Subspace-Based Signal Analysis Using Singular Value Decomposition

ALLE-JAN VAN DER VEEN, STUDENT MEMBER, IEEE, ED F. DEPRETTERE, SENIOR MEMBER, IEEE, AND A. LEE SWINDLEHURST, MEMBER, IEEE

In this paper, we present a unified approach to the (related) problems of recovering signal parameters from noisy observations and the identification of linear system model parameters from observed input/output signals, both using singular value decomposition (SVD) techniques. Both known and new SVD-based identification methods are classified in a subspace-oriented *scheme. The singular value decomposition of a matrix constructed from the observed signal data provides the key step to a robust discrimination between desired signals and disturbing signals in terms of signal and noise subspaces. The methods that are presented are contrasted by the way in which the subspaces are determined and how the signal or system model parameters are extracted from these subspaces. Typical examples such as the direction-of-arrival problem and system identification from input/output measurements are elaborated upon, and some extensions to time-varying systems are given.*

I. INTRODUCTION

The analysis of time series is a fundamental problem in almost all scientific disciplines. In engineering parlance, time series are called *signals* and their analysis generally serves at least one of two possible purposes. First, the signals themselves are of prime interest and are to be recognized or recovered by the analysis procedure, as for example in communication applications. Secondly, the signals bear information pertinent to the physical dynamical systems that produced them, or to the hypothetical dynamical systems that could have produced them. In the latter case, the analysis of the signal should provide the unknown system parameters.

A typical example of the first class of problems is the following. Consider a number of signals $s_i(t)$, modulated by a known carrier frequency, and suppose that only a number of unknown linear combinations $x_k(t)$ of these signals have been received at sensors located at different points. We assume that each of the coefficients of these linear combinations is a known function of both the (known) sensor positions and some (unknown) parameter ϕ_i of each signal. The objective is to reconstruct the original signals from the received signals, which will be possible if we first determine the actual values of the parameters ϕ_i, and subsequently identify the pairs $(\phi_i, s_i(t))$ for each of the signals. We can think of the ϕ_i as being spatial directions from which the signals of interest $s_i(t)$ are received.

As an example of the second class of problems, suppose we have recorded two signals, $u(t)$ and $y(t)$, where $u(t)$ is a test signal that is applied at some point in a system, and $y(t)$ is a response signal measured at some other point in the system. If we represent the system mathematically as the mapping $u(t) \rightarrow y(t) = T(u(t))$, where T satisfies certain causality and linearity constraints, then the problem may be stated as one of using $u(t)$ and $y(t)$ to either identify the map T, or to find a map $\hat{T}$ of low complexity that is close, in some sense, to T.

It is instructive and useful to notice that the two problems alluded to above are sometimes quite similar. For example, if the mapping T of our second example is a causal, linear and time-invariant operator, then it is in fact a matrix multiplicative operator that is completely determined by the response $h(t)$ due to a unit impulse excitation $u(t) = \delta(t)$. This impulse response and all of its time-shifted versions constitute the rows of the matrix map. Moreover, if the system is finite, meaning that it can be described by a difference equation of finite order, then this impulse response must be a linear combination of a number of exponentially decaying functions of time, where the exponential factors are the unknown parameters to be determined first. The description of this signal (a weighted sum of elementary signals described by a single parameter) is very similar to the description of each of the received signals in the first example, and the two problems may even become identical in certain specific application scenarios. What we observe here is that the impulse response $h(t)$, much as was the case with the

Manuscript received March 9, 1992; revised March 4, 1993. This review was supported in part by the commission of the EC under the ESPRIT BRA program 6632 (NANA2). Dr. Swindlehurst was supported by the National Science Foundation under Grant MIP-9 110 112.

A. J. van der Veen and E. F. Deprettere are with the Department of Electrical Engineering, Delft University of Technology, 2628 CD Delft, The Netherlands.

A. L. Swindlehurst is with the Department of Electrical and Computer Engineering, Brigham Young University, Provo, UT 84602.

IEEE Log Number 9211776.

0018–9219/93$03.00 © 1993 IEEE

recorded signals $x_k(t)$, explicitly reveals parameters, in particular the *poles* of the presumed system model that directly or indirectly define a realization for the model. The determination of the realization parameters of a predefined model is called *system identification*. System identification techniques can also be used to determine signal models as well. For example, a signal composed of a sum of damped complex exponentials may be thought of as the output of a certain linear system in response to a known or presumed excitation. Identifying this "system" will then provide a model for the signal.

Whether the objective is to recover a signal, to model a signal, or to identify a linear system, the choice of the structure of the signal (or the model of the system) plays a crucial role. Surely, *a priori* knowledge of the signal properties must be incorporated into the model, but we must also account for uncertainties in a proper way, that is to say, in such a way that they do not introduce modeling artifacts. But even when these choices have been made successfully, the subsequent signal analysis can be carried out along many different routes, and its success will depend on three important additional choices: 1) the kind of realization that we have in mind, 2) the analysis strategy, and 3) the tightness of the coupling between the analysis procedure and the system realization. What comes into play here are aspects of numerical stability, minimality, and tightness of approximation. Numerical stability guarantees robustness of the analysis procedure, minimality avoids artifacts due to opaque dependencies between excess parameters, and tightness of approximation has to do with convergence of the analysis procedure. The ideal situation occurs when the analysis procedure directly constructs a realization of the model that has been chosen to have a necessary and sufficient number of parameters, and to have low sensitivity with respect to perturbations of its parameters.

In all practical applications, the observed signals are corrupted versions of the observations that we would expect under ideal circumstances. The unavoidable contaminations are commonly called *noise*, and they obstruct the extraction of the true or desired parameters from the analysis of the observed signals. Consequently, the goal of any given identification method is to find the signal model parameters that best match the noise-corrupted observations. Commonly used approaches include maximum likelihood estimation (estimation of the parameters of the model that, in a probabilistic sense, most likely produced the observed signal) and least squares error minimization (yielding the parameters of the model that optimally approximates the observed signal in terms of minimal energy of the difference signal). For an overview of many such identification methods, see [1]–[3].

In practice, therefore, the choice of the signal or system model has to be complemented by the choice of a noise model and an optimization criterion. For example, in terms of the two classes of applications mentioned above, and with the assumption that the noise is additive, the noise could be due to interfering signals that are received from directions outside the focus area, or it could be due to receiver equipment noise (class 1). On the other hand, it could be part of the impulse response corresponding to higher order modes that are not of interest (class 2). The selection of the signal or system model, the noise model, and the optimization criterion will in general depend on any *a priori* available knowledge, desired accuracy, etc., or in short on a number of design variables. Choosing values for these variables may be quite difficult, and an optimal choice may only be possible by trial and error. This makes identification as much an art as it is a science.

In this paper, we will focus on signals and systems that fit deterministic *state-space* models. State-space models cover causal and finite systems that may be neither linear nor time-invariant. If they are linear and time-invariant, then they are closely related to constant coefficient difference equations relating input and output signals. In a function-theoretic framework, these models in turn become rational (expressed by a ratio of two polynomials), and are also called pole–zero models. However, while such models are global input/output characterizations of the system, state-space models also take the internal system behavior into account by describing the current output as a function of a current internal state and the current input, and by describing the next state as a function of the current state and the current input. A linear, time-invariant system is simply one for which these functions are themselves linear and time-invariant. The order of the state-space model is the dimension of the state vector, or more precisely, that of the state space, and is a measure of the system's memory capacity.

In this paper, we will only be concerned with linear state-space models, and we will require that all signals (input, output, and state signals) belong to certain normed spaces. The analysis of these signals and their models is done through extensive use of linear algebra. Signals are represented as (possibly infinite-length) vectors, and the state-space model is taken to be a matrix map from the input space and state space to the output and state spaces. The observations from which such a map is to be identified do not in general include the (internal) state signals, so estimation of the model order becomes an essential part of the identification problem. The presence of noise turns this problem into a difficult one, since noise tends to reveal itself as an increased state-space dimension. In order to discriminate against noise, our approach will essentially be the following. We collect the observed signal or signals in a so-called observation matrix, which will often inherit a certain (Hankel) structure from the natural ordering imposed by the state-space model. Decomposing the column (range) space of this matrix into a dominant and a subordinate part reveals which of its subspaces can be attributed to the noise-free signal or signals and which can be attributed to the noise. We will assume that these two subspaces are orthogonal to each other, which implies that in terms of inner products, the noise-free signals and disturbances are independent of one another. The dominant subspace is due to the signals and is referred to as the *signal subspace*, while the other is referred to as the *noise subspace*.

The designated tool used to decompose the range space of the observation matrix into these two complementary subspaces is the singular value decomposition (SVD). The SVD is computationally very robust and allows for *high-resolution* discrimination against noise contamination. Once the signal subspace has been determined, the model parameters are extracted from it. This approach gives rise to a number of subspace-based approaches, and we will be interested in understanding the basic differences between them. Again, these approaches correspond to different model assumptions, specific design parameters, or alternative ways of computing what are essentially the same quantities. Associated with each of these approaches is a certain algorithm: a computational scheme. However, we will focus on the basic principles of subspace modeling—also called *low-rank approximation*—rather than dwelling on the algorithmic details. We will strive to provide a unified description of low-rank approximation methods, while at the same time pointing out the particularities of each of the approaches with respect to the generic solution.

