Consistent identification of dynamic networks subject to white noise using Weighted Null-Space Fitting

Citation for published version (APA):
Fonken, S. J. M., Ferizbegovic, M., & Hjalmarsson, H. (2020). Consistent identification of dynamic networks subject to white noise using Weighted Null-Space Fitting. IFAC-PapersOnLine, 53(2), 46-51. https://doi.org/10.1016/j.ifacol.2020.12.047

DOI:
10.1016/j.ifacol.2020.12.047

Document status and date:
Published: 01/01/2020

Document Version:
Publisher’s PDF, also known as Version of Record (includes final page, issue and volume numbers)

Please check the document version of this publication:

• A submitted manuscript is the version of the article upon submission and before peer-review. There can be important differences between the submitted version and the official published version of record. People interested in the research are advised to contact the author for the final version of the publication, or visit the DOI to the publisher’s website.
• The final author version and the galley proof are versions of the publication after peer review.
• The final published version features the final layout of the paper including the volume, issue and page numbers.

Link to publication

General rights
Copyright and moral rights for the publications made accessible in the public portal are retained by the authors and/or other copyright owners and it is a condition of accessing publications that users recognise and abide by the legal requirements associated with these rights.

• Users may download and print one copy of any publication from the public portal for the purpose of private study or research.
• You may not further distribute the material or use it for any profit-making activity or commercial gain
• You may freely distribute the URL identifying the publication in the public portal.

If the publication is distributed under the terms of Article 25fa of the Dutch Copyright Act, indicated by the “Taverne” license above, please follow below link for the End User Agreement:
www.tue.nl/taverne

Take down policy
If you believe that this document breaches copyright please contact us at:
openaccess@tue.nl
providing details and we will investigate your claim.
1. INTRODUCTION

A dynamic network consists of spatially interconnected systems, where these interconnections cause the systems to be dynamically correlated with each other. Dynamic networks are generally subjected to external variables, such as excitation and noise signals. Measured signals are represented by nodes in the network while the systems causing the interactions between the nodes are called modules.

With a rising interest in data-driven modeling for identification in dynamic networks, several challenges should be addressed. These can be roughly categorized as follows: detection of network topology (Materassi and Innocenti, 2010; Materassi and Salapaka, 2012), conditions for network identifiability (Gonçalves and Warnick, 2008; Weerts et al., 2015, 2016, 2018a; Gevers et al., 2017), and identification of modules in the dynamic network. Estimating all modules in a dynamic network can be accomplished with the prediction error method (PEM), but often requires non-convex optimization. For large and complex networks the computational effort required for non-convex problems will increase, in addition, the number of local minima tend to grow as the number of modules to estimate increases.

In terms of parameter estimation, the literature focuses on obtaining consistent estimates of separate modules or a collection of modules in a dynamic network. Estimating a particular module in a dynamic network can be achieved with closed-loop identification methods. Methods such as the direct method (Ljung, 1999) and joint-IO, obtain consistent estimates when the noise model has been exactly modeled. Methods that do not depend on exactly estimating the noise model are the two-stage method (Van den Hof and Schrama, 1993) and the instrumental variable (IV) method (Söderström and Stoica, 1983). Both methods require the external excitation signals to be (partially) measurable to obtain consistent estimates. The IV method is the only aforementioned method that does not employ a cost function. A framework for identifying a single module or a collection of modules in dynamic networks has been established in Van den Hof et al. (2013), generalizing the aforementioned closed-loop identification methods. These methods can be applied to one or several multi-input-single-output (MISO) closed-loop problems. Related studies on identifying separate modules or sub-modules in a dynamic network are given in (Dankers et al., 2015; Galrinho et al., 2017; Everitt et al., 2018a,b).

Estimating the modules or collection of modules separately comes at a cost in estimation accuracy and asymptotic efficiency because certain correlations due to the interconnections are not considered.

The subject of identifying all modules simultaneously in dynamic networks is barely covered in system identification literature. The joint-direct method (Weerts et al., 2018b) is an identification approach that does predict all nodes jointly. The identification criterion used here is still in general a non-convex optimization problem.
Assuming the topology of the network is known, the aim of this paper is to propose a method to simultaneously estimate all modules in a dynamic network providing consistent and asymptotically efficient estimates, without the need to solve non-convex optimization problems. The Weighted Null-Space Fitting (WNSF) method (Galrinho et al., 2019) has been shown to provide consistent and asymptotically efficient estimates for single-input-single-output (SISO) systems. This method avoids non-convex optimization, and extensions to multivariate systems and cascaded systems are available (Galrinho et al., 2018). For networks that can be written as multivariate autoregressive moving average exogenous (ARMAX) models, a multi-step least-squares method is proposed in (Weerts et al., 2018). It is shown that this method can be interpreted as WNSF applied to multivariate ARMAX-models. Here we continue these developments and extend WNSF to linear dynamic networks of known topology, where the nodes are subject to white disturbances rather than the type of noise models considered in the ARMAX case.

