ELSEVIER

Signal Processing 52 (1996) 179–194

SIGNAL PROCESSING

Continuous-time frequency domain subspace system identification [1]

Peter Van Overschee*, Bart De Moor

ESAT/SISTA-Katholieke Universiteit Leuven, Kardinaal Mercierlaan 94, 3001 Leuven (Heverlee), Belgium

Received 26 April 1995; revised 27 February 1996

Abstract

In this paper we present a new subspace identification algorithm for the identification of multi-input multi-output linear time-invariant continuous-time systems from measured frequency response data. We show how the conditioning of the data-matrices in the algorithm can be improved by making use of recursions derived from the Forsythe polynomials. The asymptotic properties are analyzed and it is shown that, when the error distribution on the measurements is given, the algorithm can be made asymptotically unbiased through the introduction of a weighting matrix.

Zusammenfassung

Diese Arbeit stellt einen neuen Unterraum-Identifikationsalgorithmus zur Identifikation von linearen zeitinvarianten zeitkontinuierlichen Systemen mit multiplen Ein- und Ausgängen vor, wobei gemessene Übertragungsfunktionen verwendet werden. Wir zeigen, wie die Konditionierung der Datenmatrizen im Algorithmus durch Verwendung von aus Forsythe-Polynomen abgeleiteten Rekursionen verbessert werden kann. Die asymptotischen Eigenschaften werden analysiert, und es wird gezeigt, daß bei gegebener Fehlerverteilung der Meßwerte der Algorithmus durch die Einführung einer Gewichtungsmatrix asymptotisch erwartungstreu gemacht werden kann.

Résumé

Dans cet article, nous présentons un nouvel algorithme d'identification à l'aide des sous-espaces applicable à l'identification de systèmes en temps continu, invariants dans le temps, linéaires et à entrées et sorties multiples, lorsqu'on a comme données des mesures de la réponse fréquencielle. Nous montrons comment le conditionnement des matrices de données dans l'algorithme peut être amélioré en faisant usage des récursions dérivées des polynômes de Forsythe. Les propriétés asymptotiques sont analysées, et il est montré que, lorsque la distribution de l'erreur sur les mesures est donnée, l'algorithme peut être rendu asymptotiquement non biaisé par l'introduction d'une matrice de pondération.

Keywords: System identification; Frequency domain methods; State-space modeling; Subspace methods

* Corresponding author. Tel.: 32/16/32 10 95; fax: 32/16/32 19 86; e-mail: peter.vanoverschee@esat.kuleuven.ac.be.

[1] This report is available by anonymous ftp from *ftp.esat.kuleuven.ac.be* in the directory */pub/SISTA/vanoverschee/reports/freqdom2.ps.Z.*

This research was supported by Belgian Federal Government Inter-university Attraction Poles IUAP-17 (Modeling and Control of Dynamic Systems), IUAP-50 (Automation in Design and Production), Flemish Government Concerted Action GOA-MIPS (Model-based Information Processing Systems) and the European Commission Human Capital and Mobility SIMONET (System Identification and Modeling Network). The scientific responsibility rests with its authors. Peter Van Overschee is a senior research assistant and Bart De Moor a senior research associate of the Belgian National Fund for Scientific Research.

PII S0165-1684(96)00052-7

1. Introduction

Identification of multi-input multi-output (MIMO) systems of high order is still considered to be a challenge. This type of systems are encountered in the mechanical engineering area of modal analysis and in the control of flexible structures. In most cases it is desired to obtain a single model in a minimal realization which makes state-space models an excellent choice. These models also facilitate the controller design, since most modern control design techniques are state-space model based.

For time domain measurements, a vast number of state-space subspace identification algorithms is available [4, 9, 16–19]. One of the advantages of these algorithms over the classical prediction error methods [11] is that there is no need for an explicit parametrization, which is typically needed to start up the prediction-error algorithms. With subspace identification algorithms the only parameter needed is the system order, which is explicitly determined by inspection of a singular value spectrum.

Another advantage of subspace identification algorithms is the absence of nonlinear parametric optimization problems. Indeed, subspace identification algorithms are noniterative, and thus never get stuck in local minima or suffer from convergence problems. In short, they always produce a (suboptimal) result.

When using sophisticated data analyzers or data acquisition equipment, information about a system is often obtained as frequency response samples instead of time domain samples. The problem addressed in this paper is the one of fitting a continuous-time transfer function through these given measured frequency response samples. A parametric approach to this problem would consist of using a prediction-error like criterion, that is then optimized using a nonlinear search in the parameter space. However, the typical disadvantages of the time domain prediction-error methods also carry through to this frequency domain setting, i.e. an a priori given parametrization is needed (which is especially hard to find for MIMO systems), parametric optimization is required and convergence problems could occur.

More recent noniterative schemes for discrete-time system identification from frequency domain measurements are the impulse realization algorithms of [7, 8], where the impulse response is computed as the inverse discrete Fourier transform of the given frequency samples. The problem with these schemes is that the estimated impulse response samples are biased and that the whole algorithm as such will thus compute biased results. In [12] this problem is solved by using a slightly modified realization algorithm that takes the nature of the bias on the impulse responses into account. However, this technique needs the frequency points to be uniformly spaced on the frequency axis.

In [10] the time domain subspace identification algorithm of [5] is transformed to the frequency domain, resulting in an algorithm for the identification of discrete-time systems from frequency response samples. In [12] the connections between this algorithm and the inverse discrete Fourier transform algorithm of [12] are described.

In this paper we derive a new continuous-time frequency domain subspace system identification algorithm. The major contributions of this paper are the following:

(1) We show how to translate a continuous-time subspace identification algorithm to the frequency domain using the Laplace transformation. The derivation is similar to the one for the discrete-time frequency domain identification algorithm of [10]. This straightforward translation however leads to an algorithm which manipulates extremely badly conditioned data-matrices. We show how this problem can be solved by introducing the Forsythe polynomials, which in [15] were used to solve the conditioning problems of a classical maximum likelihood frequency domain identification algorithm. We show in this paper that the recursions used to construct the Forsythe polynomials can just as well be used to improve the condition number of the data-matrices involved in the frequency domain subspace identification algorithm. Indeed, the recursions can be used to find well conditioned bases for the row spaces of certain data-matrices.

(2) An asymptotic analysis of the obtained algorithm shows that it is asymptotically unbiased when the number of frequency data points goes to infinity (within a fixed, finite frequency band) and when the data is corrupted by a zero mean white error which is relative.[2] This is a striking result, since it shows that

[2] By a white error sequence we mean that the noise on two frequency response samples measured at different frequency points is not correlated.

the Forsythe recursions do not only improve the condition number of the data-matrices involved, but that they also make the algorithm asymptotically unbiased for relative errors. We also show how, by the introduction of an extra weighting matrix, the algorithm can be adjusted as to become asymptotically unbiased for any given error distribution.

This paper is organized as follows. In Section 2 the continuous-time frequency domain subspace identification problem is described and the notation is introduced. Section 3 shows how a simple translation of a time domain subspace identification algorithm to the frequency domain leads to an algorithm which manipulates badly conditioned data-matrices. This problem is solved by the introduction of the Forsythe recursions in Section 4. The asymptotic analysis of the resulting algorithm is performed in Section 5. Finally, Section 6 contains an example illustrative of the concepts derived in this paper.

2. Problem description and notation

Consider the continuous-time system with m inputs, l outputs and n states:

$$\dot{x}(t) = Ax(t) + Bu(t), \tag{1}$$

$$y(t) = Cx(t) + Du(t), \tag{2}$$

with $A \in \mathbb{R}^{n\times n}$, $B \in \mathbb{R}^{n\times m}$, $C \in \mathbb{R}^{l\times n}$ and $D \in \mathbb{R}^{l\times m}$. With the assumption $x(0) = 0$, the system equations (1)–(2) can be transformed to the Laplace domain:

$$sX(s) = AX(s) + BU(s), \tag{3}$$

$$Y(s) = CX(s) + DU(s). \tag{4}$$

The frequency domain response is given by

$$H(s) = D + C(sI - A)^{-1}B. \tag{5}$$

With an input[3] $U(s) \equiv I_m$, Eqs. (3) and (4) are rewritten as

$$sX^H(s) = AX^H(s) + BI_m, \tag{6}$$

$$H(s) = CX^H(s) + DI_m. \tag{7}$$

Note that $X^H(s)$ is now an $n \times m$ matrix, where the kth column of $X^H(s)$ contains the transformed state trajectory induced by an impulse applied to the kth input.

The problem treated in this paper can now be described as follows. Given N frequency domain response samples $H(j\omega_k)$, measured at (not necessarily equidistant) frequencies ω_k, $k = 1,\ldots,N$, find the system matrices A, B, C and D.

The extended observability matrix Γ_i and the block Toeplitz matrix Θ_i are given by[4]

$$\Gamma_i \stackrel{\text{def}}{=} \begin{bmatrix} C \\ CA \\ \cdots \\ CA^{i-1} \end{bmatrix} \in \mathbb{R}^{li\times n}, \tag{8}$$

$$\Theta_i \stackrel{\text{def}}{=} \begin{bmatrix} D & 0 & \cdots & 0 \\ CB & D & \cdots & 0 \\ \cdots & \cdots & \cdots & \cdots \\ CA^{i-2}B & CA^{i-3}B & \cdots & D \end{bmatrix} \in \mathbb{R}^{li\times mi}, \tag{9}$$

with i a user defined index which is larger than the order of the system: $i > n$. The following data matrices will play a crucial role in the frequency domain subspace algorithms:[5]

$$\mathbb{H} \stackrel{\text{def}}{=} (\mathscr{R}[\mathbb{H}^c]\ \mathscr{I}[\mathbb{H}^c]) \in \mathbb{R}^{li\times 2mN}, \tag{10}$$

$$\mathbb{I} \stackrel{\text{def}}{=} (\mathscr{R}[\mathbb{I}^c]\ \mathscr{I}[\mathbb{I}^c]) \in \mathbb{R}^{mi\times 2mN}, \tag{11}$$

$$\mathbb{X} \stackrel{\text{def}}{=} (\mathscr{R}[\mathbb{X}^c]\ \mathscr{I}[\mathbb{X}^c]) \in \mathbb{R}^{n\times 2mN}, \tag{12}$$

with (note that the superscript 'c' stands for 'complex'):

$$\mathbb{H}^c \stackrel{\text{def}}{=} \begin{bmatrix} H(j\omega_1) & H(j\omega_2) & \cdots & H(j\omega_N) \\ (j\omega_1)H(j\omega_1) & (j\omega_2)H(j\omega_2) & \cdots & (j\omega_N)H(j\omega_N) \\ \cdots & \cdots & \cdots & \cdots \\ (j\omega_1)^{(i-1)}H(j\omega_1) & (j\omega_2)^{(i-1)}H(j\omega_2) & \cdots & (j\omega_N)^{(i-1)}H(j\omega_N) \end{bmatrix},$$

[3] I_s denotes the $s \times s$ unity matrix. Note that by putting $U(s) \equiv I_m$ we consider in Eqs. (6) and (7) the information obtained from m different input vectors at once.

