\mathcal{S} \in \mathcal{M}$
- Model set $\mathcal{M}$ is globally network identifiable at $\mathcal{S}$
- There are no algebraic loops in the network (every loop has a delay)
- The present r signals are persistently exciting of a sufficiently high order (data-informativity)

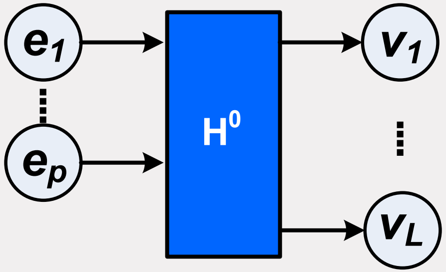
If disturbances are uncorrelated, i.e. $H(q,\theta)$ diagonal:
problem decomposed in L MISO problems

Network identification – reduced rank

What can we do if the disturbances are not full rank?

i.e. $\Phi_v(\omega)$ does not have full rank for all ω

In large scale networks there may be common sources behind multiple disturbances



$p < L$

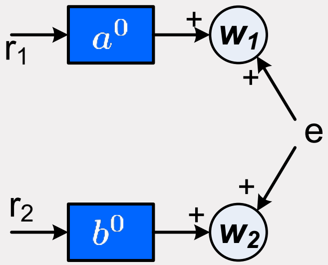
This situation is not really treated in the classical PEM literature

but known in (dynamic) factor analysis[1]

[1] Deistler, Scherrer and Anderson, 2015

Network identification – reduced rank

A simple example: 2 nodes disturbed by 1 noise



$$\varepsilon_1(t,a) = w_1(t) - a r_1(t)$$
$$\varepsilon_2(t,b) = w_2(t) - b r_2(t)$$

Estimate a^0 with parameter a by minimizing $\frac{1}{N}\sum_{t=0}^{N-1} \varepsilon_1(t,\theta)^2$

but note that using the dependency $\varepsilon_1(t,a^0) = \varepsilon_2(t,b^0)$ for the model leads to

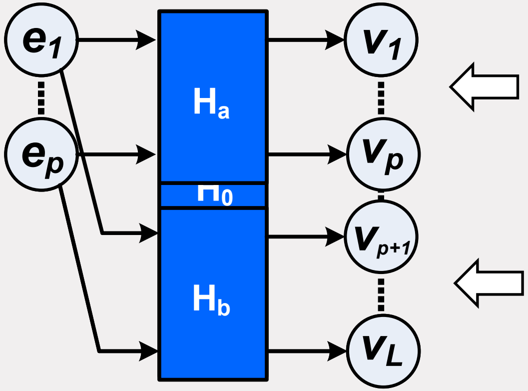
$$\varepsilon_1(t,a) = \varepsilon_2(t,b) \Leftrightarrow e + (a^0 - a)r_1 = e + (b^0 - b)r_2$$

which for persistently excitating and independent r signals gives variance-free estimates $\hat{a} = a^0$ and $\hat{b} = b^0$.

[1] N. Everitt et al., SYSID, 2015.

Network identification – reduced rank

Noise can have $\dim(e) \leq \dim(v)$



‘full rank’ v_a

‘dependent’ v_b

Assumption: $\begin{bmatrix} v_1 \\ \vdots \\ v_p \end{bmatrix}$ is a full rank noise

Can be detected from data

$$H = \begin{bmatrix} H_a \\ H_b \end{bmatrix}$$

H_a is monic, i.e. $\lim_{z\to\infty} H_a(z) = I$

H_b is non-square

Network identification – reduced rank

Particular spectral factorization result of a stochastic process v with rank $p \leq L$ and with $(v_1 \cdots v_p)$ full rank:

$$\Phi_v = \breve{H}\breve{\Delta}\breve{H}^*$$

with $\breve{H} = \begin{bmatrix} H_a & 0 \\ H_b - \Gamma & I \end{bmatrix}$, $\breve{\Delta} = \begin{bmatrix} I \\ \Gamma \end{bmatrix} \Delta \begin{bmatrix} I \\ \Gamma \end{bmatrix}^T$, $\Delta > 0$, $\Gamma = \lim_{z\to\infty} H_b(z)$

such that $\breve{H}$ and H_a are monic, stable and minimum-phase[1]