The paper can be divided into two main parts. In the first, the generic problem we are considering is described, and several relevant applications are presented. The second part of the paper is concerned with various classes of algorithms that have been developed over the years for these applications. Linking the two parts of the paper is a discussion of the SVD, which is both a theoretical and computational tool used in the analysis of the data models and the development of appropriate algorithms.

In the first part of the paper, Section II presents an introduction to linear system realization theory, which can be viewed as identification in the absence of noise. The *shift-invariance* structure present in the data matrices is shown to be a crucial property. Section III illustrates the presence of such shift-invariant data structures in four identification scenarios: realization theory for time-varying systems, pole estimation from input–output measurements, direction-of-arrival estimation in antenna array applications, and harmonic retrieval of sinusoidal signals. Section IV then contains the intermediate discussion of the properties of the SVD that we will use in this paper.

The second part of the paper consists of Sections V–IX, and contains details concerning the actual identification algorithms under consideration. An overview of these algorithms is given in Section V, which leads to a classification of the available methods into three classes, which are subsequently treated in Sections VI–VIII. The methods in Section VI (a.o. TAM, ESPRIT) are algebraic and are based on the single-shift structure observed between two submatrices of the data matrix. The methods in Section VII (Min-Norm, AAK) are in a sense intermediate; while they can be described using submatrices as in Section VI, they are based on the analytic (i.e., polynomial) properties of one vector selected from the noise subspace orthogonal to the signal subspace. This is elaborated upon in Section VIII, where the analytic properties of the full noise subspace (or equivalently, the full signal subspace) are taken into account (Max Likelihood, MUSIC, Weighted Subspace Fitting, MODE). The general objective in these approaches is to find a low-rank subspace with a shift structure that has minimal distance to the true signal space, or equivalently, that is as orthogonal to the noise subspace as possible. To conclude the paper, Section IX gives a review of recent work on the statistical accuracy and computational load of the above algorithms.

Several parts of the contents of this paper have appeared in separate tutorials and books, in particular the material on the SVD and elementary system theory. In the context of signal processing, introductory texts on SVD and linear prediction methods can be found in [4], [5]. During the review of this paper, a related tutorial by Rao and Arun on subspace-based model identification was published [6]. Obviously, there is some overlap between their paper and ours. The present paper gives more details concerning the classification of single shift-invariant methods, and also features some maximum-likelihood and Hankel-norm approximation methods. In addition, we consider an application to time-varying systems, and model identification from input/output data.

A. Notation

Throughout this paper, the superscript * denotes complex conjugate transpose and the superscript $^{\mathrm{T}}$ denotes the ordinary matrix transpose. The superscript $\hat{}$ is used either to denote a low-rank approximant of a matrix, or the reduction of a matrix to a smaller size by omitting some rows or columns. The ith column (or sometimes row) of a matrix X is denoted by $\boldsymbol{x}_i$. In addition, for the polynomial constructed from a vector $\boldsymbol{u} = [u_1 \quad u_2 \quad \cdots]^{\mathrm{T}}$, we will use the notation $u(z) = \boldsymbol{u}^* \boldsymbol{a}(z) = \bar{u}_1 + \bar{u}_2 z + \cdots$, with $\boldsymbol{a}(z) = [1 \quad z \quad z^2 \quad \cdots]^{\mathrm{T}}$, for $z \in \mathbb{C}$.

For a one-sided infinite matrix (operator) H, we denote by $H^{\uparrow}$ the operator H with its top row removed. Likewise, $H^{\leftarrow}$ is the operator H minus its first column. For a finite matrix H of size $(L+1) \times N$, $H^{(1)}$ is the $L \times N$ matrix containing the first L rows of H, and $H^{(2)}$ is the matrix containing the last L rows of H.

The matrix I_d is the identity matrix of size $d \times d$. The range of a matrix H of size $L \times N$ is the space $\{H\boldsymbol{x} : \boldsymbol{x} \in \mathbb{C}^N\}$, which is a subspace in the Euclidean space $\mathbb{C}^L$. The kernel of H is the subspace $\{\boldsymbol{x} \in \mathbb{C}^N : H\boldsymbol{x} = 0\}$. Projectors onto subspaces are denoted by Π. $\mathrm{Tr}\,(F)$ denotes the trace of a matrix F, i.e., the sum of the diagonal entries of F. $\mathrm{Eig}\,(F)$ denotes the diagonal matrix containing the eigenvalues λ_i of F.

II. Introduction to Linear System Realization Theory

The realization problem for linear systems is already a fairly old subject. A state-space approach to this problem was introduced by Nerode [7], and was subsequently formalized by Ho and Kalman [8]. The realization scheme is based on the analysis of certain subspaces spanned by "inputs in the past" in combination with "outputs in the future." In the mid-1970's, the SVD was introduced as

a tool to identify these subspaces in a numerically stable way, and for obtaining an approximate realization of lower order than the true system order [9]–[11]. This section will introduce some system theoretic notions with relevance to subspace-based system realization theory. Section III will apply this theory to a few standard identification scenarios that will be used throughout this paper. More background material on linear systems theory can be found in the books by Kailath [12] and Rugh [13].

A. System Operator

Consider a causal linear time-invariant (LTI) system with system transfer operator T, mapping an input vector (sequence) that represents an input signal

$$\boldsymbol{u} = [\cdots \quad u_{-1} \quad u_0 \quad u_1 \quad \cdots]^{\mathrm{T}}$$

to a corresponding output sequence

$$\boldsymbol{y} = [\cdots \quad y_{-1} \quad y_0 \quad y_1 \quad \cdots]^{\mathrm{T}}$$

such that $\boldsymbol{y} = T\boldsymbol{u}$. For simplicity of notation, we consider systems with only one input and one output, although the general case follows easily along the same lines. We take the input and output sequences to be of finite energy, $\|\boldsymbol{u}\|_2^2 = \boldsymbol{u}^*\boldsymbol{u} \leq M < \infty$, so that they are elements of the Hilbert space ℓ_2 (see, e.g., [14]), and we take T to be a bounded (stable) operator acting from ℓ_2 to ℓ_2. Associated with T is its impulse response

$$\begin{aligned}\boldsymbol{h} &= [\cdots 0 \quad 0 \quad h_0 \quad h_1 \quad h_2 \quad \cdots]^{\mathrm{T}} \\ &= T[\cdots 0 \quad 0 \quad 1 \quad 0 \quad 0 \cdots]^{\mathrm{T}}\end{aligned}$$

which is the response of the system to a unit impulse applied at time 0. The operator T has a matrix representation such that $\boldsymbol{y} = T\boldsymbol{u}$ fits the usual rules for matrix–vector multiplications:

$$T = \begin{bmatrix} \ddots & & & & \\ \cdots & h_0 & & \mathbf{0} & \\ & h_1 & h_0 & & \\ \cdots & h_2 & h_1 & h_0 & \\ & \vdots & & \vdots & \ddots \end{bmatrix}.$$

The ith column contains the impulse response due to an impulse at time i. Note that the above relationship relies on the linearity of the system. The input $\boldsymbol{u}$ can be thought of as consisting of a sum of impulses, one for each time instant i, weighted by u_i. The output of the system is then the weighted sum of the responses to these impulses. This description is equivalent to the familiar convolution sum $\boldsymbol{y} = \boldsymbol{h} * \boldsymbol{u}$, defined by $y_i = \sum_{k=0}^{\infty} h_k u_{i-k}$. Because of time invariance, the matrix representation has a Toeplitz structure: it is constant along diagonals. It is lower triangular due to causality.

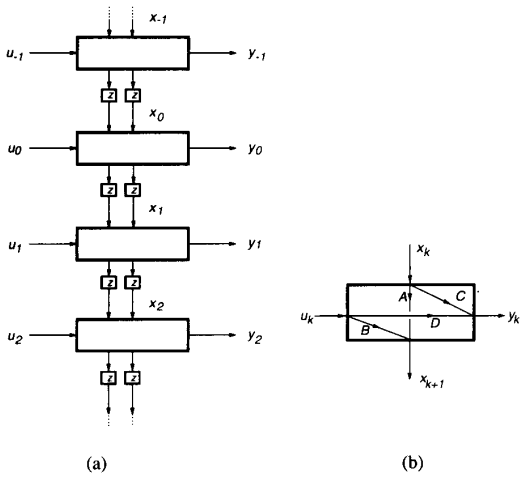


Fig. 1. LTI state-space model. (a) Mapping of an input sequence $\{u_i\}$ to an output sequence $\{y_i\}$ using an intermediate state sequence $\{x_i\}$. The state dimension is $d = 2$. Due to causality, the signal flow is from top to bottom. The delay operator z denotes a time shift here. (b) The operation at a particular time instant k is a linear map from input u_k and current state x_k to output y_k and next state x_{k+1}.