[4] Note that Γ_i and Θ_i are real.

[5] $\mathscr{R}[\cdot]$ and $\mathscr{I}[\cdot]$ denote respectively the real and imaginary part of $\cdot$.

$$\mathbb{I}^c \stackrel{\text{def}}{=} \begin{bmatrix} I_m & I_m & \cdots & I_m \\ (j\omega_1)I_m & (j\omega_2)I_m & \cdots & (j\omega_N)I_m \\ \cdots & \cdots & \cdots & \cdots \\ (j\omega_1)^{(i-1)}I_m & (j\omega_2)^{(i-1)}I_m & \cdots & (j\omega_N)^{(i-1)}I_m \end{bmatrix},$$

$$\mathbb{X}^c \stackrel{\text{def}}{=} (X^H(j\omega_1) \quad X^H(j\omega_2) \quad \ldots \quad X^H(j\omega_N)),$$

with $\mathbb{H}^c \in \mathbb{C}^{li\times mN}$, $\mathbb{I}^c \in \mathbb{C}^{mi\times mN}$ and $\mathbb{X}^c \in \mathbb{C}^{n\times mN}$.

Conforming to the Matlab convention, we define diag(P) as follows: when $P \in \mathbb{C}^{p\times 1}$ is a vector, diag(P) $\in \mathbb{C}^{p\times p}$ denotes a diagonal matrix with the elements of P on the diagonal. However, when $P \in \mathbb{C}^{p\times p}$ is a matrix, then diag(P) $\in \mathbb{C}^{p\times 1}$ denotes a vector which contains the diagonal elements of P. vec(P) denotes the vector operation which consists of stacking the columns of P on top of each other. $\kappa(P)$ is the condition number of the matrix P which is defined as the ratio of the largest to the smallest nonzero singular value of P. P^T denotes the transpose of a real matrix P, while P^* is the complex conjugated transpose of P. Finally, $P^\dagger$ denotes the Moore–Penrose inverse of P.

The complex matrix D_ω is formed from the frequency points as follows:

$$D_\omega = \text{diag}\left[((j\omega_1)\,(j\omega_2)\,\cdots(j\omega_N)) \otimes \underbrace{(1\ 1\ \cdots\ 1)}_{\in\mathbb{R}^{1\times m}}\right],$$

where $P \otimes Q$ denotes the Kronecker product [1].

The output selector function Σ_η, $\eta = 1,\ldots,l$, is defined on *square* block matrices $P \in \mathbb{C}^{li\times li}$ with i block rows and i block columns each consisting of l rows respectively columns, by $\Sigma_\eta(P) \in \mathbb{C}^{i\times i}$ whereby the rows and columns corresponding to output η are selected. A similar definition holds for *rectangular* matrices $P \in \mathbb{C}^{li\times mN}$, where $\Sigma_\eta(P) \in \mathbb{C}^{i\times mN}$ only selects the *rows* corresponding to output η.

Π_P denotes the operator that projects the row space of a matrix onto the row space of the given matrix P (which is assumed to be of full row rank): $\Pi_P = P^T(PP^T)^{-1}P$. The projection Q/P of the row space of Q onto the row space of P is then obtained as $Q/P = Q\Pi_P = QP^T(PP^T)^{-1}P$. The orthogonal complement of the row space of P is denoted by $P^\perp$, and the projection onto the orthogonal complement of P by $\Pi_{P^\perp}$. One possible expression for it is $\Pi_{P^\perp} = (I - \Pi_P)$.

3. A simple subspace identification algorithm

In this section, we introduce a basic continuous-time frequency domain algorithm which is similar to the discrete-time time domain equivalent in [5]. We first introduce the frequency domain equivalent of the matrix input–output equations.

Lemma 1 (Input–output equation)

$$\mathbb{H}^c = \Gamma_i\mathbb{X}^c + \Theta_i\mathbb{I}^c, \tag{13}$$

$$\mathbb{H} = \Gamma_i\mathbb{X} + \Theta_i\mathbb{I}. \tag{14}$$

The proof of (13) consists of a recursive substitution of Eqs. (6) and (7) evaluated at the points $s = j\omega_1,\ldots,j\omega_N$. Formula (14) follows easily from (13) since Γ_i and Θ_i are real.

The following theorem allows for the extraction of the order n and the range of the extended observability Γ_i directly from the given data-matrices $\mathbb{H}$ and $\mathbb{I}$. The theorem is similar to the discrete-time time domain and frequency domain equivalent in [5] respectively [10].

Theorem 1 (Orthogonal projection). *With* $\mathbb{H}$, $\mathbb{I}$ *and* $\mathbb{X}$ *defined as in* (10)–(12) *and with* $\text{rank}[\mathbb{X}/\mathbb{I}^\perp] = n$ *we have*

$$\text{rank}[\mathbb{H}/\mathbb{I}^\perp] = n, \tag{15}$$

$$\text{range}[\mathbb{H}/\mathbb{I}^\perp] = \text{range}[\Gamma_i]. \tag{16}$$

This theorem can be easily proven by projecting Eq. (14) onto the orthogonal complement of $\mathbb{I}$, which leads to

$$\mathbb{H}/\mathbb{I}^\perp = \Gamma_i\mathbb{X}/\mathbb{I}^\perp + \Theta_i \underbrace{\mathbb{I}/\mathbb{I}^\perp}_{=0} = \Gamma_i[\mathbb{X}/\mathbb{I}^\perp],$$

which leads to the claims of Theorem 1. Fig. 1 gives a geometrical visualization in the $2mN$ dimensional ambient space.[6] The importance of Theorem 1 is that it allows to determine the order of the system and the

[6] Note that through the input–output equation of Theorem 1, the time domain discrete-time intersection algorithm of [14] can easily be translated to the frequency domain. We will not pursue this any further in this paper.

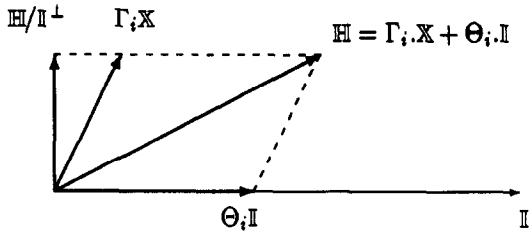

Fig. 1. Geometrical visualization of the input–output equation in the $2mN$ dimensional ambient space. The orthogonal projection $\mathbb{H}/\mathbb{I}^\perp$ determines the order n and the range of the observability matrix Γ_i.

Simple Frequency Identification Algorithm:

- Construct $\mathbb{I}$ and $\mathbb{H}$ from the given frequencies ω_k and frequency response points $H(j\omega_k)$.
- Compute $\mathbb{H}/\mathbb{I}^\perp$.
- Compute the Singular Value Decomposition:

$$\mathbb{H}/\mathbb{I}^\perp = \begin{pmatrix} U_1 & U_2 \end{pmatrix} \begin{pmatrix} S_1 & 0 \\ 0 & 0 \end{pmatrix} \begin{pmatrix} V_1^T \\ V_2^T \end{pmatrix}.$$

- Determine the order from the number of singular values S_1 different from zero.
- Determine $\mathcal{G} = U_1.S_1^{1/2}$, which is one possible estimate for the extended observability Γ_i.
- Determine A and C as:

$$C = \mathcal{G}_{\text{first } l \text{ rows}}, \quad A = [\underline{\mathcal{G}}]^\dagger.\overline{\mathcal{G}}.$$

where $\underline{\mathcal{G}}$ and $\overline{\mathcal{G}}$ denote $\mathcal{G}$ without respectively the last and first l rows.

- Determine B and D through the (least squares) solution of the linear set:

$$\begin{pmatrix} \mathcal{L}_R \\ \mathcal{L}_I \end{pmatrix} = \begin{pmatrix} \mathcal{M}_R \\ \mathcal{M}_I \end{pmatrix} \cdot \begin{pmatrix} B \\ D \end{pmatrix}, \tag{17}$$

where $\mathcal{L} \in \mathbb{C}^{lN \times m}$ and $\mathcal{M} \in \mathbb{C}^{lN \times (n+l)}$ are defined as:

$$\mathcal{L} = \mathcal{L}_R + j.\mathcal{L}_I = \begin{pmatrix} H(j\omega_1) \\ \cdots \\ H(j\omega_N) \end{pmatrix}, \tag{18}$$

$$\mathcal{M} = \mathcal{M}_R + j.\mathcal{M}_I = \begin{pmatrix} C(j\omega_1 I_n - A)^{-1} & I_l \\ \cdots & \cdots \\ C(j\omega_N I_n - A)^{-1} & I_l \end{pmatrix} \tag{19}$$

Fig. 2. Simple (straightforwardly derived) continuous-time frequency domain subspace identification algorithm.

range of the extended observability matrix Γ_i directly from the data-matrices $\mathbb{H}$ and $\mathbb{I}$. Using Theorem 1, the system matrices A, B, C, D can then be easily determined as described in the algorithm of Fig. 2 (see also [4, 16]).