Consequently: $v(t) = \breve{H}^0(q)e(t) = \begin{bmatrix} H_a^0 & 0 \\ H_b^0 - \Gamma^0 & I \end{bmatrix} \begin{bmatrix} e \\ \Gamma^0 e \end{bmatrix}$

with $\begin{bmatrix} e_a \\ e_b \end{bmatrix} := \breve{H}^0(q)^{-1} v(t)$ it follows that $\Gamma^0 e_a(t) - e_b(t) = 0$ for all t

[1] Cao, Picci & Lindquist (Automatica, 2023) have shown that this requires the assumption that the deterministic mapping from v_a to v_b is stable.

Network identification – reduced rank

$$\varepsilon(t,\theta) = \begin{bmatrix}\varepsilon_a(t,\theta)\\ \varepsilon_b(t,\theta)\end{bmatrix} = \begin{bmatrix} H_a(q,\theta) & 0\\ H_b(q,\theta)-\Gamma(\theta) & I\end{bmatrix}^{-1} [(I-G(q,\theta))w(t) - R(q,\theta)r(t)]$$

with the constraint: $\Gamma(\theta)\varepsilon_a(t,\theta) - \varepsilon_b(t,\theta) = 0$ for all t

Weighted least-squares method: discard dependencies in noise:

$$\theta^\star = \arg\min_\theta \mathbb{E}\, \varepsilon^T(t,\theta)\; Q\; \varepsilon(t,\theta) \qquad Q > 0$$

leads to consistent estimates of the network,
under the same conditions as for the full rank case

However: For ML / minimum variance results we typically would need $Q = [cov(e)]^{-1}$
but $cov(e)$ is not invertible

TU/e

Network identification

Constrained least-squares method: include the noise constraint:

$$\theta^\star = \arg\min_\theta \bar{\mathbb{E}} \, \{\varepsilon_a^T(t,\theta) \; Q_a \; \varepsilon_a(t,\theta)\} \qquad\qquad Q_a > 0$$

$$\text{subject to } \bar{\mathbb{E}}\{Z^T(t,\theta)Z(t,\theta)\} = 0$$

$$\text{with } Z(t,\theta) := \Gamma(\theta)\varepsilon_a(t,\theta) - \varepsilon_b(t,\theta)$$

leads to consistent estimates of the network,
under the same conditions as before

This approach provides Maximum Likelihood Estimates,
through constrained optimization of the log-likelihood function

Network identification

Constraint can be infeasible $\Longrightarrow$ Relax the constraint

$$\theta^\star = \arg\min_\theta \bar{\mathbb{E}} \, \{\varepsilon_a^T(t,\theta) \; Q_a \; \varepsilon_a(t,\theta)\} + \lambda\bar{\mathbb{E}}\{Z^T(t,\theta)Z(t,\theta)\}$$

with $Z(t,\theta) := \Gamma(\theta)\varepsilon_a(t,\theta) - \varepsilon_b(t,\theta)$
and λ a tuning parameter

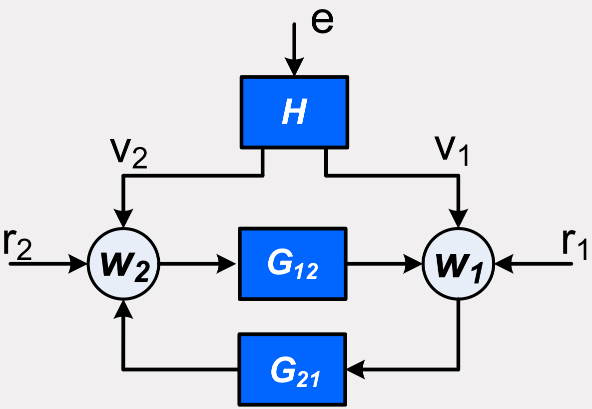
Which is a WLS with parameterized weight $\theta^\star = \arg\min_\theta \bar{\mathbb{E}} \, \{\varepsilon_a^T(t,\theta) \; Q_\lambda \; \varepsilon_a(t,\theta)\}$

$$Q_\lambda(\Gamma) = \begin{bmatrix} Q_a + \lambda\Gamma^T(\theta)\Gamma(\theta) & -\lambda\Gamma^T(\theta) \\ \lambda\Gamma(\theta) & \lambda I \end{bmatrix}$$