B. State-Space Representation

The familiar state-space model used to describe causal LTI systems is

$$\begin{aligned} x_{k+1} &= Ax_k + Bu_k \\ y_k &= Cx_k + Du_k \end{aligned}$$

in which x_k is the state vector (assumed to have d entries), A is a $d \times d$ matrix, B and C^{T} are $d \times 1$ vectors, and D is a scalar (see Fig. 1). The integer d is called the state dimension or system order. All finite-dimensional linear systems can be described in this way. The *realization problem* is to find a state-space representation that matches a given system operator T, i.e.

$$\boldsymbol{y} = T\boldsymbol{u} \quad \Leftrightarrow \quad \begin{bmatrix} x_{k+1} \\ y_k \end{bmatrix} = \begin{bmatrix} A & B \\ C & D \end{bmatrix} \begin{bmatrix} x_k \\ u_k \end{bmatrix} \tag{1}$$

such that the impulse response of the state-space system

$$\boldsymbol{h} = [\cdots \quad 0 \quad D \quad CB \quad CAB \quad CA^2B \quad \cdots]^{\mathrm{T}} \tag{2}$$

matches the impulse response of T. In principle, there exist an infinite number of state-space realizations for a given system. For example, the state vector x_k might contain some states that are not observed in the output or that are never excited by the input. Hence, we will limit our attention to minimal state-space models, that is, models for which the state dimension d is minimal. It is well known that for minimal systems, in order to have $\boldsymbol{h} \in \ell_2$, the eigenvalues of A must be smaller than 1 in absolute value, although eigenvalues on the unit circle are allowed in some applications.

Even for minimal systems, the representation (1) is not at all unique. An equivalent system representation (yielding the same input–output relationship) is obtained by applying a state transformation R (an invertible $d \times d$ matrix) to

define a new state vector $x'_k = Rx_k$. The equivalent system is

$$x'_{k+1} = A'x'_k + B'u_k$$
$$y_k = C'x'_k + Du_k$$

where the new state space quantities are given by

$$\begin{bmatrix} A' & B' \\ C' & D \end{bmatrix} = \begin{bmatrix} R^{-1} & \\ & 1 \end{bmatrix} \begin{bmatrix} A & B \\ C & D \end{bmatrix} \begin{bmatrix} R & \\ & 1 \end{bmatrix}.$$

The eigenvalues of A remain invariant under this transformation since $R^{-1}AR$ is a similarity transformation [15]. The eigenvalues of A are directly related to the poles of the system, a fact that is easily verified if these poles are distinct. Under the assumption of distinct poles, another way to describe linear systems is via a partial fraction expansion of the z-transform of the impulse response $\boldsymbol{h}$

$$h(z) = \sum_0^\infty h_n z^n = r_0 + \sum_{i=1}^{d} \frac{r_i z}{1 - \phi_i z} \tag{3}$$

where ϕ_i^{-1}, $i = 1, \cdots, d$, are the d poles of the system, and r_i, $i = 1, \cdots, d$, their respective residues. A corresponding state-space realization is

$$\left[\begin{array}{cc} A & B \\ C & D \end{array}\right] = \left[\begin{array}{ccc|c} \phi_d & & & r_d \\ & \ddots & & \vdots \\ & & \phi_2 & r_2 \\ & & & \phi_1 \quad r_1 \\ \hline 1 & \cdots & 1 & 1 \quad r_0 \end{array}\right].$$

Another way to obtain this decomposition is to start from a given realization $\{A, B, C, D\}$ and apply an appropriate state similarity transformation that will diagonalize the A matrix: $A = R\Phi R^{-1}$. This is an eigenvalue decomposition of A, and the entries of Φ are the eigenvalues of A. A sufficient condition for the existence of this decomposition (i.e., an invertible R) is that the poles of the system be distinct [12].

C. Hankel Operator

We now turn to the realization problem: given a system transfer operator T (or equivalently an impulse response $\boldsymbol{h}$), how can a state-space model that realizes this transfer operator be determined? The solution to the realization problem in a subspace context calls for the Hankel operator, which we define presently.

The idea is to apply inputs only up to time $t = -1$ (called "the past" with respect to the present time instant $t = 0$) and measure the resulting outputs from $t = 0$ on (the future; see Fig. 2). Writing $\boldsymbol{y} = T\boldsymbol{u}$, we have

$$\begin{bmatrix} \vdots \\ \times \\ \times \\ y_0 \\ y_1 \\ y_2 \\ \vdots \end{bmatrix} = \begin{bmatrix} & \ddots & & & & & \\ \cdots & \times & \times & & & \mathbf{0} & \\ & \times & \times & \times & & & \\ \cdots & h_3 & h_2 & h_1 & \times & & \\ & & h_3 & h_2 & \times & \times & \\ & \iddots & & h_3 & \times & \times & \ddots \\ & & & \vdots & & \vdots & \end{bmatrix} \begin{bmatrix} \vdots \\ u_{-3} \\ u_{-2} \\ u_{-1} \\ 0 \\ 0 \\ \vdots \end{bmatrix}. \tag{4}$$

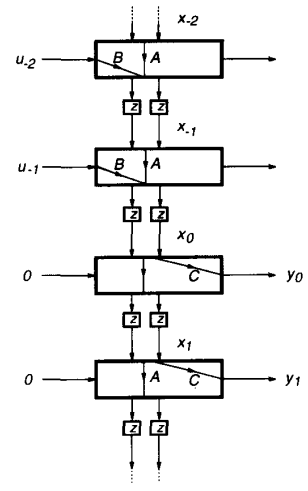


Fig. 2. Applying inputs up to $t = -1$ and recording outputs from $t = 0$ on yields information about the state at $t = 0$. From this, a state space realization can be derived.

From this equation it is seen that only the lower left corner of T is actually used. Since this part operates on a one-sided infinite sequence, we can bring it into a more familiar form by defining a past input sequence and a future output sequence as the one-sided sequences

$$\boldsymbol{u}_- = [u_{-1} \quad u_{-2} \quad \cdots]^{\mathrm{T}} \qquad \boldsymbol{y}_+ = [y_0 \quad y_1 \quad \cdots]^{\mathrm{T}}$$

from which we can write (4) as $\boldsymbol{y}_+ = H\boldsymbol{u}_-$ with H defined by

$$H = \begin{bmatrix} h_1 & h_2 & h_3 & \cdots \\ h_2 & h_3 & & \\ h_3 & & \ddots & \\ \vdots & & & \end{bmatrix}. \tag{5}$$

The matrix H has what is called Hankel structure: it is constant along the antidiagonals. As outlined below, it has a number of important properties that will enable us to derive state-space models from it.

1) *H has rank d, equal to the minimal system order.* This follows from inserting (2) into (5), or alternatively, by inspection of Fig. 2 directly: $\boldsymbol{y}_+ = H\boldsymbol{u}_-$ is computed in two stages,

$$\begin{cases} x_0 &= \mathcal{C}\boldsymbol{u}_- \\ \boldsymbol{y}_+ &= \mathcal{O}x_0 \end{cases}$$

where

$$\mathcal{O} = \begin{bmatrix} C \\ CA \\ CA^2 \\ \vdots \end{bmatrix}; \qquad \mathcal{C} = [B \quad AB \quad A^2B \quad \cdots]. \tag{6}$$

Clearly, H has a factorization $H = \mathcal{O}\mathcal{C}$. $\mathcal{C}$ is called the controllability operator and $\mathcal{O}$ is called the observability operator, and for a minimal realization they have by definition full rank d. Since H is an

outer product of rank d matrices, it must be of rank d itself. Even for minimal realizations, there is of course an ambiguity in this factorization. With R an invertible $d \times d$ matrix, we can also factor H as $H = \mathcal{O}'\mathcal{C}' = \mathcal{O}R \cdot R^{-1}\mathcal{C}$, corresponding to a state-space model that has undergone a state transformation by R as described above. Factorizations modulo R lead to equivalent systems.

2) *H has a shift-invariance structure.* Denote by $H^{\uparrow}$ the operator H with its top row deleted. Likewise, denote by $H^{\leftarrow}$ the operator H with its first column deleted. Shift-invariance means that the range (column space) of the shifted operator is contained in the range of the original operator. This property can be deduced directly from the Hankel structure in (5)

$$\begin{aligned} H^{\uparrow} &= \mathcal{O}^{\uparrow}\mathcal{C} = \mathcal{O}A \cdot \mathcal{C} \\ H^{\leftarrow} &= \mathcal{O}\mathcal{C}^{\leftarrow} = \mathcal{O} \cdot A\mathcal{C}. \end{aligned}$$

Thus it is seen that shifting H upwards or to the left is equivalent to a multiplication by A in the center of the factorization.

There is a physical interpretation of this shift invariance. Just as the range of H contains all possible outputs of the system from $t = 0$ on, due to inputs that last until $t = -1$, the range of $H^{\uparrow}$ contains all possible outputs of the system from $t = 1$ on, due to inputs that stop at $t = -1$. Because of the time-invariance of the system, this is the same as stating that $H^{\uparrow}$ contains the outputs of the system from $t = 0$ on, due to all inputs that stop at $t = -2$. This set of inputs is a subspace in the set of all inputs in the past, and hence the resulting set of future outputs (the range of $H^{\uparrow}$) must be a subspace contained in the original set of future outputs (the range of H).

D. Realization Scheme

Using the above two properties of the Hankel operator H—i.e., that it is of finite rank with some minimal factorization $H = \mathcal{O}\mathcal{C}$, and that it is shift-invariant—we will show how to obtain a state-space realization as in (1) from a given transfer operator T.