This algorithm works well for small n and i. However, because of their typical 'block-Vandermonde structure' the condition number of the matrices $\mathbb{H}$ and $\mathbb{I}$ becomes extremely high when i (and thus n) grows bigger. This high condition number lies at the basis of the poor performance of the algorithm described above (see also the example in Section 6).

4. Towards a better conditioning

In this section, we derive a frequency domain subspace identification which uses only well conditioned data-matrices. Through the Forsythe recursions, we implicitly construct a well conditioned basis for the row spaces of $\mathbb{H}$ and $\mathbb{I}$.

The problem with the simple, straightforward algorithm of the previous section is that the matrices $\mathbb{H}$ and $\mathbb{I}$ become badly conditioned. If we could find matrices $L_{\mathbb{H}}$ ($L_{\mathbb{I}}$) such that $L_{\mathbb{H}}\mathbb{H}$ ($L_{\mathbb{I}}\mathbb{I}$) had a good condition number, the problem would be solved. However, these matrices $L_{\mathbb{H}}$ and $L_{\mathbb{I}}$ would be as badly conditioned as the matrices $\mathbb{H}$ and $\mathbb{I}$ themselves. We thus need to determine the products $L_{\mathbb{H}}\mathbb{H}$ and $L_{\mathbb{I}}\mathbb{I}$ without explicitly computing $L_{\mathbb{H}}$, $L_{\mathbb{I}}$, $\mathbb{H}$ and $\mathbb{I}$. The solution to this problem lies in the implicit determination of a well conditioned basis for the row space of $\mathbb{H}$ and $\mathbb{I}$ through the Forsythe recursion [15] (and references therein), without explicit computation of the matrices $\mathbb{H}$ and $\mathbb{I}$. The following definition and theorem describe how to do this.

Definition 1 (*Output Forsythe recursions*). Given the user defined index i and the frequency measurements $H(j\omega_k)$, $k = 1, \ldots, N$, the output Forsythe recursions are defined as follows:

1. Initialization:

$$R_0 = (H(j\omega_1) \ldots H(j\omega_N)),$$
$$Z_0 = \text{diag}(\text{diag}(R_0 R_0^*)), \tag{20}$$

$$R_1 = R_0 D_\omega, \quad Z_1 = \text{diag}(\text{diag}(R_1 R_1^*)). \tag{21}$$

2. Recursion: for $k = 2$ to $i - 1$

$$R_k = R_{k-1} D_\omega + Z_{k-1}[Z_{k-2}]^{-1} R_{k-2}, \tag{22}$$

$$Z_k = \text{diag}(\text{diag}(R_k R_k^*)). \tag{23}$$

This definition allows for the determination of a well conditioned basis $\mathbb{H}_F$ for the row space of $\mathbb{H}$, as is described in the following theorem.

Theorem 2 (Output basis). *With*

$$\mathbb{H}_F^c = \begin{bmatrix} Z_0^{-1/2} R_0 \\ Z_1^{-1/2} R_1 \\ \dots \\ Z_{i-1}^{-1/2} R_{i-1} \end{bmatrix}. \tag{24}$$

$$\mathbb{H}_F = (\mathscr{R}[\mathbb{H}_F^c] \quad \mathscr{I}[\mathbb{H}_F^c]),$$

we have for $\eta = 1, \dots, l$

$$\Sigma_\eta(\mathbb{H}_F)(\Sigma_\eta(\mathbb{H}_F))^T = I_i, \tag{25}$$

$$\kappa(\Sigma_\eta(\mathbb{H}_F)) = 1, \tag{26}$$

$$\text{row space}(\mathbb{H}_F) = \text{row space}(\mathbb{H}). \tag{27}$$

Theorem 2 thus states that $\mathbb{H}_F$ is an orthonormal basis for $\mathbb{H}$ for one output ($l = 1$). For more than one output, each submatrix corresponding to one output $\Sigma_\eta(\mathbb{H}_F)$, $\eta = 1, \dots, l$, is orthonormal. The condition number of $\mathbb{H}_F$ itself is hard to determine, but from simulations we found it to improve much over the conditioning of $\mathbb{H}$. A proof of Theorem 2 can be found in Appendix A.

A definition similar to Definition 1 holds for the input Forsythe recursions [7] but in this case the recursion has to be initialized with

$$R_0 = (I_m \ I_m \ \dots \ I_m).$$

Similarly as in Theorem 2, this leads to the matrix $\mathbb{I}_F$, for which we have the following theorem.

Theorem 3 (Input basis). *With a similar definition for* $\mathbb{I}_F^c$ *and* $\mathbb{I}_F$ *as the one for* $\mathbb{H}_F^c$ *and* $\mathbb{H}_F$ *in Theorem* 2, *we have*

$$\mathbb{I}_F \mathbb{I}_F^T = I_{mi}, \tag{28}$$

$$\kappa(\mathbb{I}_F) = 1, \tag{29}$$

$$\text{row space}(\mathbb{I}_F) = \text{row space}(\mathbb{I}). \tag{30}$$

[7] Note that for simplicity of notation, in the following, Z_k will *always* denote the Z_k matrices that are computed from the *output* Forsythe recursions in Definition 1, even though the *input* Forsythe recursions also compute a (different) Z_k sequence.

In Appendix A it is shown that the condition number of $\mathbb{I}_F$ is indeed equal to one, even when there is more than one input. Finally note that Eq. (28) allows for a simplification of the numerical implementation as will be illustrated below.

The Forsythe recursions thus allow the determination of a well conditioned basis $\mathbb{I}_F$ and $\mathbb{H}_F$ for the row spaces of $\mathbb{I}$ and $\mathbb{H}$, without explicitly constructing these last matrices. This is equivalent to stating that there exist matrices $L_{\mathbb{H}}$ and $L_{\mathbb{I}}$ such that

$$\mathbb{I}_F = L_{\mathbb{I}} \mathbb{I}, \tag{31}$$

$$\mathbb{H}_F = L_{\mathbb{H}} \mathbb{H}. \tag{32}$$

Note that the matrices $L_{\mathbb{I}}$ and $L_{\mathbb{H}}$ are never explicitly computed. We will see below however that they facilitate the theoretical derivations to come.

It is now easy to see that (15) and (16) can alternatively be written as

$$\text{rank}[\mathbb{H}_F / \mathbb{I}_F^\perp] = n, \tag{33}$$

$$\text{range}[\mathbb{H}_F / \mathbb{I}_F^\perp] = \text{range}[L_{\mathbb{H}} \Gamma_i], \tag{34}$$

which shows that the order of the system can still be retrieved from the well conditioned orthogonal projection $\mathbb{H}_F / \mathbb{I}_F^\perp$. The range of Γ_i is however perturbed by the linear combinations $L_{\mathbb{H}}$. This could be solved by explicitly computing $L_{\mathbb{H}}$, but this is something we want to avoid, since $L_{\mathbb{H}}$ is as badly conditioned as the matrix $\mathbb{H}$. The following theorem however shows how the matrices A and C can be computed directly from the linear combination $L_{\mathbb{H}} \Gamma_i$.

Theorem 4 (Alternative shift-structure). *Let* $\mathscr{G} \in \mathbb{R}^{li \times n}$ *be a matrix such that*

$$\text{range}(\mathscr{G}) = \text{range}(L_{\mathbb{H}} \Gamma_i),$$

then we have, with

$$D_1 = \begin{bmatrix} Z_1^{1/2} Z_2^{-1/2} & 0 & \dots & 0 \\ 0 & Z_2^{1/2} Z_3^{-1/2} & \dots & 0 \\ \dots & \dots & \dots & \dots \\ 0 & 0 & \dots & Z_{i-2}^{1/2} Z_{i-1}^{-1/2} \end{bmatrix},$$

$$D_2 = \begin{bmatrix} Z_1 (Z_0 Z_2)^{-1/2} & 0 & \dots & 0 \\ 0 & Z_2 (Z_1 Z_3)^{-1/2} & \dots & 0 \\ \dots & \dots & \dots & \dots \\ 0 & 0 & \dots & Z_{i-2} (Z_{i-3} Z_{i-1})^{-1/2} \end{bmatrix},$$

that the matrices A and C can be retrieved from $\mathcal{G}$ as

$$A = [D_1\overline{\underline{\mathcal{G}}}]^{\dagger}[\overline{\overline{\mathcal{G}}} - D_2\underline{\underline{\mathcal{G}}}], \tag{35}$$

$$C = Z_0^{1/2}\mathcal{G}_{\text{first } l \text{ rows}}, \tag{36}$$

where $\underline{\underline{\mathcal{G}}}$ denotes $\mathcal{G}$ without the last 2l rows, $\overline{\underline{\mathcal{G}}}$ denotes $\mathcal{G}$ without the l first and l last rows and $\overline{\overline{\mathcal{G}}}$ denotes $\mathcal{G}$ without the first 2l rows. [8]

A proof can be found in Appendix B. The combination of Definition 1, formulas (33) and (34) and Theorem 4, leads to the algorithm of Fig. 3, where only well conditioned data-matrices are involved. Three additional comments on the algorithm are in order:

Frequency Identification Algorithm:

- Compute $\mathbb{I}_F$ and $\mathbb{H}_F$ and the matrices $Z_0, \ldots, Z_{i-1}$ through Definition 1.
- Compute $\mathbb{H}_F/\mathbb{I}_F^{\perp} = \mathbb{H}_F - \mathbb{H}_F.\mathbb{I}_F^T.\mathbb{I}_F$.
- Compute the Singular Value Decomposition, where $W \in \mathbb{R}^{li \times li}$ is a given weighting matrix (see Section 5):

$$W.\left[\mathbb{H}_F/\mathbb{I}_F^{\perp}\right] = \begin{pmatrix} U_1 & U_2 \end{pmatrix}\begin{pmatrix} S_1 & 0 \\ 0 & 0 \end{pmatrix}\begin{pmatrix} V_1^T \\ V_2^T \end{pmatrix}.$$

- Determine the order from the number of singular values S_1 different from zero.
- Determine $\mathcal{G} = W^{-1}.U_1.S_1^{1/2}$, which is one possible estimate of $L_{\mathbb{H}}.\Gamma_i$.
- Determine A and C as:

$$A = \left[D_1.\overline{\underline{\mathcal{G}}}\right]^{\dagger}.\left[\overline{\overline{\mathcal{G}}} - D_2.\underline{\underline{\mathcal{G}}}\right],$$

$$C = Z_0^{1/2}.\mathcal{G}_{\text{first } l \text{ rows}},$$

where D_1 and D_2 are defined in Theorem 4.