Computationally more attractive

Network identification – simulation example

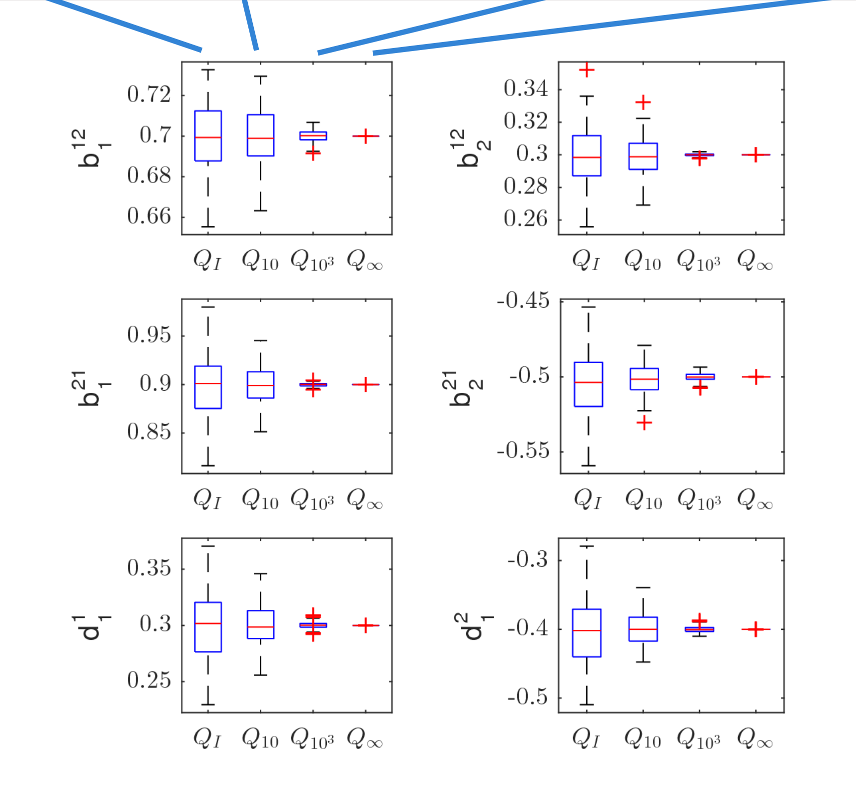
$N = 1000$ samples,
100 realizations of data,
$\sigma_e^2 = 100\sigma_{r_1}^2 = 100\sigma_{r_2}^2$



$$G_{ij}(q,\theta_0) = b_1^{ij} q^{-1} + b_2^{ij} q^{-2}$$
$$H_i(q,\theta_0) = \frac{1}{1+d_1^i q^{-1}}$$

6 parameters in total

WLS: $Q = I$,   Relaxed $\lambda = 10$,   Relaxed $\lambda = 1000$,   CLS



Variance-free when dependencies taken into account

Weerts et al, Automatica, December 2018

Network identification

However: methods based on non-convex optimization scale poorly to larger dimensions

Alternatives: Sequential / multi-step methods / weighted nullspace fitting[1]-[3]

- Estimate a (regularized) high-order ARX model
- Either reconstruct the innovation signal to become a measured input, or approximate the high-order model with linear techniques
- Iterate to find the optimal criterion weighting for optimal variance

To be further developed to arrive at robust algorithms

[1] Galrinho, Rojas and Hjalmarsson, TAC 2019
[2] Weerts et al, SYSID 2018
[3] Fonken et al, Automatica, 2022