The paper proceeds with a problem statement in Section 2 and describes the dynamic network setup that is considered for the extension of the WNSF method. Section 3 provides relevant background regarding the WNSF method, followed by the proposed extension on the WNSF method in Section 4. Section 5 provides a simulation to verify the theoretical results. Here WNSF for dynamic networks is compared to PEM initialized at the true parameters.

2. PROBLEM STATEMENT

A dynamic network can be expressed by \( L \) internal signals or nodes denoted as \( w(t) = [w_1(t), \ldots, w_L(t)]^\top \). The modules considered are time invariant (LTI) systems. Following Van den Hof et al. (2013), the network is expressed as

\[
\begin{align*}
    w(t) &= G(q, \theta)w(t) + R(q, \theta)r(t) + e(t) \\
    w(t) &= (I - G(q, \theta))^{-1} (R(q, \theta)r(t) + e(t)),
\end{align*}
\]

where

- \( G(q, \theta) \) is an off-diagonal matrix with its elements \( G_{ij}(q, \theta) \) either stable proper rational transfer functions containing at least one delay, or zero.
- \( r(t) = [r_1(t), \ldots, r_M(t)]^\top \) is a vector of external excitation signals, and \( R(q, \theta) \) is an \( L \times M \) transfer function matrix with proper rational elements. In addition, \( R(q, \theta) \) is assumed full rank.
- \( e(t) = [e_1(t), \ldots, e_L(t)]^\top \) is a vector of unmeasured white noise sequences with zero mean and \( \mathbb{E}[e(t)e^\top(t)] = \Delta \delta_{t-s} \) where \( \delta_r \) is the Kronecker delta function.

It is assumed that the nodes \( w(t) \) and the external excitations \( r(t) \) are available to the user and are sufficiently informative so that the network is identifiable. In addition, it is assumed that the network is well-posed, implying that all principle minors of \( (I - G(q, \infty))^{-1} \) are nonzero ( Dankers, 2014 ). Thus \( I - G(q, \theta) \) is invertible and the inverse consists of causal transfer functions. In addition, the topology of the network is assumed to be known. Furthermore, we assume the data is generated according to (1) with \( \theta = \theta_0 \), where \( \theta \) consists of numerator and denominator coefficients of the transfer functions \( G_{ij}, i, j = 1, \ldots, L \), and \( R_{ij}, i = 1, \ldots, L, j = 1, \ldots, M \).

The objective of the paper is to obtain a method that can estimate \( \theta \) without non-convex optimization.

3. WEIGHTED NULL-SPACE FITTING

The Weighted Null-Space Fitting (WNSF) method from Galrinho et al. (2019) and Galrinho (2018) is a multi-step least-squares method drawing on the seminal work of Åström and Wittenmark (1985) where the properties of the least-squares method applied to ARX-models with increasing model order is analyzed. WNSF was originally developed for SISO systems, the origins of this method can be traced back to the work of Durbin (1959), see Galrinho (2018). Below we give a brief review which in the next section is followed by our extension to the network case.

The first step is an intermediate step, estimating a non-parametric (high-order) finite impulse response (FIR) or autoregressive exogenous (ARX) model. The second step aims to reduce this estimate to a parametric model using least squares. In the final step, the parametric model is re-estimated with weighted least squares. This leads to a consistent and asymptotically efficient estimate.

Step 1: Non-parametric model

For ease of notation superscript \( \eta \) is used to indicate the non-parametric model while \( \theta \) is used for the parametric model. Consider a SISO output error (OE) model where the plant is defined by \( G(q, \theta) \), assumed to have one delay. The stable rational function \( G(q, \theta) \) can be well approximated using a non-parametric FIR model \( y(t) = B^n u(t) + \epsilon(t) \) with \( B^n = \sum_{k=1}^n b_k q^{-k} \), by choosing a sufficiently large order \( n \). This model can be written in regressor form, resulting in the least-squares estimate

\[
\hat{b}_N^n = \left[ \frac{1}{N} \sum_{t=1}^{N} \varphi(t) \varphi^\top(t) \right]^{-1} \frac{1}{N} \sum_{t=1}^{N} \varphi(t) y(t).
\]

where

\[
\varphi(t) = [u(t-1) \cdots u(t-n)]
\]

The estimation error can be expressed as \( \epsilon_n^\eta = \hat{b}_N^n - g_0^n \), with \( g_0^n \) being the vector of the first impulse response coefficients of the true system. This error has approximately a normal distribution with zero mean and covariance

\[
P = \sigma_e^2 \left[ \frac{1}{N} \sum_{t=1}^{N} \varphi(t) \varphi^\top(t) \right]^{-1},
\]

where \( \sigma_e^2 \) is the noise variance of \( \epsilon(t) \).