- Determine B and D through the (least squares) solution of the set of linear equations:

$$W_{ls}.\begin{pmatrix} \text{vec}[\mathcal{L}_R] \\ \text{vec}[\mathcal{L}_I] \end{pmatrix} = W_{ls}.\begin{pmatrix} I_m \otimes \mathcal{M}_R \\ I_m \otimes \mathcal{M}_I \end{pmatrix}.\text{vec}\begin{pmatrix} B \\ D \end{pmatrix}. \tag{37}$$

where $\mathcal{L}$ and $\mathcal{M}$ are defined in (18)-(19) and where $W_{ls} \in \mathbb{R}^{2mlN \times 2mlN}$ is a user defined row-weight (see Section 5).

Fig. 3. Continuous-time frequency domain subspace system identification algorithm. Through the introduction of the Forsythe recursions, only well conditioned data-matrices are manipulated.

(1) Eq. (28) allows for a significant simplification of the implementation of the orthogonal projection:

$$\mathbb{H}_F/\mathbb{I}_F^{\perp} = \mathbb{H}_F - \mathbb{H}_F\mathbb{I}_F^{\mathrm{T}}\underbrace{[\mathbb{I}_F\mathbb{I}_F^{\mathrm{T}}]}_{=I_{mi}}{}^{-1}\mathbb{I}_F$$

$$= \mathbb{H}_F - \mathbb{H}_F\mathbb{I}_F^{\mathrm{T}}\mathbb{I}_F.$$

(2) The extra weighting matrices W and W_{ls} in the algorithm of Fig. 3 are for now equal to I_{li} and I_{2mlN}. They will however play an important role in the next section, where we analyze the asymptotic behavior of the algorithm.

(3) The determination of the matrices B and D in the algorithm of Fig. 3 (Eq. (37)) is slightly different compared to the algorithm of Fig. 2 (Eq. (17)). This different formulation will allow for an 'optimal' extraction of B and D, as will be described in the next section. Note that (37) can be easily derived from (17) by making use of the equality [1]

$$\text{vec}(AXB) = (B^{\mathrm{T}} \otimes A)\text{vec}(X).$$

5. Asymptotic analysis

In this section we investigate the asymptotic behavior of the algorithm of Fig. 3, when the number of frequency measurements goes to infinity $N \to \infty$ (within a fixed, finite band). For $W = I_{li}$ it is shown that the algorithm is asymptotically unbiased [9] for relative statistical errors on the measured frequency samples. On the other hand, when the given statistical distribution of the errors is not relative, it is shown how the weighting matrix W can be chosen appropriately to make the algorithm of Fig. 3 asymptotically unbiased.

In the previous sections, we assumed the given data $H(j\omega_k)$ to be noise-free. In this section, we will investigate what happens, when this data is corrupted by zero mean additive noise:

$$\hat{H}(j\omega_k) = H(j\omega_k) + N(j\omega_k),$$

[8] Note that additionally the estimate of C in (36) could be scaled with a scalar, for instance the first diagonal element of $Z_0^{1/2}$.

[9] With asymptotically unbiased we mean that the identified system converges with probability 1 to the true system when $N \to \infty$.

with the noise $N(j\omega_k) \in \mathbb{C}^{l\times m}$ equal to

$$N(j\omega_k) = N_R(j\omega_k) + jN_I(j\omega_k)$$

$$= \begin{bmatrix} N_R^{11}(j\omega_k) & \cdots & N_R^{1m}(j\omega_k) \\ \cdots & \cdots & \cdots \\ N_R^{l1}(j\omega_k) & \cdots & N_R^{lm}(j\omega_k) \end{bmatrix}$$

$$+j\begin{bmatrix} N_I^{11}(j\omega_k) & \cdots & N_I^{1m}(j\omega_k) \\ \cdots & \cdots & \cdots \\ N_I^{l1}(j\omega_k) & \cdots & N_I^{lm}(j\omega_k) \end{bmatrix}, \quad (38)$$

where the subscripts 'R' and 'I' denote the real respectively imaginary part. We assume that the noise is uncorrelated between the different inputs and outputs and also between different frequency points ω_k. These conditions can be expressed as [10]

$$\boldsymbol{E}[N_R^{\eta_r\rho_s}(j\omega_r)N_R^{\eta_s\rho_s}(j\omega_s)]$$

$$= [\sigma_R^{\eta_r\rho_r}(j\omega_r)]^2\delta_{rs}\delta_{\eta_r\eta_s}\delta_{\rho_r\rho_s}, \quad (39)$$

$$\boldsymbol{E}[N_I^{\eta_r\rho_r}(j\omega_r)N_I^{\eta_s\rho_s}(j\omega_s)]$$

$$= [\sigma_I^{\eta_r\rho_r}(j\omega_r)]^2\delta_{rs}\delta_{\eta_r\eta_s}\delta_{\rho_r\rho_s}, \quad (40)$$

$$\boldsymbol{E}[N_R^{\eta_r\rho_r}(j\omega_r)N_I^{\eta_s\rho_s}(j\omega_s)] = 0, \quad (41)$$

where $\sigma_R^{\eta\rho}(j\omega_r)$ and $\sigma_I^{\eta\rho}(j\omega_r)$ are given standard deviations for $k = 1,\ldots,N$, $\eta = 1,\ldots,l$ and $\rho = 1,\ldots,m$. Due to the infinite number of data at our disposition, and with

$$\boldsymbol{E}_{\mathbb{N}}[\cdot] = \lim_{N\to\infty}\left[\frac{1}{N}[\cdot]\right],$$

we get the following properties (similar properties hold for $N_I^{\eta\rho}(j\omega_k)$):

$$\boldsymbol{E}_{\mathbb{N}}\left[\sum_{k=1}^{N} N_R^{\eta\rho}(j\omega_k)\right] = 0, \quad (42)$$

$$\boldsymbol{E}_{\mathbb{N}}\left[\sum_{k=1}^{N}[N_R^{\eta\rho}(j\omega_k)]^2\right] = \boldsymbol{E}_{\mathbb{N}}\left[\sum_{k=1}^{N}[\sigma_R^{\eta\rho}(j\omega_k)]^2\right] \quad (43)$$

[10] $\boldsymbol{E}$ denotes the expected value operator and δ_{rs} the Kronecker delta.

By making use of (38)–(41) and the central limit theorem one can indeed prove that the left-hand side of (42) is a Gaussian random variable with expected value and standard deviation equal to zero (see also [6]). Similarly the left-hand side of (43) has an expected value equal to the right-hand side of (43) while its standard deviation is equal to zero (if the stochastic signals have finite fourth-order moments [3]). That is why, when $N \to \infty$, Eqs. (42) and (43) hold with probability 1.

In the following theorem we analyze the asymptotic behavior of the algorithm in Fig. 3. In the theorem, an auxillary signal $S(j\omega_k) \in \mathbb{C}^{l\times m}$ is introduced to allow for the computation of the weighting matrix W. As will be shown in Appendix C, this signal $S(j\omega)$ is chosen in such a way that (λ is a positive scalar)

$$\boldsymbol{E}_{\mathbb{N}}[\Sigma_\eta(\mathbb{N})(\Sigma_\eta(\mathbb{N}))^T]$$

$$= \lambda\boldsymbol{E}_{\mathbb{N}}[\Sigma_\eta(\mathbb{S})(\Sigma_\eta(\mathbb{S})^T], \quad \eta = 1,\ldots,l,$$

where the data matrices $\mathbb{N}$ and $\mathbb{S}$ are constructed from $N(j\omega_k)$ respectively $S(j\omega_k)$ in the same way as $\mathbb{H}$ was constructed from $H(j\omega_k)$ in Eq. (10). The next theorem states how this auxillary signal should be chosen and how the weight W can be derived from it.

Theorem 5 (Asymptotic properties)

- *With* $\hat{H}(j\omega_k) = H(j\omega_k) + N(j\omega_k)$ *where the zero mean noise sequence* $N(j\omega_k)$ *has the properties* (38)–(41),
- *With the auxillary signal* $S(j\omega_k) \in \mathbb{C}^{l\times m}$ *satisfying the following relation*:

$$\lambda|S^{\eta\rho}(j\omega_k)|^2 = [\sigma_R^{\eta\rho}(j\omega_k)]^2 + [\sigma_I^{\eta\rho}(j\omega_k)]^2, \quad (44)$$

where λ *is a positive scalar,*

- *with the matrix* $\mathbb{S}$ *constructed from the auxilary signal* $S(j\omega_k)$ *in a similar way as* $\mathbb{H}$ *was constructed from* $H(j\omega_k)$ *in Eq.* (10); *and with the matrix*

$$\mathbb{S}_F = L_{\mathbb{H}}\mathbb{S}, \quad (45)$$

where $L_{\mathbb{H}}$ *was implicitly defined in Eq.* (32),

The algorithm of Fig. 3 is asymptotically unbiased ($N \to \infty$) when the weighting matrix W is defined as [11]

$$\Sigma_\eta[W] \stackrel{\text{def}}{=} \left[\Sigma_\eta(\mathbb{S}_F)(\Sigma_\eta(\mathbb{S}_F))^{\mathrm{T}}\right]^{-1/2}, \quad \eta = 1,\ldots,l, \tag{46}$$

and equal to zero where it is not explicitly defined [12] *by* (46).