1) Given T, construct the Hankel operator H as in (5). Determine the rank d of the operator, and a factorization $H = \mathcal{O}\mathcal{C}$, where $\mathcal{O}$ and $\mathcal{C}$ are of full rank d. The SVD is a robust tool for doing this, as will be discussed later.

2) At this point, we know that $\mathcal{C}$ and $\mathcal{O}$ have the shift-invariant structure of (6). Use this property to derive

$$\mathcal{O}A = \mathcal{O}^{\uparrow} \qquad \Rightarrow \qquad A = \mathcal{O}^{+}\mathcal{O}^{\uparrow}$$

where $\mathcal{O}^{+}$ is the pseudo-inverse of $\mathcal{O}$ such that $\mathcal{O}^{+}\mathcal{O} = I_d$. Because $\mathcal{O}$ is of full row rank d, we have $\mathcal{O}^{+} = (\mathcal{O}^{*}\mathcal{O})^{-1}\mathcal{O}^{*}$. This determines A. The matrices B, C and D follow simply as

$$\begin{aligned} B &= \mathcal{C}_{(:,1)} \\ C &= \mathcal{O}_{(1,:)} \\ D &= h_0 \end{aligned}$$

where the subscript $(:, 1)$ denotes the first column of the associated operator, and $(1, :)$ the first row.

Various issues emerge here to make this realization scheme feasible in practice. First, we are only willing to do computations on matrices of finite size. In particular, H should have finite size. This issue can be dealt with relatively easily. Suppose we have available a top-left $(L+1) \times N$ window of the infinitely dimensioned H:

$$\begin{aligned} H_{L+1,N} &= \begin{bmatrix} h_1 & h_2 & \cdots & h_N \\ h_2 & h_3 & & h_{N+1} \\ \vdots & & \ddots & \vdots \\ h_{L+1} & h_{L+2} & \cdots & h_{N+L} \end{bmatrix} \\ &= \begin{bmatrix} C \\ CA \\ \vdots \\ CA^{L} \end{bmatrix} \cdot \begin{bmatrix} B & AB & A^2B & \cdots & A^{N-1}B \end{bmatrix} \\ &= \mathcal{O}_{L+1} \cdot \mathcal{C}_N \,. \end{aligned} \tag{7}$$

Define $\mathcal{O}^{(1)}$ and $\mathcal{O}^{(2)}$ by

$$\mathcal{O}_{L+1} = \begin{bmatrix} \mathcal{O}^{(1)} \\ CA^{L} \end{bmatrix} = \begin{bmatrix} C \\ \mathcal{O}^{(2)} \end{bmatrix}$$

and as before, let d be the rank of H. If L and N are equal to or larger than d, then the rank of $H_{L+1,N}$ is also equal to d, and in particular $\mathcal{O}^{(1)}$ and $\mathcal{O}^{(2)}$ are of full rank d. The shift-invariance property in this finite-size case is now

$$\mathcal{O}^{(2)} = \mathcal{O}^{(1)}A \qquad \Rightarrow \qquad A = \mathcal{O}^{(1)+}\mathcal{O}^{(2)} \tag{8}$$

and $A = \mathcal{O}^{(1)+}\mathcal{O}^{(2)}$ is the same matrix as obtained in the infinite case.

A second issue is how to handle an inaccurate T. This is more difficult to treat, and in fact is the subject of most of the remaining part of the paper. Suppose that T is corrupted by additive noise, or alternatively, that we have measured an impulse response sequence $\boldsymbol{h}$ which contains additive noise. The matrix H will thus be constructed from noisy measurements, and will therefore have full rank in general. H will also have high rank if T represents a system of high order for which a "reduced order" model is desired. In both cases (system identification and model reduction) the objective is to find an approximate system with Hankel matrix $\hat{H}$ of low rank d that, in some suitable norm, is as close to the original noisy H as possible. In essence, the problem is to determine the optimal (or close-to-optimal) positions of the poles of the approximating system, or in other words to estimate the $d \times d$ diagonal matrix $\Phi = \mathrm{eig}\,(A)$ given a finite extent of the impulse response. At first, we will consider only the shift-invariant structure in the observability matrix $\mathcal{O}$. The key problem (and also the major distinction between the various algorithms) is how to enforce the shift-invariant structure present in the original or noisy $\mathcal{O}$ to be present in the approximation too.

E. Discussion

There is a subspace theory underlying the low-rank and shift-invariance properties that we have used implicitly. We

assumed the existence of a model as in (1), and used the resulting properties to derive the structure of $\mathcal{C}$ and $\mathcal{O}$ as in (6). A proof of the existence of this model starts from some system transfer operator T and its Hankel operator H. We briefly touch upon this subject. Let the minimal system order be d, and let $H = \mathcal{O}\mathcal{C}$ with $\mathcal{O}$ and $\mathcal{C}$ of full rank d. The output state space $\mathcal{H}_0$ is the subspace defined by

$$\mathcal{H}_0 = \{\boldsymbol{y}_+ : \boldsymbol{y}_+ = H\boldsymbol{u}_-, \text{ all } \boldsymbol{u}_- \in \ell_2\}.$$

$\mathcal{H}_0$ is the subspace of all possible outputs in "the future" that can be reached by inputs in the past. Mathematically, $\mathcal{H}_0$ is the range ("column space") of H, and the d columns of $\mathcal{O}$ constitute a minimal basis for it. Likewise, define the input null space $\mathcal{M}$ and input state space $\mathcal{H}$ as

$$\mathcal{M} = \{\boldsymbol{u}_- : \boldsymbol{y}_+ = H\boldsymbol{u}_- = 0\}$$
$$\mathcal{H} = \mathcal{M}^{\perp}.$$

$\mathcal{M}$ is the kernel of H and consists of all inputs in the past that yield zero output in the future. $\mathcal{H}$ is the orthogonal complement of $\mathcal{M}$ and is equal to the column space of H^*, or the conjugate transpose of the row range space of H. The d columns of $\mathcal{C}^*$ constitute a minimal basis for $\mathcal{H}$.

Using the above spaces $\mathcal{H}$ and $\mathcal{H}_0$ and making use of the assumption that they are of finite dimension d, it is possible to formally derive that there must exist a state-space model in the form of (1). We omit this derivation, but remark that crucial in the derivation is the fact that $\mathcal{H}$ and $\mathcal{H}_0$ are shift-invariant; e.g., the space $\mathcal{H}_0^{\uparrow}$ is contained in $\mathcal{H}_0$. It follows that their bases must also be shift-invariant, and hence that there must be some matrix A to express the shifted basis in terms of the original: $\mathcal{O}^{\uparrow} = \mathcal{O}A$. This gives rise to the now familiar structures of $\mathcal{C}$ and $\mathcal{O}$, and is the content of the abstract realization theory in [16], [17].

III. Applications of Subspace-Based Realization Theory

In this section, we discuss a number of related identification problems that rely on the same type of low-rank and shift-invariance properties described in the previous section. We first discuss the realization problem for time-varying systems, and show that the resulting time-varying Hankel operator is of low rank and has a shift-invariance property which can be used to determine a time-varying state-space realization. A second application is system identification using input–output data. In this problem, the impulse response is not specified, but instead a measured collection of inputs and their corresponding outputs is given. The third application is the direction-of-arrival estimation problem, in which one attempts to determine the incidental directions of a number of narrowband plane wave signals impinging on an antenna array. Finally, in the fourth application, we discuss the classical harmonic retrieval problem, where one attempts to determine the frequencies and decay factors of multiple cisoids.

A. Realization of a Time-Varying System

The purpose of this section is to give a brief introduction to realization theory for time-varying systems, primarily to demonstrate the generality of the subspace concept. The derivation is very similar to the time-invariant case, and a more detailed discussion along these lines can be found in [18], [19]. Consider again an input sequence $\boldsymbol{u} \in \ell_2$, which is mapped by an operator T to a corresponding output sequence $\boldsymbol{y} = T\boldsymbol{u}$, where

$$\boldsymbol{u} = [\cdots \quad u_{-1} \quad u_0 \quad u_1 \quad \cdots]^{\mathrm{T}}$$
$$\boldsymbol{y} = [\cdots \quad y_{-1} \quad y_0 \quad y_1 \quad \cdots]^{\mathrm{T}}.$$

T is assumed to be bounded and causal, and hence has a matrix representation

$$T = \begin{bmatrix} \ddots & & & & \\ \cdots & h_{00} & & \mathbf{0} & \\ & h_{10} & h_{11} & & \\ \cdots & h_{20} & h_{21} & h_{22} & \\ & \vdots & & \vdots & \ddots \end{bmatrix}.$$

As before, the ith column of T is the response of the system to an impulse applied at time $t = i$, but because the system is time-varying, these impulse responses can change with time. We have thus lost the Toeplitz structure of T.