Step 2: Reduction to parametric model

While the estimate in Step 1 is attractive from a computational point of view, its variance will typically be high since the number of parameters, \( n \), has to be large in order for the approximation to be valid. Reducing the non-parametric model from the previous step to a parametric model allows us to reduce the variance. Neglecting the truncation error, the relation between the models is given by

\[
B^n = \sum_{k=1}^{n} b_k q^{-k} = \frac{L^n}{F_{\theta}} = G^\theta
\]

By rewriting (5) to
the relation can be rewritten to be linear in the parameters $\theta$ according to

$$b^n - Q(b^n)\theta = 0,$$

with $Q(b^n) = \left[ -Q^k(b^n) I_{m \times m} \right]_n^{n \times m}$. \hfill (7)

where $Q(b^n) = T_n \times \{1 \ b_1 \cdots b_{m-1}\}$ is a lower triangular Toeplitz matrix with $[0 \ b_1 \cdots b_{m-1}]$ in the first column. By substituting the estimate of $b^n$ in $Q(b^n)$ an estimate of $\theta$ can be obtained from (7) via weighted least squares

$$\hat{\theta}_N = \left[ Q^\top(b_N) W Q(\hat{b}_N) \right]^{-1} Q^\top(\hat{b}_N) W b^n,$$ \hfill (8)

where $W$ is either set to $W = I$ or $W = P^{-1}$ for obtaining the initial estimate $\hat{\theta}_N^{(0)}$. Notice that the unknown noise variance $\sigma^2_n$ appears in (4), but since it is a scalar quantity it can be discarded in the weighting matrix without affecting the estimate.

**Step 3: Re-estimation of the parametric model**

When substituting the estimates of $b^n$ into the left-hand side of (7), the expression no longer equals zero but instead, neglecting the truncation error, it holds that

$$\hat{\theta}_N = Q^\top(b_N) W Q(\hat{b}_N),$$ \hfill (9)

with $T(\theta) = T_n \times \{1 \ f_1 \cdots f_m\}$, depending on only the denominator coefficients of the parametric model. With $c^n$ being approximately zero mean normal with covariance matrix $P$, it follows that the right-hand side of (9) is approximately zero mean normal with covariance matrix $T(\theta) P T^\top(\theta)$. This correlation structure should be accounted for when estimating $\theta$ and can be handled using weighted least-squares in the following way. The aforementioned covariance matrix can be approximated using the estimate of $\theta$ obtained in Step 2 as $T(\hat{\theta}_N^{(0)}) P T^\top(\hat{\theta}_N^{(0)})$. A new estimate of $\theta$ is formed by setting the weighting matrix $W$ to the inverse of this matrix, i.e.

$$W = W(\hat{\theta}_N^{(0)}),$$ \hfill (10)

It is proven in (Galarinho et al., 2019) that both $\hat{\theta}_N^{(0)}$ and $\hat{\theta}_N^{(1)}$ are consistent estimates and in addition, $\hat{\theta}_N^{(1)}$ is proven to be asymptotic efficient. Since finite sample size $N$ is used in practice it could be beneficial to iterate the last step, i.e. to use $\hat{\theta}_N^{(k)}$ in (10) and then computing $\hat{\theta}_N^{(k+1)}$ using (8), iterating until a final value or stopping criterion is reached, see for an example Section 5.

4. WEIGHTED NULL-SPACE FITTING FOR DYNAMIC NETWORKS

4.1 Algorithm

In this section WNSF is extended such that it is suitable for identifying all the modules in a dynamic network simultaneously. The steps described in the previous section can be followed.

**Step 1: Non-parametric model**

Consider the system defined in (1), where the rational functions of the separate modules in matrices $G(q, \theta)$ and $R(q, \theta)$ are defined as

$$G_{ij}(q, \theta) = \frac{L_{ij}(q, \theta)}{F_{ij}(q, \theta)}, \quad R_{ij}(q, \theta) = \frac{C_{ij}(q, \theta)}{D_{ij}(q, \theta)},$$ \hfill (11)

with

$$L_{ij}(q, \theta) = l_{ij}^1 q^{-1} + \cdots + l_{ij}^m q^{-m}, \quad F_{ij}(q, \theta) = 1 + f_{ij}^1 q^{-1} + \cdots + f_{ij}^m q^{-m},$$

$$C_{ij}(q, \theta) = c_{ij}^1 q^{-1} + \cdots + c_{ij}^m q^{-m}, \quad D_{ij}(q, \theta) = 1 + d_{ij}^1 q^{-1} + \cdots + d_{ij}^m q^{-m}.$$ \hfill (12)

For ease of notation the orders $m_I, m_F, m_L, m_C$ are assumed to be equal for the respective polynomials.