A proof of the theorem can be found in Appendix C.

Discussion

- Since the matrix $\mathbb{S}_F$ is constructed from $\mathbb{S}$ by taking exactly the same linear combinations $L_{\mathbb{H}}$ as for the construction of $\mathbb{H}_F$ out of $\mathbb{H}$ (see formula (45)), $\mathbb{S}_F$ can be computed by applying the output Forsythe recursions to $S(\mathrm{j}\omega_k)$.
- When the noise $N(\mathrm{j}\omega_k)$ is relative (with σ_{R} and σ_{I} positive scalars),

$$\sigma_{\mathrm{R}}^{\eta\rho}(\mathrm{j}\omega_k) = |H^{\eta\rho}(\mathrm{j}\omega_k)|\sigma_{\mathrm{R}}, \quad \sigma_{\mathrm{I}}^{\eta\rho} = |H^{\eta\rho}(\mathrm{j}\omega_k)|\sigma_{\mathrm{I}},$$

we find that (44) is satisfied for $S(\mathrm{j}\omega_k) = H(\mathrm{j}\omega_k)$ since

$$\begin{aligned}&[\sigma_{\mathrm{R}}^{\eta\rho}(\mathrm{j}\omega_k)]^2 + [\sigma_{\mathrm{I}}^{\eta\rho}(\mathrm{j}\omega_k)]^2\\ &\quad = |H^{\eta\rho}(\mathrm{j}\omega_k)|^2(\sigma_{\mathrm{R}}^2 + \sigma_{\mathrm{I}}^2)\\ &\quad = \lambda|H^{\eta\rho}(\mathrm{j}\omega_k)|^2.\end{aligned}$$

This implies that $\mathbb{S}_F$ can be taken equal to $\mathbb{H}_F$. Moreover, through the construction of $\mathbb{H}_F$ and Theorem 2 (25), we find that

$$\Sigma_\eta(\mathbb{S}_F)(\Sigma_\eta(\mathbb{S}_F))^{\mathrm{T}} = \Sigma_\eta(\mathbb{H}_F)(\Sigma_\eta(\mathbb{H}_F))^{\mathrm{T}} = I_i.$$

From this last equation and from (46), we conclude that the weighting matrix $W = I_{li}$. This thus implies that when the weighting matrix $W = I_{li}$ in the algorithm of Fig. 3, the algorithm is asymptotically unbiased for *relative* noise perturbations. It is quite surprising to note that the Forsythe recursions not only drastically improve the condition numbers of $\mathbb{H}$ and $\mathbb{I}$ but that they also make the algorithm asymptotically unbiased for relative noise perturbations on the frequency samples.
- Note that when the measurements are noisy, the singular value decomposition of the algorithm in Fig. 3 becomes ($N \to \infty$)

$$W[\mathbb{H}_F/\mathbb{I}_F^{\perp}] = (U_1\ U_2)\begin{bmatrix} S_1 + \sqrt{\lambda}I_n & 0 \\ 0 & \sqrt{\lambda}I_{li-n}\end{bmatrix}\begin{bmatrix} V_1^{\mathrm{T}} \\ V_2^{\mathrm{T}}\end{bmatrix},$$

where the smallest singular values are not zero any more but equal to $\sqrt{\lambda}$.
- Since the standard deviation of the noise on each of the measurements is given (38), the optimal [13] row-weight W_{ls} in the set of least squares equations in B and D, (37) can be easily computed as

$$W_{\mathrm{ls}} = \left[\mathrm{diag}\begin{bmatrix}\mathrm{vec}[\mathscr{W}_{\mathrm{R}}] \\ \mathrm{vec}[\mathscr{W}_{\mathrm{I}}]\end{bmatrix}\right]^{-1},$$

[11] $B = A^{-1/2}$ implies that $B^{\mathrm{T}}B = A^{-1}$. This square root can be computed by for instance the Cholesky factorization.

[12] Note that due to the introduction of the selector matrix Σ_η, W is only defined by (46) at the elements W^{ij} for which the row index i and column index j satisfies $i = j \pm \zeta l$, where ζ is a positive integer.

[13] Optimal in the sense that the solution computed with this weighting has a variance matrix smaller (in the sense of positive definiteness) than the solution computed with any other row-weighting matrix. It should be noted that this weight is optimal only for the solution of the least squares problem (37) and does not imply optimality for the overall identification procedure. The weighting is also only optimal when A and C are exactly known. This is because the perturbations introduced by estimated A and C on the left-hand side of the least squares problem could be correlated with the perturbations on the right-hand side.

where

$$\mathscr{W}_{\mathrm{R}} = \begin{bmatrix} \sigma_{\mathrm{R}}^{11}(\mathrm{j}\omega_1) & \cdots & \sigma_{\mathrm{R}}^{1m}(\mathrm{j}\omega_1) \\ \cdots & \cdots & \cdots \\ \sigma_{\mathrm{R}}^{l1}(\mathrm{j}\omega_1) & \cdots & \sigma_{\mathrm{R}}^{lm}(\mathrm{j}\omega_1) \\ \cdots & \cdots & \cdots \\ \sigma_{\mathrm{R}}^{11}(\mathrm{j}\omega_N) & \cdots & \sigma_{\mathrm{R}}^{1m}(\mathrm{j}\omega_N) \\ \cdots & \cdots & \cdots \\ \sigma_{\mathrm{R}}^{l1}(\mathrm{j}\omega_N) & \cdots & \sigma_{\mathrm{R}}^{lm}(\mathrm{j}\omega_N) \end{bmatrix},$$

$$\mathscr{W}_{\mathrm{I}} = \begin{bmatrix} \sigma_{\mathrm{I}}^{11}(\mathrm{j}\omega_1) & \cdots & \sigma_{\mathrm{I}}^{1m}(\mathrm{j}\omega_1) \\ \cdots & \cdots & \cdots \\ \sigma_{\mathrm{I}}^{l1}(\mathrm{j}\omega_1) & \cdots & \sigma_{\mathrm{I}}^{lm}(\mathrm{j}\omega_1) \\ \cdots & \cdots & \cdots \\ \sigma_{\mathrm{I}}^{11}(\mathrm{j}\omega_N) & \cdots & \sigma_{\mathrm{I}}^{1m}(\mathrm{j}\omega_N) \\ \cdots & \cdots & \cdots \\ \sigma_{\mathrm{I}}^{l1}(\mathrm{j}\omega_N) & \cdots & \sigma_{\mathrm{I}}^{lm}(\mathrm{j}\omega_N) \end{bmatrix}.$$

It should be noted that the asymptotic analysis of this section is based on the assumption of a known noise spectrum in contrast to the method of [12] which is shown consistent [13] also for the case of an unknown noise spectrum.

6. Example

In this section we consider a simple example to illustrate the algorithm of Fig. 3. We consider the following sixth-order system:

$$A = \begin{bmatrix} 0 & 1 & 0 & 0 & 0 & 0 \\ -1 & -0.2 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & -25 & -0.5 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & -9 & -0.12 \end{bmatrix},$$

$$B = \begin{bmatrix} 0 \\ 1 \\ 0 \\ 1 \\ 0 \\ 1 \end{bmatrix}, \qquad C = \begin{bmatrix} 1 \\ 0 \\ 1 \\ 0 \\ 1 \\ 0 \end{bmatrix}^{\mathrm{T}},$$

and $D = 0$. The frequency axis ω_k goes from 0.01 up to 9 rad/s, with a spacing of 0.05 rad/s between two frequency points ($N = 180$). The number of block rows i is taken equal to 15. We ran two different series of Monte Carlo simulations in Matlab, each consisting of 100 runs:

- In a first Monte Carlo series the measurement points were perturbed by *relative* zero mean, white Gaussian noise with a standard deviation equal to 15% of the magnitude:

$$\hat{H}(\mathrm{j}\omega_k) = H(\mathrm{j}\omega_k) + 0.15|H(\mathrm{j}\omega_k)|[e_{\mathrm{R}}(\mathrm{j}\omega_k) + \mathrm{j}e_{\mathrm{I}}(\mathrm{j}\omega_k)],$$

  where $e_{\mathrm{R}}(\mathrm{j}\omega_k)$ and $e_{\mathrm{I}}(\mathrm{j}\omega_k)$) are unit variance, zero mean, white Gaussian noise sequences. The identification (Fig.3) was performed once with the weight $W = I$ (relative errors) and once with the weight W corresponding to absolute errors.
- In a second Monte Carlo series the measurement points were perturbed by *absolute* white Gaussian noise with a standard deviation equal to 0.03:

$$\hat{H}(\mathrm{j}\omega_k) = H(\mathrm{j}\omega_k) + 0.03[e_{\mathrm{R}}(\mathrm{j}\omega_k) + \mathrm{j}e_{\mathrm{I}}(\mathrm{j}\omega_k)],$$

  where $e_{\mathrm{R}}(\mathrm{j}\omega_k)$ and $e_{\mathrm{I}}(\mathrm{j}\omega_k)$) are unit variance, zero mean, white Gaussian noise sequences. The identification (Fig.3) was performed once with the weight $W = I$ (relative errors) and once with the weight W corresponding to absolute errors.