A time-varying state-space realization has the form

$$x_{k+1} = A_k x_k + B_k u_k$$
$$y_k = C_k x_k + D_k u_k$$

in which x_k is the state vector at time k (taken to have d_k entries; the state dimensions need not be constant now), A_k is a $d_{k+1} \times d_k$ (possibly nonsquare) matrix, B_k is a $d_{k+1} \times 1$ vector, C_k is a $1 \times d_k$ vector, and D_k is a scalar. Note that, with time-varying state dimensions, the A_k matrices are no longer square matrices, and hence they do not have the eigenvalue decompositions which were used in the time-invariant case to compute the poles of the system. Nonetheless, it is possible to compute time-varying state realizations for a given time-varying system transfer operator T, as the next paragraph will show.

Suppose a time-varying system transfer operator T is given, for which we want to determine a time-varying state-space realization. The approach is as in the time-invariant case. Denote a certain time instant as "current time," apply all possible inputs in the "past" with respect to this instant, and measure the corresponding outputs in "the future," from the current time instant on (see Fig. 3). As in the time-invariant case, we select in this way a lower-left submatrix of T. For example, for the current time $t = 2$,

$$\begin{bmatrix} \vdots \\ \times \\ \times \\ y_2 \\ y_3 \\ y_4 \\ \vdots \end{bmatrix} = \left[\begin{array}{ccccc|ccc} & \ddots & & & & & & \\ \cdots & \times & \times & & \mathbf{0} & & & \\ & \times & \times & \times & & & & \\ \hline \cdots & h_{2,-1} & h_{20} & h_{21} & & \times & & \\ & & h_{30} & h_{31} & & \times & \times & \\ & \iddots & & h_{41} & & \times & \times & \ddots \\ & & & \vdots & & & \vdots & \end{array}\right] \begin{bmatrix} \vdots \\ u_{-1} \\ u_0 \\ u_1 \\ 0 \\ 0 \\ \vdots \end{bmatrix}.$$

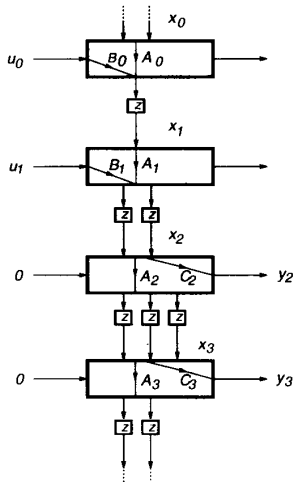


Fig. 3. Principle of the identification of a time-varying state-space model. In this picture, the "current time" is $t = 2$, all possible inputs up till time $t = 1$ ("the past") are applied, and the corresponding output sequences are recorded from time $t = 2$ on ("the future"). This yields H_2, a Hankel operator at instant $t = 2$. This should be done in turn for all t.

Denote by H_2 the one-sided infinite equivalent of this submatrix:

$$H_2 = \begin{bmatrix} h_{21} & h_{20} & h_{2,-1} & \cdots \\ h_{31} & h_{30} & & \\ h_{41} & & \ddots & \\ \vdots & & & \end{bmatrix}.$$

In analogy with the time-invariant case, we call H_2 the Hankel operator at time 2, although in reality it does not possess an antidiagonal Hankel structure. By doing this in turn for all time, we obtain from T a sequence of Hankel operators H_k, viz.,

$$T = \left[\begin{array}{ccc|c|cc|c} \ddots & & & & & & \\ \cdots\ h_{0,-1} & h_{00} & & & & \mathbf{0} & \\ \cdots\ h_{1,-1} & h_{10} & h_{11} & & & & \\ \cdots\ h_{2,-1} & h_{20} & h_{21} & h_{22} & & & \\ \hline & h_{30} & h_{31} & h_{32} & h_{33} & & \\ \iddots & & h_{41} & h_{42} & h_{43} & \ddots & \\ & & \vdots & \vdots & \vdots & & \\ & H_1 & H_2 & H_3 & & & \end{array}\right]. \quad (9)$$

Although we have lost the Hankel structure, we retain the following two important properties:

1) *H_k has rank d_k equal to the minimal system order at time instant k.* While this can be derived formally, it also follows from inspection of Fig. 3. For example, at time 2 we have that the realization satisfies

$$H_2 = \begin{bmatrix} C_2 \\ C_3A_2 \\ C_4A_3A_2 \\ \vdots \end{bmatrix} [B_1 \quad A_1B_0 \quad A_1A_0B_{-1} \quad \cdots]$$
$$= \mathcal{O}_2\mathcal{C}_2$$

where for a minimal system both $\mathcal{O}_2$ and $\mathcal{C}_2$ are of full rank d_2. $\mathcal{C}_k$ and $\mathcal{O}_k$ can be regarded as time-varying controllability and observability matrices.

2) *H_k has shift-invariant properties.* For example

$$H_2^{\uparrow} = \begin{bmatrix} C_3 \\ C_4A_3 \\ C_5A_4A_3 \\ \vdots \end{bmatrix} A_2 [B_1 \quad A_1B_0 \quad A_1A_0B_{-1} \quad \cdots]$$
$$= \mathcal{O}_3A_2\mathcal{C}_2$$
$$H_3 = \begin{bmatrix} C_3 \\ C_4A_3 \\ C_5A_4A_3 \\ \vdots \end{bmatrix} [B_2 \quad A_2B_1 \quad A_2A_1B_0 \quad \cdots]$$
$$= \mathcal{O}_3\mathcal{C}_3$$

The shift-invariance property is now reflected by the fact that the range of $H_k^{\uparrow}$ is contained in the column space of H_{k+1}. This can also be seen from (9). The physical interpretation is the same as in the time-invariant case; i.e., the range of $H_2^{\uparrow}$ contains the output sequences from $t = 3$ on, due to inputs in the past up to $t = 1$, whereas the range of H_3 contains the output sequences from $t = 3$ on, due to inputs that run up to $t = 2$. The latter set of inputs properly contains the former, hence the range of H_3 contains the range of $H_2^{\uparrow}$.

The above properties form the ingredients for obtaining a realization of a given time-varying transfer operator T:

1) First construct Hankel operators H_k from T. Compute the rank d_k of each Hankel operator; this is the system order at time instant k. Compute a decomposition $H_k = \mathcal{O}_k\mathcal{C}_k$ into full rank d_k factors $\mathcal{O}_k$ and $\mathcal{C}_k$. The columns of $\mathcal{O}_k$ form a basis for the output state space at time k, and likewise the columns of $\mathcal{C}_k^*$ form a basis for the input state space at time k.
2) Having obtained $\mathcal{C}_k$ and $\mathcal{O}_k$ for all time instants k, apply the shift-invariance property

$$\mathcal{O}_{k+1}A_k = \mathcal{O}_k^{\uparrow} \quad \Rightarrow \quad A_k = \mathcal{O}_{k+1}^{+}\mathcal{O}_k^{\uparrow}.$$

Thus A_k is of size $d_{k+1} \times d_k$. The matrices B_k, C_k, and D_k follow much as before

$$\begin{aligned} B_k &= \mathcal{C}_{k+1(:,1)} \\ C_k &= \mathcal{O}_{k(1,:)} \\ D_k &= h_{kk} \end{aligned}.$$

It is instructive to perform the above recipe on a numerical example. Consider the finite lower triangular matrix

$$T = \begin{bmatrix} 1 & & & \\ 1/2 & 1 & & \\ 1/6 & 1/3 & 1 & \\ 1/24 & 1/12 & 1/4 & 1 \end{bmatrix}.$$

Finite matrices form a special class of time-varying systems. The Hankel matrices with nonvanishing dimensions

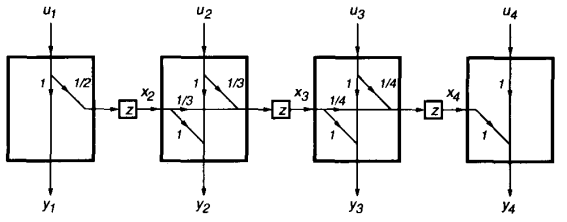


Fig. 4. Time-varying state realization of a finite matrix.

are

$$H_2 = \begin{bmatrix} 1/2 \\ 1/6 \\ 1/24 \end{bmatrix} = \begin{bmatrix} 1 \\ 1/3 \\ 1/12 \end{bmatrix} 1/2$$

$$H_3 = \begin{bmatrix} 1/3 & 1/6 \\ 1/12 & 1/24 \end{bmatrix} = \begin{bmatrix} 1 \\ 1/4 \end{bmatrix} [1/3 \quad 1/6]$$

$$H_4 = [1/4 \quad 1/12 \quad 1/24].$$

Since $\operatorname{rank}(H_k) = 0$ for $k < 2$ and $k > 4$, no states are needed at these points in time. One state is needed for x_2 and one for x_4, because $\operatorname{rank}(H_2) = \operatorname{rank}(H_4) = 1$. Finally, also only one state is needed for x_3, because $\operatorname{rank}(H_3) = 1$. In fact, this is (for this example) the only nontrivial rank condition: if one of the entries in H_3 would have been different, then two states would have been needed. The realization algorithm leads to the sequence of realization matrices

$$\begin{bmatrix} A_1 & B_1 \\ C_1 & D_1 \end{bmatrix} = \begin{bmatrix} \cdot & 1/2 \\ \cdot & 1 \end{bmatrix}$$

$$\begin{bmatrix} A_2 & B_2 \\ C_2 & D_2 \end{bmatrix} = \begin{bmatrix} 1/3 & 1/3 \\ 1 & 1 \end{bmatrix}$$

$$\begin{bmatrix} A_3 & B_3 \\ C_3 & D_3 \end{bmatrix} = \begin{bmatrix} 1/4 & 1/4 \\ 1 & 1 \end{bmatrix}$$

$$\begin{bmatrix} A_4 & B_4 \\ C_4 & D_4 \end{bmatrix} = \begin{bmatrix} \cdot & \cdot \\ 1 & 1 \end{bmatrix}$$

where the "$\cdot$" indicates entries that actually have dimension 0 because the corresponding states do not exist. The corresponding realization is depicted in Fig. 4, and it is not difficult to see that it indeed computes the matrix–vector multiplication $y = Tu$. The above example of the derivation of a "computational network" shows how system theory can be used to obtain efficient algorithms for linear algebra problems (in this case matrix–vector multiplications of lower triangular matrices, but also inversion, Cholesky factorization, etc., is possible) [19].