The parameter vector to estimate contains the rational functions of both $G(q, \theta)$ and $R(q, \theta)$ and has structure

$$\theta = [\theta^1 \cdots \theta^L]^\top \in \mathbb{R}^{L(m_I + m_F + m_L + m_C)},$$ \hfill (13)

where

$$\theta^k = [f_i^1 \cdots f_i^L \ b_i^1 \cdots b_i^L]_k,$$ \hfill (14)

$$f_i^j = [f_{ij}^1 \cdots f_{ij}^m]_j, \quad l_i^j = [l_{ij}^1 \cdots l_{ij}^m]_j,$$ \hfill (15)

and

$$d_i^j = [d_{ij}^1 \cdots d_{ij}^m]_j, \quad c_i^j = [c_{ij}^1 \cdots c_{ij}^m]_j.$$ \hfill (16)

Moreover, to satisfy the structure of $G(q, \theta)$ given in (1) the elements $f_i^j$ and $l_i^j$ corresponding to the diagonal elements are omitted for $i = 1, \ldots, L$ since these elements are zero.

In the first step a non-parametric ARX model is used

$$\tilde{A}^n y(t) = B^n r(t) + e(t),$$ \hfill (17)

where

$$\tilde{A}^n = I_L - \begin{bmatrix} a_{11} \cdots a_{1L} \\ \vdots \vdots \\ a_{L1} \cdots a_{L1} \end{bmatrix}, \quad a_{ij} = \sum_{k=1}^n a_{ij}^k q^{-k},$$

with the parameter vector $\eta^n$ is defined as

$$\eta^n = [\eta^1 \cdots \eta^L]^\top,$$ \hfill (18)

with $\eta^j = [a_1^j \cdots a_L^j \ b_1^j \cdots b_L^j],$ \hfill (19)

and $a_{ij}^k = [a_{ij}^1 \cdots a_{ij}^m], \quad b_{ij}^k = [b_{ij}^1 \cdots b_{ij}^m].$

The dynamic network, described by the non-parametric model (16), can be expressed in regressor form

$$y(t) = \varphi_\top(t) \eta^n + e(t),$$ \hfill (20)

where the elements of $\varphi(t)$ are functions of delayed outputs and excitation signals

$$\varphi_\top(t) = [\varphi_y(t) \varphi_r(t)],$$ \hfill (21)

with

$$\varphi_y(t) = [\varphi_{y1}(t), \cdots, \varphi_{yL}(t)], \quad \varphi_r(t) = [\varphi_{r1}(t), \cdots, \varphi_{rm}(t)],$$ \hfill (22)

and

$$\varphi_{y1}(t) = [y_1(t-1), \cdots, y_L(t-n)], \quad \varphi_{rk}(t) = [r_k(t), \cdots, r_k(t-n-1)].$$ \hfill (23)

The least squares estimate of $\eta^n$ is obtained similar to (2), replacing $\hat{b}_N^\top$ with $\tilde{b}_N^\top$ and using $\varphi(t)$ defined in (20).

Furthermore, the estimation error covariance can be derived as was done in Section 3, see (4), but the noise
is no longer scalar. An appropriate approximation of the covariance matrix is given by
\[
P(\hat{\Lambda}) = \left[ \frac{1}{N} \sum_{t=1}^{N} \varphi(t) \hat{\Lambda}^{-1} \varphi^T(t) \right]^{-1},
\]
(24)
with prediction error \( \varepsilon(t, \hat{\eta}_N^n) = y(t) - \varphi^T(t) \hat{\eta}_N^n \).