Figs. 4 and 5 illustrate the results. Fig. 4 shows the average (over the 100 experiments) singular value spectrum of the orthogonal projection. Fig. 5 shows the exact and average transfer function. This example clearly illustrates the importance of the right weighting function W in the algorithm of Fig. 3. If this weighting matrix is chosen appropriately, the results are unbiased. However, a wrong weighting function leads to biased results. Finally note that typical condition numbers for this example were

$$\kappa(\mathbb{H}) = 10^{16}, \qquad \kappa(\mathbb{I}) = 10^{16},$$
$$\kappa(\mathbb{H}_F) = 1, \qquad \kappa(\mathbb{I}_F) = 1.$$

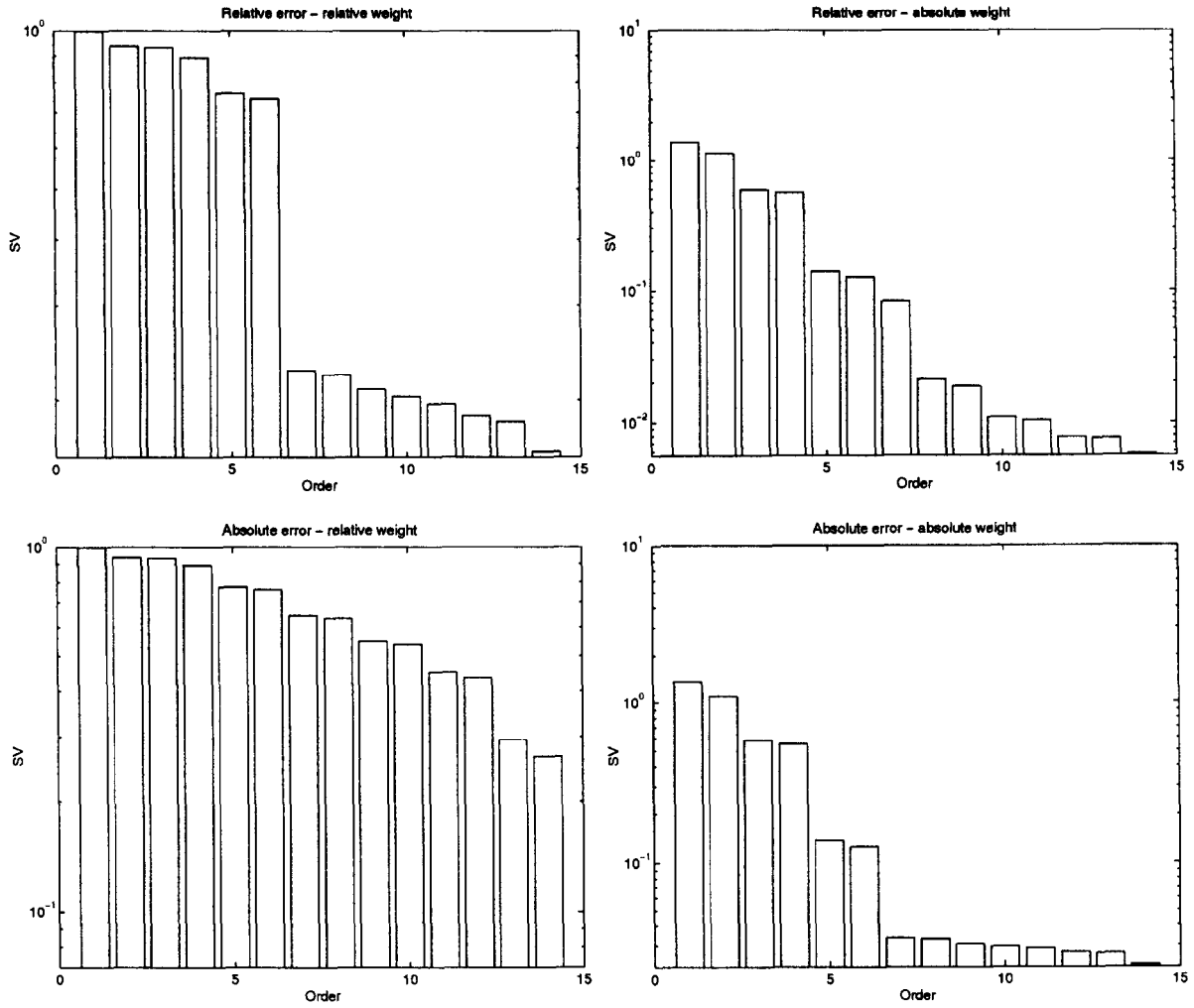

Fig. 4. Average singular value spectrum. (a) Relative errors and $W = I$. (b) Relative errors and W corresponding to absolute errors. (c) Absolute errors and $W = I$. (d) Absolute errors and W corresponding to absolute errors. This figure illustrates the importance of choosing the right weighting. In (a) and (d), the order can be clearly seen to be equal to six. In (b) and (c) however (where the weighting W did not match the noise distribution), the order is not clear.

The solution computed by the algorithm of Fig. 2 (which uses the badly conditioned matrices $\mathbb{H}$ and I) was completely different from the exact solution. This is due to the high condition numbers listed above.

7. Conclusions

In this paper we derived a new subspace identification algorithm for the identification of continuous-time systems from frequency response data. The conditioning of the data-matrices was drastically improved through the introduction of the Forsythe recursions. An asymptotic analysis of the algorithm proved that these recursions also assure an asymptotically unbiased solution for relative errors. An additional weighting matrix makes the algorithm asymptotically unbiased for other given error distributions.

Acknowledgements

We would like to thank Thomas McKelvey for generously sharing his ideas.

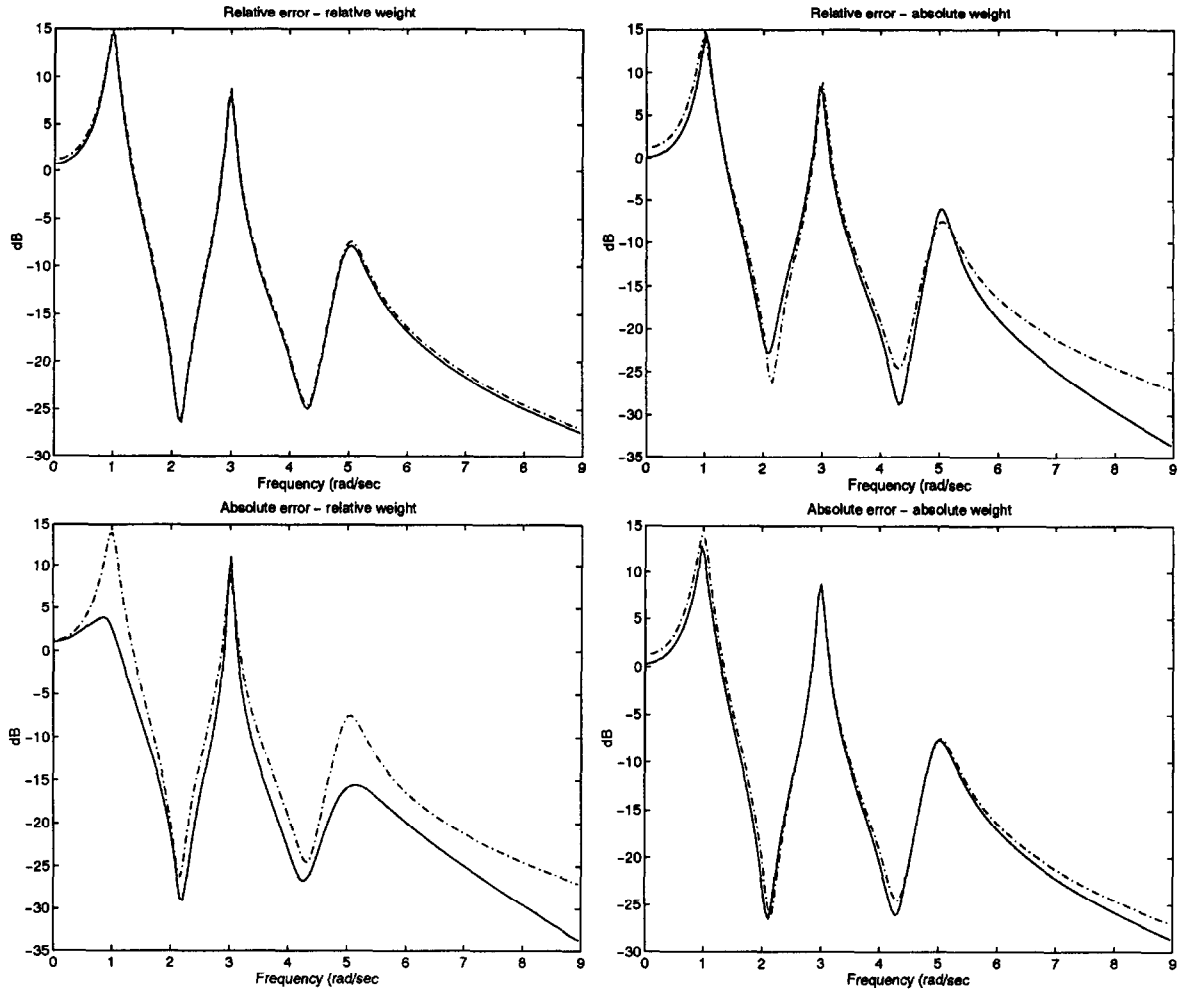

Fig. 5. Average estimated (full line) and exact (dashed line) transfer function. (a) Relative errors and $W = I$. (b) Relative errors and W corresponding to absolute errors. (c) Absolute errors and $W = I$. (d) Absolute errors and W corresponding to absolute errors. This figure illustrates the importance of choosing the right weighting. In (a) and (d), the transfer function is estimated well, while in (b) and (c), where the weighting matrix W does not match the noise conditions, there is a clear bias.

Appendix A. Proof of the Forsythe recursion

We prove that, for one output, Eq. (25) holds. Once (25) is proven, Eq. (26) follows directly. Eq. (27) follows directly from Definition 1.

We need to prove that (for one output)

$$\mathbb{H}_F(\mathbb{H}_F)^{\mathrm{T}} = \mathscr{R}[\mathbb{H}_F^{\mathrm{c}}(\mathbb{H}_F^{\mathrm{c}})^*] = I_i.$$

Since (for one output)

$$\mathbb{H}_F = \begin{pmatrix} R_0/\sqrt{Z_0} \\ \cdots \\ R_{i-1}/\sqrt{Z_{i-1}} \end{pmatrix},$$

we have to prove that

$$\frac{\mathscr{R}[R_r R_s^*]}{\sqrt{Z_r Z_s}} = \delta_{rs}, \quad 0 \leqslant r, s \leqslant i-1.$$

The proof for $r = s$ is trivial, since it follows from the definition of Z_r that $(R_r R_r^*)/Z_r = 1$. The proof for $r \neq s$ is by induction. [14]

[14] In the following R_k^v denotes the vth column of R_k.