Although the development of a time-varying state-space theory started in the 1950's (or even earlier), the realization approach presented here is fairly recent, and based on [18]. Some other important approaches that parallel the given presentation can be found in the monograph by Feintuch and Saeks [20], in which a Hilbert resolution space approach is taken, and in recent work by Kamen *et al.* [21], [22], where time-varying systems are put into an algebraic framework of polynomial rings. However, many results, in particular on controllability, detectability, stabilizability, etc., have been discussed by a number of authors without using these specialized mathematical means (see, e.g., Anderson and Moore [23] and references therein, and Gohberg *et al.* [24]) by time-indexing the state-space matrices $\{A, B, C, D\}$ as above.

B. Realization from Input/Output Measurements

In Section II, we assumed that impulse response measurements h_i of the system to be identified were somehow available. In many practical situations, however, instead of the impulse response one is given only a segment of the response of the system to some known nonimpulsive input sequence. A deconvolution operation could be used to determine the impulse response, from which the system can subsequently be identified, but this does not yield a very convincing algorithm because the deconvolution operation itself needs some estimate of the system parameters. We would like to use the Hankel approach of the previous section, where we obtained a realization by applying all possible inputs in the past (inputs that are zero from $t = 0$ on), and determined the range of the corresponding output sequences from $t = 0$ on.

We first look at a slightly different scenario. Suppose we have applied a collection of N independent input sequences $\{\underline{\boldsymbol{u}}_i\}$, $i = 0, \ldots, N-1$, but have measured only a finite segment of the corresponding output sequences $\underline{\boldsymbol{y}}_i$, say from time $t = 0$ to $t = L$, with $d \leq L \ll N$. We denote the known part of each $\underline{\boldsymbol{y}}_i$ by $\boldsymbol{y}_i$, which thus is an $(L+1)$-dimensional vector. Likewise, $\boldsymbol{u}_i$ is defined to be the segment of $\underline{\boldsymbol{u}}_i$ from time $t = 0$ to $t = L$, which will be the only part of each input sequence that will be used in the algorithm. Because the input sequences are not zero from $t = 0$ on, we cannot apply the Hankel approach directly. However, the system is linear, and hence we can construct new input sequences by taking linear combinations of the given sequences, and compute the corresponding output sequences by applying the same linear combinations to the original output sequences. In particular, if we choose the linear combinations such that all known future segments of the input sequence $\boldsymbol{u}_i$ become zero vectors, then we have in fact constructed an input that lives entirely in the past (is zero from $t = 0$), with corresponding output sequences known only from $t = 0$ up to $t = L$. This leads to a transformation

$$\begin{bmatrix} \boldsymbol{u}_0 & \boldsymbol{u}_1 & \cdots & \boldsymbol{u}_{N-1} \\ \boldsymbol{y}_0 & \boldsymbol{y}_1 & \cdots & \boldsymbol{y}_{N-1} \end{bmatrix} Q = \begin{bmatrix} \boldsymbol{u}'_0 & \cdots & \boldsymbol{u}'_L & 0 & \cdots & 0 \\ \boldsymbol{y}'_0 & \cdots & \boldsymbol{y}'_L & \boldsymbol{y}'_{L+1} & \cdots & \boldsymbol{y}'_{N-1} \end{bmatrix} \quad (10)$$

in which Q is an $N \times N$ matrix representing the appropriate linear combinations. Note that for independent $\{\boldsymbol{u}_i\}$, we cannot expect to make all $\boldsymbol{u}'_i$ zero; $L+1$ independent nonzero $\boldsymbol{u}'_i$ will remain. From the analysis in Section II, it is clear that the output vectors $\boldsymbol{y}'_{L+1}, \cdots, \boldsymbol{y}'_{N-1}$ are contained

in the output state space restricted to $t \in [0, L]$; i.e.,

$$[\boldsymbol{y}'_{L+1} \quad \cdots \quad \boldsymbol{y}'_{N-1}] = \begin{bmatrix} C \\ CA \\ \vdots \\ CA^L \end{bmatrix} \cdot X_0 = \mathcal{O}_{L+1} X_0 \tag{11}$$

where X_0 is an unknown $d \times (N - L - 1)$ matrix that can be regarded as containing the initial states (at time $t = 0$) due to the portion of each of the new set of inputs in the (unknown) past. Only if X_0 is of full rank d will the above decomposition determine $\mathcal{O}_{L+1}$ up to a state transformation, and in this case we arrive at a model identification problem that is slightly less restricted than that associated with (7), since in (11) only $\mathcal{O}_{L+1}$ has a shift-invariance property. From this shift invariance, we can obtain A and C as before. The determination of B and D is more involved now, and requires a least squares fit of the given input–output relations (we omit the details) [25].

A few remarks are in place. First, the appropriate transformation Q in (10) can be conveniently computed via a QR (or rather LQ) factorization:

$$\begin{bmatrix} \boldsymbol{u}_0 & \boldsymbol{u}_1 & \cdots & \boldsymbol{u}_{N-1} \\ \boldsymbol{y}_0 & \boldsymbol{y}_1 & \cdots & \boldsymbol{y}_{N-1} \end{bmatrix} = \begin{bmatrix} R_{11} & 0 \\ R_{21} & R_{22} \end{bmatrix} \begin{bmatrix} Q_1^* \\ Q_2^* \end{bmatrix} \tag{12}$$

where the matrices R_{11} and R_{22} are lower triangular matrices of dimension $(L + 1) \times (L + 1)$, and $[Q_1 \quad Q_2]$ are the first $2(L + 1)$ columns of the unitary matrix Q having dimension $N \times N$. Consequently

$$[\boldsymbol{y}'_{L+1} \quad \cdots \quad \boldsymbol{y}'_{N-1}] = [R_{22} \quad 0]$$

and it is seen from (11) that R_{22} must have rank d and a range space that spans that of $\mathcal{O}_{L+1}$. Hence it is shift-invariant, and A can be determined from R_{22} as $A = R_{22}^{(1)+} R_{22}^{(2)}$.

Secondly, when only one input–output sequence is given, of length $N + L$ say, then we can use the time invariance of the system to construct a set of N "independent" input–output sequences of length L, as

$$\begin{bmatrix} u_0 & u_1 & \cdots & u_{N-1} \\ u_1 & u_2 & \cdots & u_N \\ \vdots & & \ddots & \vdots \\ u_L & u_{L+1} & \cdots & u_{N+L-1} \end{bmatrix} \begin{bmatrix} y_0 & y_1 & \cdots & y_{N-1} \\ y_1 & y_2 & \cdots & y_N \\ \vdots & & \ddots & \vdots \\ y_L & y_{L+2} & \cdots & y_{N+L-1} \end{bmatrix}. \tag{13}$$

Finally, it is essential that X_0 in (11) has full rank d. In order to realize this, the set of inputs should be sufficiently "rich." More precisely, we must have

1) the part of the inputs for $t < 0$ should span at least the input state space $\mathcal{H}$ (which is unknown); and
2) $L \geq d$, $N \geq L + 1 + d$.

A set of N inputs $\{\underline{\boldsymbol{u}}_i\}$ that satisfies condition 1) for all possible input state spaces $\mathcal{H}$ of a certain rank is called *persistently exciting*. We will not discuss precise conditions for a set of inputs (or a single input, from which a set of N inputs is constructed by considering shifts as in (13)) to be persistently exciting. In practice, however, if one takes $N \gg d$ and ensures that the span of the past inputs has dimension N, one can be "almost sure" that the rank of X_0 is equal to d. Typically, this will be the case when a stochastic input (zero mean white noise) is applied to the system. Alternatively, one can construct a deterministic input sequence which also has this property.