**Step 2: Reduction to parametric model**
The non-parametric model (16) is related to the parametric model (1) by \( A^0 = 1 - G^0 \) and \( B^n = R^n \). Since the WNSF algorithm allows element-wise parameterization, the relation using (17) can be rewritten as \( A^n_{ij} = G^n_{ij} \) and \( B^n_{ij} = R^n_{ij} \) giving
\[
A^n_{ij} = \sum_{k=1}^{n} a^i_k q^{-k} = \frac{I^n_{ij}}{I^n_{ij}},
\]
(25)
\[
B^n_{ij} = \sum_{k=0}^{n} b^i_k q^{-k} = \frac{C^n_{ij}}{D^n_{ij}} = R^n_{ij}.
\]
Rewriting this gives
\[
\sum_{k=1}^{n} a^i_k q^{-k} F^n_{ij} - L^n_{ij} = 0, \quad \sum_{k=0}^{n} b^i_k q^{-k} D^n_{ij} - C^n_{ij} = 0.
\]
(26)
The same procedure for determining \( Q(\eta^n) \) from Section 3 can be followed, giving
\[
Q(\eta^n) = \text{diag}(Q^1, \ldots, Q^L),
\]
\[
Q^i(\eta^n) = \begin{bmatrix} Q^{i,l} & 0 \\ 0 & Q^{i,c} \end{bmatrix}
\]
(27)
where \( Q^{i,l} \) and \( Q^{i,c} \) share the same structure, defined by
\[
Q^{i,l} = \text{diag}([-Q^{i,1} , Q^{i,1}], \ldots, [-Q^{i,L} , Q^{i,L}]),
\]
\[
Q^{i,c} = \text{diag}([-Q^{i,1} , Q^{i,1}], \ldots, [-Q^{i,L} , Q^{i,L}]).
\]
(28)
Moreover,
\[
Q^{i,l}(\eta^n) = \mathcal{T}_{n,m} \{ [a^1_l, \ldots, a^L_l]^T \} \quad Q^{i,c}(\eta^n) = I_{n \times m},
\]
\[
Q^{d,l}(\eta^n) = \mathcal{T}_{n,m} \{ [b^0_l, \ldots, b^L_l]^T \} \quad Q^{d,c}(\eta^n) = I_{n \times m},
\]
(29)
where the Toeplitz matrices are lower triangular and \( I_{n \times m} \) has \( I_{m \times n} \) at the top with zeros elsewhere.

The initial estimate \( \hat{\theta}_N^{(0)} \) can now be obtained analogously to (8), using \( Q(\hat{\eta}_N^n) \), and with \( W = P^{-1}(\hat{\Lambda}) \) as weighting.

**Step 3: Re-estimation of the parametric model**
With the initial estimate of \( \hat{\theta} \) obtained in the previous step, the noise variance estimate is updated according
\[
\hat{\Lambda} = \frac{1}{N} \sum_{t=1}^{N} \varepsilon(t, \hat{\theta}_N^{(0)}) \varepsilon^T(t, \hat{\theta}_N^{(0)}),
\]
(30)
and the prediction error with
\[
\varepsilon(t, \hat{\theta}_N^{(0)}) = y(t) - [G(q, \hat{\theta}_N^{(0)}) \ R(q, \hat{\theta}_N^{(0)})] \ y(t).
\]
(31)
Again the same procedure for determining \( T(\theta) \) from Section 3 can be followed. \( T(\theta) \) is now defined by
\[
T(\theta) = \text{diag}(T^1(\theta), \ldots, T^L(\theta))
\]
\[
T^i(\theta) = \text{diag}(T^{f^i}(\theta), \ldots, T^{f^k}(\theta), T^{d^i}(\theta), \ldots, T^{d^n}(\theta)),
\]
(32)
where \( T^{f^i}(\theta) = \mathcal{T}_{n,n} \{ [f^i_1, \ldots, f^i_L]^T \} \), \( T^{d^i}(\theta) = \mathcal{T}_{n,n} \{ [d^i_1, \ldots, d^i_L]^T \} \),
are lower triangular Toeplitz matrices. The weighting matrix can now be defined as \( W = W(\hat{\theta}_N^{(0)}, \hat{\Lambda}) \), where
\[
W(\theta, \Lambda) := T^{-1}(\theta) P^{-1}(\Lambda) T^{-1}(\theta).
\]
(33)
Again analogously to (8) the new estimate \( \hat{\theta}_N^{(1)} \) is obtained by using the newly defined \( W \) in (34) and \( Q(\hat{\eta}_N^n) \) in (27).

**Algorithm 1** The algorithm for WNSF suitable for general dynamic networks is constructed as

(1) estimate non-parametric AR model (16), using least squares (2) to obtain \( \hat{\eta}_N^n \),
(2) reduce the non-parametric model to a parametric model (8), using \( Q(\hat{\eta}_N^n) \) defined in (27) and \( W = P^{-1}(\hat{\Lambda}) \) from (24), obtaining \( \hat{\theta}_N^{(0)} \),
(3) improve the estimates by updating weighting matrix in (8) to \( W(\hat{\theta}_N^{(0)}, \hat{\Lambda}) \) according to (34) with \( \hat{\Lambda} \) from (30), resulting in \( \hat{\theta}_N^{(1)} \). Continuing to iterate gives \( \hat{\theta}_N^{(k+1)} \).