A.1. Initialization: $\mathscr{R}[R_r R_0] = 0$ *for* $r = 1,\dots,i-1$:

$r = 1$:

$$\begin{aligned}\mathscr{R}[R_1 R_0^*] &= \mathscr{R}[R_0 D_\omega R_0^*] \\ &= \mathscr{R}\left[\mathrm{j}\sum_{v=1}^{mN} \omega_k |R_0^v|^2\right] \\ &= 0.\end{aligned}$$

$r = 2$:

$$\begin{aligned}\mathscr{R}[R_2 R_0^*] &= \mathscr{R}\left[R_1 D_\omega R_0^* + \frac{R_1 R_1^*}{R_0 R_0^*}(R_0 R_0^*)\right] \\ &= \mathscr{R}[-R_1 (R_0 D_\omega)^* + R_1 R_1^*] \\ &= \mathscr{R}[-R_1 R_1^* + R_1 R_1^*] \\ &= 0.\end{aligned}$$

Since R_r can be written as a linear combination of R_{r-1} and R_{r-2}, it is easy to see that $\mathscr{R}[R_r R_0^*]$ will be zero for $3 \leqslant r \leqslant i-1$.

A.2. Induction

Say it is given that $\mathscr{R}[R_r R_s^*] = 0$ for $s = 0,\dots,k-1$ and $r = s+1,\dots,i-1$. We now prove that $\mathscr{R}[R_r R_k^*] = 0$ for $r = k+1,\dots,i-1$.
$r = k+1$:

$$\begin{aligned}\mathscr{R}[R_{k+1} R_k^*] &= \mathscr{R}\left[R_k D_\omega R_k^* + \frac{Z_k}{Z_{k-1}} \underbrace{R_{k-1} R_k^*}_{=0 \text{ (given)}}\right] \\ &= \mathscr{R}\left[\mathrm{j}\sum_{v=1}^{mN} \omega_k |R_k^v|^2\right] \\ &= 0.\end{aligned}$$

$r = k+2$:

$$\begin{aligned}&\mathscr{R}[R_{k+2} R_k^*] \\ &= \mathscr{R}\left[R_{k+1} D_\omega R_k^* + \frac{R_{k+1} R_{k+1}^*}{R_k R_k^*}(R_k R_k^*)\right] \\ &= \mathscr{R}[-R_{k+1}(R_k D_\omega)^* + R_{k+1} R_{k+1}^*] \\ &= \mathscr{R}\left[-R_{k+1}\left(R_{k+1} - \frac{Z_{k-1}}{Z_{k-2}} R_{k-1}\right)^* + R_{k+1} R_{k+1}^*\right] \\ &= \mathscr{R}\left[\frac{Z_{k-1}}{Z_{k-2}} \underbrace{(R_{k+1} R_{k-1}^*)}_{=0 \text{ given}}\right] \\ &= 0.\end{aligned}$$

Since R_r can be written as a linear combination of R_{r-1} and R_{r-2}, it is easy to see that $\mathscr{R}[R_r R_k^*]$ will be zero for $k+3 \leqslant r \leqslant i-1$.

For more than one output, we find that

$$Z_0^{-1/2} R_0$$

and thus that $\mathbb{H}_F$ is not orthonormal. However, it is easy to prove that the matrix $\Sigma_\eta(\mathbb{H}_F)$ is an orthonormal matrix for each $\eta = 1,\dots,l$. This explains why we found good condition numbers for $\mathbb{H}_F$, even for more than one output.

Finally, independently of the number of inputs, we have

$$\mathbb{I}_F \mathbb{I}_F^{\mathrm{T}} = I_{mi},$$

since the rows of $(I_m \ \dots \ I_m)$ are orthonormal, which proves (28) and (29). □

Appendix B. Proof of the alternative shift structure

We denote with h_k and h_k^F respectively the kth block row of $\mathbb{H}^c$ and $\mathbb{H}_F^c = L_{\mathbb{H}} \mathbb{H}^c$. Similarly, we denote with γ_k and γ_k^F the kth block row of Γ_i and $L_{\mathbb{H}} \Gamma_i$. The following equations are now easy to derive:

$$h_k = h_{k-1} D_\omega, \tag{B.1}$$

$$\gamma_k = \gamma_{k-1} A, \tag{B.2}$$

$$\begin{aligned}h_k^F &= Z_{k-1}^{1/2} Z_k^{-1/2} h_{k-1}^F D_\omega \\ &\quad + Z_{k-1}(Z_{k-2} Z_k)^{-1/2} h_{k-2}^F,\end{aligned} \tag{B.3}$$

$$h_k^F = \sum_{v=0}^{k} \alpha_v h_v, \tag{B.4}$$

$$\gamma_k^F = \sum_{v=0}^{k} \alpha_v \gamma_v. \tag{B.5}$$

Eqs. (B.1) and (B.2) are derived from the definitions of $\mathbb{H}^c$ and Γ_i. Eq. (B.3) follows from (22), while

Eqs. (B.4) and (B.5) follow from the fact that the kth block row of $\mathbb{H}_F^c$ and $L_{\mathbb{H}}\Gamma_i$ can be written as the same linear combination of the block rows of $\mathbb{H}^c$ (Γ_i). Using (B.1) through (B.5), we find

$$\begin{aligned} h_k^F &= Z_{k-1}^{1/2} Z_k^{-1/2} h_{k-1}^F D_\omega + Z_{k-1}(Z_{k-2}Z_k)^{-1/2} h_{k-2}^F \\ &= Z_{k-1}^{1/2} Z_k^{-1/2} \left(\sum_{v=0}^{k-1} \alpha_v h_v D_\omega \right) \\ &\quad + Z_{k-1}(Z_{k-2}Z_k)^{-1/2} h_{k-2}^F \\ &= Z_{k-1}^{1/2} Z_k^{-1/2} \left(\sum_{v=0}^{k-1} \alpha_v h_{v+1} \right) \\ &\quad + Z_{k-1}(Z_{k-2}Z_k)^{-1/2} h_{k-2}^F. \end{aligned}$$

This last equation also states how γ_k^F can be written as a linear combination of the block rows of Γ_i and of γ_{k-2}^F:

$$\begin{aligned} \gamma_k^F &= Z_{k-1}^{1/2} Z_k^{-1/2} \left(\sum_{v=0}^{k-1} \alpha_v \gamma_{v+1} \right) \\ &\quad + Z_{k-1}(Z_{k-2}Z_k)^{-1/2} \gamma_{k-2}^F \\ &= Z_{k-1}^{1/2} Z_k^{-1/2} \left(\sum_{v=0}^{k-1} \alpha_v \gamma_v \right) A \\ &\quad + Z_{k-1}(Z_{k-2}Z_k)^{-1/2} \gamma_{k-2}^F \\ &= Z_{k-1}^{1/2} Z_k^{-1/2} \gamma_{k-1}^F A + Z_{k-1}(Z_{k-2}Z_k)^{-1/2} \gamma_{k-2}^F. \end{aligned}$$

Eq. (35) can be easily derived from this last equation. Eq. (36) follows trivially from

$$\gamma_0^F = Z_0^{-1/2}\gamma_0 = Z_0^{-1/2} C. \qquad \square$$

Appendix C. Proof of the asymptotic properties

The input–output equation (14) can be rewritten with the extra noise source $N(j\omega_k)$ as

$$\mathbb{H} = \Gamma_i \mathbb{X} + \Theta_i \mathbb{I} + \mathbb{N},$$

where $\mathbb{N}$ is constructed from $N(j\omega_k)$ in the same way as $\mathbb{H}$ and $\mathbb{I}$. This leads to

$$W\mathbb{H}_F/\mathbb{I}_F^\perp = WL_{\mathbb{H}}\Gamma_i \mathbb{X}/\mathbb{I}_F^\perp + WL_{\mathbb{H}}\mathbb{N}/\mathbb{I}_F^\perp.$$

For the algorithm to be consistent ($N \to \infty$), we need [2]

$$\begin{aligned} &\underbrace{\lim_{N\to\infty} [L_{\mathbb{H}}\mathbb{N}/\mathbb{I}_F^\perp (\mathbb{N}/\mathbb{I}_F^\perp)^{\mathrm{T}} L_{\mathbb{H}}^{\mathrm{T}}]}_{=\mathscr{A}} \\ &= \lambda \underbrace{\lim_{N\to\infty} [(W^{\mathrm{T}}W)^{-1}]}_{=\mathscr{B}}. \end{aligned} \tag{C.1}$$

We now consider the terms $\mathscr{A}$ and $\mathscr{B}$ separately. The term $\mathscr{A}$ in (C.1) can be rewritten as

$$\lim_{N\to\infty} \left(\sqrt{N} L_{\mathbb{H}} \left[\frac{\mathbb{N}/\mathbb{I}_F^\perp (\mathbb{N}/\mathbb{I}_F^\perp)^{\mathrm{T}}}{N} \right] L_{\mathbb{H}}^{\mathrm{T}} \sqrt{N} \right). \tag{C.2}$$

Since $\sqrt{N}L_{\mathbb{H}}$ is finite,[15] we can rewrite (C.2) as

$$\lim_{N\to\infty} (\sqrt{N}L_{\mathbb{H}}) \boldsymbol{E}_{\mathbb{N}} [\mathbb{N}/\mathbb{I}_F^\perp (\mathbb{N}/\mathbb{I}_F^\perp)^{\mathrm{T}}] \lim_{N\to\infty} (L_{\mathbb{H}}^{\mathrm{T}}\sqrt{N}). \tag{C.3}$$