As a simple example illustrating the above, consider the system described by the first-order difference equation

$$y_k = u_k - \alpha y_{k-1}$$

which has a (trivial) state model in which $x_k = y_{k-1}$ is the state, and where $A = \alpha$ is the pole to be identified. Suppose that we have applied the input sequence $\boldsymbol{u} = [\cdots, 1, 2, 1, 1, \cdots]$, which resulted in the output sequence $\boldsymbol{y} = [\cdots, 1, 2 + \alpha, 1 + 2\alpha + \alpha^2, 1 + \alpha + 2\alpha^2 + \alpha^3, \cdots]$. With $L = 1$ and $N = 3$, the Hankel matrices constructed on the data according to (13) are

$$\begin{bmatrix} U \\ Y \end{bmatrix} = \left[\begin{array}{ccc} 1 & 2 & 1 \\ 2 & 1 & 1 \\ \hline 1 & 1 + 2\alpha & 1 + 2\alpha + \alpha^2 \\ 1 + 2\alpha & 1 + 2\alpha + \alpha^2 & 1 + \alpha + 2\alpha^2 + \alpha^3 \end{array}\right].$$

Taking linear combinations of the columns to zero the third column of U leads to

$$\begin{bmatrix} U' \\ Y' \end{bmatrix} = \left[\begin{array}{ccc} 1 & 2 & 0 \\ 2 & 1 & 0 \\ \hline 1 & 1 + 2\alpha & 5\alpha + 3\alpha^2 \\ 1 + 2\alpha & 1 + 2\alpha + \alpha^2 & 5\alpha^2 + 3\alpha^3 \end{array}\right]$$

so that $[\boldsymbol{y}'_{L+1}]$ can be written as

$$\begin{bmatrix} 1 \\ \alpha \end{bmatrix} (5\alpha + 3\alpha^2)$$

(cf. (11)). The above technique thus yields $C = 1$ and $A = \alpha$.

The material in this section is primarily based on recent work of Verhaegen [26]–[28], whose subspace model identification scheme was in turn inspired by De Moor *et al.* [25], [29], and Moonen [30], [31]. It is also possible to derive a combined stochastic/deterministic identification scheme [32], [33].

C. Direction of Arrival Estimation

The third application arises in antenna array signal processing, and concerns the estimation of the angles of arrival of d narrow-band plane waves impinging upon an antenna array. This is the so-called direction-of-arrival (DOA) estimation problem (see Fig. 5). For simplicity, the narrow-band signals $s_k(t)$ associated with each plane wave are modeled as complex-valued sinusoids $s_k(t) = \hat{s}_k(t) \exp(j2\pi f t)$, where $j = \sqrt{-1}$, $\hat{s}_k(t)$ is the amplitude of the signal (assumed to be slowly time-varying), and f its

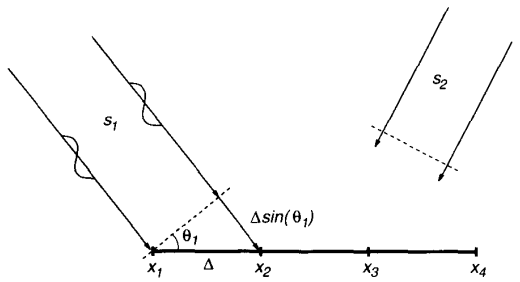


Fig. 5. Direction of arrival estimation. Shown is a uniform linear array consisting of four sensors, and two impinging signals. The angle-of-arrival θ_i of signal s_i is computed from an estimate of the phase shift corresponding to the distance $\Delta \sin(\theta_i)$.

center frequency. The assumption of complex (or *analytic*) signals is supported by the fact that most antenna receivers decompose the received signals into both *in-phase* and *quadrature* components.

An analytic signal model is convenient here since, for narrow-band signals, it allows a time delay to be represented as multiplication by a complex exponential. Consequently, corresponding to each angle of incidence θ_k is a complex constant ϕ_k of unit modulus that represents the phase shift due to the propagation delay τ_k of a plane wave signal between two neighboring sensors of the array separated by a distance Δ. Thus $s_k(t-\tau_k) = s_k(t)\phi_k$, with $\phi_k = \exp(j2\pi f \Delta \sin(\theta_k))$. We will parameterize the DOA problem in ϕ_k rather than θ_k.

Assuming that the sensors and associated receiver hardware are approximately linear, the array output signal at the ith sensor, $x_i(t)$, is given as a weighted sum of the d input signals:

$$x_i(t) = \sum_{k=1}^{d} a_i(\phi_k)\, s_k(t), \quad i = 1, \cdots, L+1 \tag{14}$$

where $a_i(\phi_k)$ represents the response of the ith sensor to a signal arriving from the direction associated with ϕ_k, and we have assumed that there are a total of $L+1$ sensors. Suppose that N samples are taken at time instants $t_1, \cdots, t_N$, and collect the data $x_i(t_j)$ into a $(L+1) \times N$ matrix X with entries $X_{i,j} = x_i(t_j)$. Because of (14), X may be decomposed into the product of a $(L+1) \times d$ matrix $\mathcal{A}(\Phi)$ and a $d \times N$ matrix $\mathcal{S}$

$$X = \mathcal{A}(\Phi)\mathcal{S} \tag{15}$$

where the kth row of $\mathcal{S}$ contains the samples $s_k(t_j)$, $\Phi = \text{diag}(\phi_1, \cdots, \phi_d)$ is a diagonal matrix containing the parameters ϕ_k that are to be identified, $\mathcal{A}(\Phi) = [\boldsymbol{a}(\phi_1) \ \cdots \ \boldsymbol{a}(\phi_d)]$ is a matrix with columns of the form $\boldsymbol{a}(\phi_k) = [a_1(\phi_k) \ \cdots \ a_{L+1}(\phi_k)]^{\mathrm{T}}$, which is the array response vector due to a signal impinging from direction ϕ_k. This vector depends only on the geometrical construction of the array and the directional response of the sensors. For a uniform linear array (ULA) of identical equispaced sensors, $\boldsymbol{a}(\phi)$ is given by $\boldsymbol{a}(\phi) = [1 \ \ \phi \ \ \phi^2 \ \ \cdots \ \ \phi^L]^{\mathrm{T}}$, and $\mathcal{A}(\Phi)$ by

$$\mathcal{A}(\Phi) = \begin{bmatrix} 1 & 1 & \cdots & 1 \\ \phi_1 & \phi_2 & & \phi_d \\ \phi_1^2 & \phi_2^2 & & \phi_d^2 \\ \vdots & & & \vdots \\ \phi_1^L & \phi_2^L & \cdots & \phi_d^L \end{bmatrix}. \tag{16}$$

$\mathcal{A}$ has a structure that is known as Vandermonde structure, and its column space is clearly shift-invariant. Letting $\mathcal{A}^{(1)}$ represent the first L rows of $\mathcal{A}$, and $\mathcal{A}^{(2)}$ the last L rows (and likewise for $X^{(1)}$ and $X^{(2)}$), we have

$$\mathcal{A}^{(2)} = \mathcal{A}^{(1)}\Phi$$

and

$$\begin{aligned} X^{(1)} &= \mathcal{A}^{(1)}\mathcal{S} \\ X^{(2)} &= \mathcal{A}^{(2)}\mathcal{S} = \mathcal{A}^{(1)}\Phi\mathcal{S}. \end{aligned} \tag{17}$$

As before, the equation $\mathcal{A}^{(2)} = \mathcal{A}^{(1)}\Phi$ illustrates the shift-invariant structure present in the array due to the uniform distribution of its (identical) sensors. If no two of the d signals $s_k(t)$ are fully correlated, then $\mathcal{S}$ is of full rank d.[1]

As before, a decomposition of X into minimal rank-d factors is not unique, and will not reveal the Vandermonde structure. We may obtain the decomposition as

$$\begin{aligned} X^{(1)} &= \mathcal{O}^{(1)}\mathcal{C} &&= \mathcal{A}^{(1)}R^{-1}\cdot R\mathcal{S} \\ X^{(2)} &= \mathcal{O}^{(2)}\mathcal{C} = \mathcal{O}^{(1)}A\mathcal{C} &&= \mathcal{A}^{(1)}R^{-1}\cdot R\Phi R^{-1}\cdot R\mathcal{S} \end{aligned}$$

where R is some unknown invertible $d \times d$ matrix, playing the role of a similarity transformation. However, $\mathcal{O}$ is shift-invariant, and A can be determined as in (8): $A = \mathcal{O}^{(1)+}\mathcal{O}^{(2)} = R\Phi R^{-1}$, so that the eigenvalues of A are equal to the ϕ_k.

A related shift structure arises if, instead of a ULA, the array is known to be composed of two identical but otherwise arbitrary subarrays. In this case, $\mathcal{A}(\Phi)$ will satisfy

$$\mathcal{A}(\Phi) = \begin{bmatrix} \mathcal{A}_0 \\ \mathcal{A}_0\Phi \end{bmatrix} \tag{18}$$

for some full rank $\mathcal{A}_0$. This kind of block-shift structure is the parameterization assumed by the so-called ESPRIT algorithm [34]–[36]. Techniques for exploiting this structure are described in Section VI.