### 4.2 Theoretical analysis

According to Remark 5 (Chapter 3) in Ljung and Wahlberg (1992) their asymptotic results for SISO ARX models can be extended to multivariate systems. The SISO case is therefore used to prove the asymptotic properties of the dynamic network by considering the element-wise notation. The data generating system, denoted in the element-wise ARX structure is given by
\[
A^n_{ij}(q) = \sum_{k=1}^{\infty} a^{0,i}_k q^{-k}, \quad B^n_{ij}(q) = \sum_{k=0}^{\infty} b^{0,i}_k q^{-k}
\]
(35)
It should be observed that the term \( b^{0,i}_k \) is no longer necessarily assumed zero, since \( R(q, \theta) \) does not have to contain a delay. Although this case is not covered in Assumption S1 of Ljung and Wahlberg (1992), Remark 4 (Chapter 2) is still true, meaning the impulse responses still decrease at a certain lowest rate and Assumption S1 still holds. In addition, the input is obtained in open loop, where the sequence \{\( r(t) \)\} is assumed to be sufficiently exciting and independent of \{\( e(t) \)\}, where \( e(t) \) is considered to be Gaussian white noise.

The assumptions and conditions from Galinho (2018) and Ljung and Wahlberg (1992) therefore still hold. This indicates that Theorem 7.1 from Galinho (2018), for applying WNSF to multivariate systems, also is valid for dynamic networks. Hence, for LTI dynamic networks subjected to white noise disturbances, WNSF obeys the following properties under assumptions of identifiability and sufficient richness of the excitation:

(i) \( \hat{\theta}_N^{(0)} \to \theta_0 \), as \( N \to \infty \) w.p.1.,
(ii) \( \hat{\theta}_N^{(1)} \to \theta_0 \), as \( N \to \infty \) w.p.1.,
(iii) \( \sqrt{N}(\hat{\theta}_N^{(1)} - \theta_0) \sim N(0, M) \)
with
\[
M = \lim_{n \to \infty} [Q^T (\eta_0^n) T^{-1} (\theta_0) \bar{P}^{-1}(\Lambda) T^{-1} (\theta_0) Q(\eta_0^n)]^{-1},
\]
where \(\bar{P}^{-1}(\Lambda) = \mathbb{E} [\varphi(t) \Lambda^{-1} \varphi^T(t)]\). (36)

5. NUMERICAL ILLUSTRATION

In this section we compare the performance of Algorithm 1 with PEM initialized at the true parameters by means of a simulation study. PEM provides asymptotically efficient estimates when converging to a global minimum. The simulation results suggest that Algorithm 1 has the same large sample properties as PEM, supporting the theoretical considerations in the preceding section.

In the simulation we consider the following system
\[
G(q, \theta) = 
\begin{bmatrix}
0 & 0 & G_{14} & 0 & 0 \\
0 & 0 & 0 & 0 & G_{26} \\
G_{31} & 0 & 0 & 0 & G_{35} \\
0 & G_{42} & 0 & 0 & 0 \\
G_{51} & 0 & 0 & 0 & G_{56} \\
0 & 0 & G_{63} & 0 & 0 \\
\end{bmatrix},
\]
with the elements \(G_{ij}(q, \theta) = \frac{L_{ij}}{q^{0.5}}\) given by
\[
\begin{bmatrix}
L_{14} \\
L_{26} \\
L_{31} \\
L_{35} \\
L_{42} \\
L_{51} \\
L_{56} \\
L_{63} \\
\end{bmatrix} = \begin{bmatrix}
0.13 & 0 & 0.11 & 0 & 0.13 & 0 & -0.67 & 0.44 & q^{-1} \\
0.11 & 0 & 0.11 & 0 & 0.11 & 0 & -0.11 & 0 & q^{-2} \\
-0.67 & 0.44 & -0.11 & 0 & q^{-1} & q^{-2} & q^{-3} & q^{-4} \\
-0.30 & 0.28 & -0.32 & 0.19 & -0.67 & 0.44 & q^{-2} & q^{-3} & q^{-4} \\
-0.32 & 0.19 & -0.30 & 0.28 & -0.11 & 0 & q^{-3} & q^{-4} & q^{-5} \\
-0.32 & 0.19 & -0.30 & 0.28 & -0.11 & 0 & q^{-4} & q^{-5} & q^{-6} \\
\end{bmatrix},
\]
(38)

and the elements \(R_{ij}(q, \theta)\) given by
\[
\begin{align*}
R_{11} &= \frac{1}{1 + 0.25q^{-1}}, & R_{22} &= \frac{0.20 - 0.40q^{-1}}{1 - 1.30q^{-1} + 0.36q^{-2}}, \\
R_{33} &= \frac{1}{1 + 0.45q^{-1}}, & R_{44} &= \frac{0.50}{1 + 0.58q^{-1}}, \\
R_{55} &= \frac{1.40 + 0.45q^{-1}}{1 + 0.95q^{-1}}, & R_{66} &= \frac{0.21}{1 + 0.73q^{-1}},
\end{align*}
\]
and the elements \(R_{ij}(q, \theta)\) given by
\[
\begin{align*}
R_{11} &= \frac{1}{1 + 0.25q^{-1}}, & R_{22} &= \frac{0.20 - 0.40q^{-1}}{1 - 1.30q^{-1} + 0.36q^{-2}}, \\
R_{33} &= \frac{1}{1 + 0.45q^{-1}}, & R_{44} &= \frac{0.50}{1 + 0.58q^{-1}}, \\
R_{55} &= \frac{1.40 + 0.45q^{-1}}{1 + 0.95q^{-1}}, & R_{66} &= \frac{0.21}{1 + 0.73q^{-1}},
\end{align*}
\]
(40)