From Eq. (42) we find that $\boldsymbol{E}_{\mathbb{N}}[\mathbb{N}/\mathbb{I}^{\mathrm{F}}] = 0$ (see also [16]) which means that (C.3) can be rewritten as

$$\lim_{N\to\infty} (\sqrt{N}L_{\mathbb{H}}) \boldsymbol{E}_{\mathbb{N}} [\mathbb{N}\mathbb{N}^{\mathrm{T}}] \lim_{N\to\infty} (L_{\mathbb{H}}^{\mathrm{T}}\sqrt{N}). \tag{C.4}$$

Since the noise $N(j\omega_k)$ is uncorrelated between different outputs (39)–(41), the only elements in formula (C.4) different from zero are the elements selected by the selector matrices Σ_η, $\eta = 1, \ldots, l$. Using (43) and (C.4) it is now tedious though straightforward to prove that

$$\begin{aligned} &\Sigma_\eta(\mathscr{A}) \\ &= \lim_{N\to\infty} (\sqrt{N}\Sigma_\eta(L_{\mathbb{H}})) \boldsymbol{E}_{\mathbb{N}} [\Sigma_\eta(\mathbb{N})\Sigma_\eta(\mathbb{N})^{\mathrm{T}}] \\ &\quad \times \lim_{N\to\infty} (\Sigma_\eta(L_{\mathbb{H}}^{\mathrm{T}})\sqrt{N}) \\ &= \lim_{N\to\infty} (\sqrt{N}\Sigma_\eta(L_{\mathbb{H}})) \\ &\quad \times \begin{bmatrix} \mathscr{N}_0 & 0 & -\mathscr{N}_2 & \cdots \\ 0 & \mathscr{N}_2 & 0 & \cdots \\ -\mathscr{N}_2 & 0 & \mathscr{N}_4 & \cdots \\ \cdots & \cdots & \cdots & \cdots \end{bmatrix} \\ &\quad \times \lim_{N\to\infty} (\Sigma_\eta(L_{\mathbb{H}}^{\mathrm{T}})\sqrt{N}), \end{aligned} \tag{C.5}$$

[15] Note that due to the scaling $Z_k^{-1/2}$ of the block rows in formula (24) we have that $\lim_{N\to\infty} L_{\mathbb{H}} = 0$ while $\lim_{N\to\infty} \sqrt{N}L_{\mathbb{H}}$ is finite.

with (using (43))

$$\mathscr{N}_p = \boldsymbol{E}_{\mathbb{N}}\left[\sum_{k=1}^{N}\sum_{\rho=1}^{m}\omega_k^p[(N_{\mathrm{R}}^{\eta\rho}(\mathrm{j}\omega_k))^2+(N_{\mathrm{I}}^{\eta\rho}(\mathrm{j}\omega_k))^2]\right]$$

$$= \boldsymbol{E}_{\mathbb{N}}\left[\sum_{k=1}^{N}\sum_{\rho=1}^{m}\omega_k^p[(\sigma_{\mathrm{R}}^{\eta\rho}(\mathrm{j}\omega_k))^2+(\sigma_{\mathrm{I}}^{\eta\rho}(\mathrm{j}\omega_k))^2]\right]. \tag{C.6}$$

We now consider $\mathscr{B}$, the right-hand side of formula (C.1). Since both sides of (C.1) are only different from zero for elements selected by the selector matrices $\Sigma_\eta, \eta = 1,\dots,l$, we only consider $\Sigma_\eta(\mathscr{B})$,

$$\begin{aligned}\Sigma_\eta(\mathscr{B}) &= \lim_{N\to\infty}\Sigma_\eta[(W^{\mathrm{T}}W)^{-1}]\\ &= \lim_{N\to\infty}\Sigma_\eta(\mathbb{S}_F)\Sigma_\eta(\mathbb{S}_F)^{\mathrm{T}}\\ &= \lim_{N\to\infty}(\sqrt{N}\Sigma_\eta(L_{\mathbb{H}}))\boldsymbol{E}_{\mathbb{N}}[\Sigma_\eta(\mathbb{S})\Sigma_\eta(\mathbb{S})^{\mathrm{T}}]\\ &\quad\times\lim_{N\to\infty}(\Sigma_\eta(L_{\mathbb{H}}^{\mathrm{T}})\sqrt{N}).\end{aligned}$$

Once again, it is tedious though straightforward to prove that

$$\begin{aligned}\Sigma_\eta(\mathscr{B}) &= \lim_{N\to\infty}(\sqrt{N}\Sigma_\eta(L_{\mathbb{H}}))\\ &\quad\times\begin{bmatrix}\mathscr{P}_0 & 0 & -\mathscr{P}_2 & \cdots\\ 0 & \mathscr{P}_2 & 0 & \cdots\\ -\mathscr{P}_2 & 0 & \mathscr{P}_4 & \cdots\\ \cdots & \cdots & \cdots & \cdots\end{bmatrix}\\ &\quad\times\lim_{N\to\infty}(\Sigma_\eta(L_{\mathbb{H}}^{\mathrm{T}})\sqrt{N}),\end{aligned} \tag{C.7}$$

with

$$\mathscr{P}_p = \boldsymbol{E}_{\mathbb{N}}\left[\sum_{k=1}^{N}\sum_{\rho=1}^{m}\omega_k^p|S^{\eta\rho}(\mathrm{j}\omega_k)|^2\right]. \tag{C.8}$$

Comparing (C.5) and (C.7) and the expressions (C.6) and (C.8) we see that the necessary condition for (C.1) to be satisfied is that

$$\mathscr{N}_p = \lambda\mathscr{P}_p,$$

which can be rewritten as

$$\lambda|S^{\eta\rho}(\mathrm{j}\omega_k)|^2 = [\sigma_{\mathrm{R}}^{\eta\rho}(\mathrm{j}\omega_k)]^2 + [\sigma_{\mathrm{I}}^{\eta\rho}(\mathrm{j}\omega_k)]^2.$$

This is exactly Eq. (44) in Theorem 5.

References

[1] J.W. Brewer, "Kronecker products and matrix calculus in system theory", *IEEE Trans. Circuits Systems*, Vol. CAS-25, No. 9, 1978, pp. 772–780.

[2] B. De Moor, Mathematical concepts and techniques for modeling of static and dynamic systems, Ph.D. Thesis, Department of Electrical Engineering, Katholieke Universiteit Leuven, Belgium, 1988.

[3] B. De Moor, "The singular value decomposition and long and short spaces of noisy matrices", *IEEE Trans. Signal Process.*, Vol. 41, No. 9, 1993, pp. 2826–2838.

[4] B. De Moor and P. Van Overschee, "Numerical algorithms for subspace state space system identification", in: A. Isidori, ed., *Trends in Control. A European Perspective*, European Control Conference, Italy, 1995, pp. 385–422.

[5] B. De Moor and J. Vandewalle, "A geometrical strategy for the identification of state space models of linear multivariable systems with singular value decomposition", *Proc. 3rd Internat. Symp. on Applications of Multivariable System Techniques*, Plymouth, UK, 13–15 April 1987, pp. 59–69.

[6] W.A. Gardner, *Introduction to Random Processes*, McGraw-Hill, New York, 1990.

[7] B.L. Ho and R.E. Kalman, "Efficient construction of linear state variable models from input/output functions", *Regelungstechnik*, Vol. 14, 1966, pp. 545–548.

[8] S.Y. Kung, "A new identification method and model reduction algorithm via singular value decomposition", *12th Asilomar Conf. on Circuits, Systems and Comp.*, Asilomar, CA, 1978, pp. 705–714.

[9] W.E. Larimore, "Canonical variate analysis in identification, filtering and adaptive control", *Proc. 29th Conf. on Decision and Control*, Hawai, USA, 1990, pp. 596–604.

[10] K. Liu, R.N. Jacques and D.W. Miller, "Frequency domain structural system identification by observability range space extraction", *Proc. Amer. Control Conf.*, Baltimore, MD, Vol. 1, 1994, pp. 107–111.

[11] L. Ljung, *System Identification–Theory for the User*, Prentice Hall, Englewood Cliffs, NJ, 1987.

[12] T. McKelvey, "An efficient frequency domain state-space identification algorithm", *Proc. 33rd IEEE Conf. on Decision and Control*, Lake Buena Vista, FL, USA, 14–16 December 1994, pp. 3359–3364.

[13] T. McKelvey, Identification of state-space models from time and frequency data, Ph.D. Thesis, Department of Electrical Engineering, Division of Automatic Control, Linköping University, Sweden, April 1995.

[14] M. Moonen, B. De Moor, L. Vandenberghe and J. Vandewalle, "On and off-line identification of linear state space models", *Internat. J. Control*, Vol. 49, No. 1, 1989, pp. 219–232.

[15] Y. Rolain, R. Pintelon, K.Q. Xu and H. Vold, "On the use of orthogonal polynomials in high order frequency domain system identification and its application to modal parameter estimation", *Proc. 33rd IEEE Conf. on Decision and Control*, Lake Buena Vista, FL, USA, 14–16 December 1994, pp. 3365–3373.

[16] P. Van Overschee and B. De Moor, *Subspace Identification for Linear Systems. Theory – Implementations – Applications*, Kluwer Academic Publishers, Dordrecht, 1996.

[17] P. Van Overschee and B. De Moor, "N4SID: subspace algorithms for the identification of combined deterministic-stochastic systems", *Automatica, Special Issue on Statistical Signal Processing and Control*, Vol. 30, No. 1, 1994, pp. 75–93.

[18] M. Verhaegen, "Identification of the deterministic part of MIMO state space models given in innovations form from input–output data", *Automatica, Special Issue on Statistical Signal Processing and Control*, Vol. 30, No. 1, 1994, pp. 61–74.

[19] M. Viberg, "Subspace methods in system identification", *Proc. SYSID '94*, Vol. 1, 4–6 July 1994, Copenhagen, Denmark, pp. 1–12.