and the elements \(R_{ij}(q, \theta)\) given by
\[
\begin{align*}
R_{11} &= \frac{1}{1 + 0.25q^{-1}}, & R_{22} &= \frac{0.20 - 0.40q^{-1}}{1 - 1.30q^{-1} + 0.36q^{-2}}, \\
R_{33} &= \frac{1}{1 + 0.45q^{-1}}, & R_{44} &= \frac{0.50}{1 + 0.58q^{-1}}, \\
R_{55} &= \frac{1.40 + 0.45q^{-1}}{1 + 0.95q^{-1}}, & R_{66} &= \frac{0.21}{1 + 0.73q^{-1}},
\end{align*}
\]
(41)

and the parameters are rounded to two decimal places.

Table 1. Successful computation of estimates with Algorithm 1 given in % over sample sizes

| N   | 300  | 1212 | 4899 | 19797 | 80000 |
|---|---|---|---|---|---|
| Algorithm 1 | 28% | 61% | 91% | 100% | 100% |

Table 2. Average computation time in seconds over sample sizes

| N   | 300  | 1212 | 4899 | 19797 | 80000 |
|---|---|---|---|---|---|
| Algorithm 1 | 15.75 | 7.60 | 7.52 | 22.84 | 99.26 |
| PEM-true | 104.84 | 55.67 | 44.67 | 48.20 | 93.58 |

The excitation signals \(\{r(t)\}\) and noise sequences \(\{e(t)\}\) are normally distributed with zero mean, where the variances of \(\{r(t)\}\) are set to 1 and the variances of \(\{e(t)\}\) are set to \{3, 1, 2, 1, 3, 2.5\}. The output data is gathered

Fig. 1. MSE between \(\hat{\theta}_N\) and \(\theta_{0}\) as function of sample size, averaged over the Monte Carlo runs

Fig. 2. MSE between \(\hat{\Lambda}\) and \(\Lambda_{0}\) as function of sample size, averaged over the Monte Carlo runs
noted that Algorithm 1 does not handle small N well when
the orders of the module transfer functions are increased,
meaning it does not always successfully compute an esti-
mate of $\theta$. This success rate, for the simulation shown in
Fig. 1, is presented in Table 1. Therefore, the MSE is only
taken over the successful Monte Carlo runs.

Table 2 presents the average computation time for both
methods. For Algorithm 1 the computation time is shown
using $\hat{\theta}_{N}^{(i)}$. As seen from the table, the computation
time for Algorithm 1 increases with the sample size.
However, the code has not been optimized and we have
seen that tailoring the code to the network topology can
significantly reduce the computation time.

Fig. 2 shows the MSE for the noise variances $\text{MSE}_{i}(N, \varepsilon) = \frac{1}{N} \sum_{i=1}^{M} ||\mathbf{A}(\varepsilon) - \mathbf{A}_{0}||^{2}$, where only the diagonal elements
$\sigma_{i}^{2}$, and its estimates are evaluated and where the pre-
diction error $\varepsilon$ depends on $\hat{\eta}_{N}^{(i)}$ or $\hat{\eta}_{N}^{(i),f(nal)}$ for Monte
Carlo run $l$. The estimated noise variances converge to
the true noise variances as $N$ increases, except for the noise
variance computed with $\varepsilon(t, \hat{\eta}_{N}^{(i)})$. A possible explanation
for this may be that the truncation error becomes visible.
Besides, it should be noted that there is no significant dif-
ference between deriving the noise variance using predic-
tion error $\varepsilon(t, \hat{\eta}_{N}^{(i)})$ or $\varepsilon(t, \hat{\eta}_{N}^{(i),f(nal)})$. The latter phenomenon
also appears in other simulations we have tested. This
suggests that the improvements in accuracy for $\hat{\theta}_{N}^{(i+1)}$
mainly comes from the improved estimate of $T(\theta_{0})$ used
in the weighting matrix $W(\hat{\theta}_{N}^{(i)}, \Lambda)$ (34). Moreover, other
simulation tests also seem to suggest the $R(q, \theta)$ is not
restricted to be full rank.

The simulation results confirm the consistency of $\hat{\theta}_{N}^{(i),k+1}$
, derived in the preceding section. In addition, the results
suggest that for a large sample size Algorithm 1 is asymp-
totically efficient for $\hat{\theta}_{N}^{(i),k+1}$.

6. CONCLUSION

In this contribution we have presented an extension of
WNSF that is tailored to identify LTI dynamic networks
with known topology, without the need to solve non-
convex optimization problems. The main assumption is
that the nodes are subject to white noise. Simulations on
a fairly challenging network indicate that the method
is consistent and competitive with PEM (even in the
idealized situation when this method is initialized at the
true parameter values). An interesting extension under
study is to handle colored noise.

REFERENCES

Dankers, A., Van den Hof, P., Bombois, X., and Heuberger,
P. (2015). Errors-in-Variables identification in dynamic
networks—consistency results for an instrumental vari-
able approach. Automatica, 62, 39–50.

Dankers, A. (2014). System Identification in Dynamic
networks. Ph.D. thesis, Delft University of Technology.

Durbin, J. (1959). Efficient estimation of parameters in
moving-average models. Biometrika, 46(3/4), 306–316.

Everitt, N., Bottegal, G., and Hjalmarsson, H. (2018a).
An empirical Bayes approach to identification of modules in
dynamic networks. Automatica, 91, 144–151.

Everitt, N., Galrino, M., and Hjalmarsson, H. (2018b).
Open-loop asymptotically efficient model reduction with
the Steiglitz-Mcbride method. Automatica, 89, 221–234.

Galrino, M. (2018). System Identification with Multi-
Step Least-Squares Methods. Ph.D. thesis, KTH Royal
Institute of Technology.

Galrino, M., Everitt, N., and Hjalmarsson, H. (2017).
Incorporating noise modeling in dynamic networks using
non-parametric models. IFAC-PapersOnLine, 50(1),
10568 – 10573.

Galrino, M., Rojas, C., and Hjalmarsson, H. (2019).
Parametric identification using Weighted Null-Space
Fitting. IEEE Tran. on Automatic Control, 64(7),
2798–2813.

Galrino, M., Prota, R., Ferizbegovic, M., and Hjalmar-
son, H. (2018). Weighted Null-Space Fitting for iden-
tification of cascade networks. IFAC-PapersOnLine,
51(15), 856–861.

Gevers, M., Bazanella, A.S., and Parraga, A. (2017).
On the identifiability of dynamical networks. IFAC-
PapersOnLine, 50(1), 10580–10585.

Gonçalves, J. and Warnick, S. (2008). Necessary and suf-
cient conditions for dynamical structure reconstruction
of LTI networks. IEEE Trans. on Automatic Control,
53(7), 1670–1674.

Ljung, L. (1999). System Identification. Theory for the
User, 2nd ed. Prentice-Hall.

Ljung, L. and Wahlberg, B. (1992). Asymptotic proper-
ties of the least-squares method for estimating transfer
functions and disturbance spectra. Advances in Applied
Probability, 24(2), 412–440.

Materassi, D. and Innocenti, G. (2010). Topological
identification in networks of dynamical systems. IEEE
Trans. on Automatic Control, 55(8), 1860–1871.

Materassi, D. and Salapaka, M. (2012). On the problem of
reconstructing an unknown topology via locality prop-
erties of the Wiener filter. IEEE Trans. on Automatic
Control, 57(7), 1765–1777.

Söderström, T. and Stoica, P. (1983). Optimal instru-
mental variable estimation and approximate implementa-
tions. IEEE Trans. on Automatic Control, 28(7),
757–772.

Van den Hof, P. and Schrama, R. (1993). An indirect
method for transfer function estimation from closed loop
data. Automatica, 31(6), 1523–1527.

Van den Hof, P., Dankers, A., Heuberger, P., and Bombois,
X. (2013). Identification of dynamic models in complex
networks with prediction error methods: basic methods
for consistent module estimates. Automatica, 49(10),
2994–3006.

Weerts, H., Dankers, A., and Van den Hof, P. (2015).
Identifiability in dynamic network identification. IFAC-
PapersOnLine, 48(28), 1409–1414.

Weerts, H., Van den Hof, P., and Dankers, A. (2016).
Identifiability of dynamic networks with part of the
nodes noise-free. IFAC-PapersOnLine, 49(13), 19–24.

Weerts, H., Galrino, M., Bottegal, G., Hjalmarsson, H.,
and Van den Hof, P. (2018). A sequential least squares
algorithm for ARMAX dynamic network identification.
IFAC-PapersOnLine, 51(15), 844–849.

Weerts, H., Van den Hof, P., and Dankers, A. (2018a).
Identifiability of linear dynamic networks. Automatica,
89, 247–258.

Weerts, H.H., den Hof, P.M.V., and Dankers, A.G.
(2018b). Prediction error identification of linear dy-
namic networks with rank-reduced noise. Automatica,
98, 256 – 268.