The matrix X above will drop rank if either the array response matrix $\mathcal{A}(\Phi)$ or the signal matrix $\mathcal{S}$ has rank less than d. When $\mathcal{A}(\Phi)$ has rank less than d, the array is referred to as being *ambiguous*, and the signal parameters ϕ_k are not identifiable. This corresponds in some sense to an unobservable linear system. This type of rank deficiency can be avoided by proper array design, or in cases where the signal location parameters are restricted to some subset of possible phase delays. For example, the ULA described above is guaranteed to be unambiguous if and only if $\Delta < \lambda/2$, where λ is the wavelength of the narrow-band signals. When $\mathcal{S}$ is rank-deficient, it usually indicates that

[1] Note that perfect sinusoidal signals of the same frequency are the same, up to a difference in phase and amplitude, and consequently $\mathcal{S}$ will have rank 1. The rank condition is satisfied if $\hat{s}_k(t)$ is not constant but slowly time-varying, and the sampling time is long enough.

some subset of the signals are perfectly *coherent*; that is, (at least) one of the signals is just a scaled and delayed version of another signal. This type of situation arises when the *multipath* phenomenon is present, such as occurs when both a direct-path signal and one (or more) reflections are received by the array. Unlike the case of an ambiguous array, the location parameters ϕ_k are often still identifiable when $\mathcal{S}$ is rank deficient [37], [38].

D. Harmonic Retrieval

The relationship between the Hankel decomposition $H = \mathcal{O}\mathcal{C}$ in (6) and the decomposition $X = \mathcal{A}(\Phi)\mathcal{S}$ in (15) is not coincidental. The Hankel matrix decomposition can also be written in terms of Vandermonde matrices if the poles of the system are distinct. Under this condition, recall the partial fraction expansion of the z-transform of the impulse response in (3)

$$\begin{aligned} h(z) &= \sum_0^\infty h_n z^n = r_0 + \sum_{k=1}^{d} \frac{r_k z}{1-\phi_k z} \\ &= r_0 + \sum_{k=1}^{d} r_k z(1 + \phi_k z + \phi_k^2 z^2 + \cdots) \end{aligned} \tag{19}$$

where the ϕ_k are the poles of the system and r_k their residues. The corresponding decomposition of the Hankel matrix is

$$\begin{aligned} H_{L+1,N} &= \sum_{k=1}^{d} \boldsymbol{a}(\phi_k)_{L+1}\, r_k\, \boldsymbol{a}(\phi_k)_N^{\mathrm{T}} \\ &= \begin{bmatrix} 1 & \cdots & 1 \\ \phi_1 & & \phi_d \\ \phi_1^2 & & \phi_d^2 \\ \vdots & & \vdots \\ \phi_1^L & \cdots & \phi_d^L \end{bmatrix} \begin{bmatrix} r_1 & & \\ & \ddots & \\ & & r_d \end{bmatrix} \\ &\quad \cdot \begin{bmatrix} 1 & \phi_1 & \phi_1^2 & \cdots & \phi_1^{N-1} \\ \vdots & & & & \vdots \\ 1 & \phi_d & \phi_d^2 & \cdots & \phi_d^{N-1} \end{bmatrix} \\ &= \mathcal{A}_{L+1}(\Phi)\,\mathcal{S}_N \end{aligned} \tag{20}$$

with $\boldsymbol{a}(\phi_k)_{L+1}^{\mathrm{T}} = [1 \;\; \phi_k \;\; \phi_k^2 \;\; \cdots \;\; \phi_k^L]$, $\mathcal{A} = [\boldsymbol{a}(\phi_1) \;\; \cdots \;\; \boldsymbol{a}(\phi_d)]$, and $\mathcal{S}$ equal to the product of the last two matrices in the decomposition. The same decomposition would have been obtained from (7) starting from any realization $\{A, B, C, D\}$ by applying a state similarity transformation that diagonalizes the A-matrix, $A = R\Phi R^{-1}$, for an appropriate choice of R. Letting $\mathcal{O}R =: \mathcal{A}$ and $R^{-1}\mathcal{C} =: \mathcal{S}$ will map (7) to (20).

Another connection between the models of (6) and (17) can be made by means of the harmonic retrieval problem. Assume for the moment that we have the following realization of a linear system with distinct poles:

$$\begin{aligned} A &= \Phi & B &= [r_1 \;\cdots\; r_d]^{\mathrm{T}} \\ C &= [1 \;\cdots\; 1] & D &= r_0 \end{aligned}$$

where we allow the ϕ_k and r_k to possibly be complex. If we let $\phi_k = e^{\alpha_k + j\omega_k}$, the time-domain version of the impulse response of (19) can be written as

$$h_n = \sum_{k=1}^{d} r_k e^{(\alpha_k + j\omega_k)n} \tag{21}$$

which is just a sum of d damped exponential signals. Thus the problem of determining the poles of a linear system from observations of its impulse response can be recast as one of estimating the frequencies and decay factors of multiple exponential signals. This latter problem is referred to as *harmonic retrieval*, and has been studied by researchers for many years in fields as diverse as economics, zoology, and physics, not to mention engineering. One of the earliest written accounts of such work was given by the Baron de Prony in the late eighteenth century [39]. Comparing (20) with (15), we see that the matrix $\mathcal{A}(\Phi)$ defined here is analogous to the array manifold in the DOA estimation problem, and will be "unambiguous," (i.e., full rank d) if $L + 1 > d$ and $\omega_k < \pi$. When $N - 1 > d$, the Nyquist assumption $\omega_k < \pi$ also can be shown to guarantee that $\mathcal{S}$ is full rank d.

IV. Singular Value Decomposition

In the previous section, the notions of subspace, column space, rank, and factorization of matrices have been introduced conceptually, and it was noted that the singular value decomposition (SVD) of matrices is a robust tool for computing them. In sections to follow we will make extensive use of this tool, and therefore we shall take a closer look at it in this section. For a more detailed account (and an overview of algorithms for its computation) we refer to [15]. Tutorial information as well as related technical papers on the subject of SVD and signal processing are provided by [4] and the series [40], [41].

A. Subspaces

Starting with a given matrix X of size $L \times N$ and with entries in $\mathbb{C}$, one may want to know how many columns (rows) of this matrix are nonparallel or independent of each other. We will assume throughout that the dimensions L and N are finite (however, most of the results will still hold when the dimensions are not finite, provided X is a so-called *compact* operator, i.e., when the sum of its squared entries is bounded). If there are $d \le L \le N$ independent columns in X, then this matrix is said to have a d-dimensional range or column space, which is a subspace of the L-dimensional Euclidian space $\mathbb{C}^L$. The rank of the matrix is the dimension of this subspace. If $d = L$, then the matrix is of full rank, and for $d < L$ it is rank-deficient. Now $\mathbb{C}^L$ is spanned by the columns of any unitary matrix in $\mathbb{C}^{L\times L}$, the Euclidean space of square, complex-valued L-dimensional matrices. The same holds for $\mathbb{C}^N$ of which the row space of X is a d-dimensional subspace: the columns of any $N \times N$ unitary matrix in $\mathbb{C}^{N\times N}$ span the vector space $\mathbb{C}^N$. Assuming $d \le L \le N$, we can choose a unitary U such that the d-dimensional column space of X is spanned

by a subset of d columns of U, say the first d columns, which together form a matrix $\hat{U}$:

$$U = \overset{\overset{d}{\leftrightarrow}}{{}_L\updownarrow\left(\hat{U}\right.} \quad \overset{\overset{L-d}{\leftrightarrow}}{\left.\hat{U}^{\perp}\right)}.$$

Since U is a unitary matrix, we shall have

1) From $U^*U = I_L$:

$$(a)\ \hat{U}^*\hat{U} = I_d$$
$$(b)\ \hat{U}^*\hat{U}^{\perp} = 0$$
$$(c)\ (\hat{U}^{\perp})^*\hat{U}^{\perp} = I_{L-d}\,.$$

2) From $UU^* = I_L$:

$$(d)\ \hat{U}\hat{U}^* + \hat{U}^{\perp}(\hat{U}^{\perp})^* = I_L$$

where I_d is the identity matrix of order d, and similarly for I_L and I_{L-d}. Relations (a)–(d) tell us that any vector $\boldsymbol{x} \in \mathbb{C}^L$ can be decomposed into two mutually orthogonal vectors $\hat{\boldsymbol{x}}$ and $\hat{\boldsymbol{x}}^{\perp}$ in the spaces spanned by the columns of $\hat{U}$ and $\hat{U}^{\perp}$, respectively. These two spaces are d-dimensional and $(L-d)$-dimensional orthogonal subspaces in $\mathbb{C}^L$, and their direct sum is equal to $\mathbb{C}^L$. Therefore, the orthogonal complement in $\mathbb{C}^L$ of the column space of X is spanned by the columns of the matrix $\hat{U}^{\perp}$. The matrices $\hat{U}\hat{U}^* = \Pi_c$ and $\hat{U}^{\perp}(\hat{U}^{\perp})^* = \Pi_c^{\perp}$ in the above relation (d) are the orthogonal projectors onto the column space of X and its orthogonal complement in $\mathbb{C}^L$, respectively. That is, $\hat{\boldsymbol{x}} = \Pi_c\boldsymbol{x}$ and $\hat{\boldsymbol{x}}^{\perp} = \Pi_c^{\perp}\boldsymbol{x}$.

The unitary matrix V can be similarly decomposed:

$$V = \overset{\overset{d}{\leftrightarrow}}{{}_N\updownarrow\left(\hat{V}\right.} \quad \overset{\overset{N-d}{\leftrightarrow}}{\left.\hat{V}^{\perp}\right)}.$$

Here, the matrices $\hat{V}\hat{V}^* = \Pi_r$ and $\hat{V}^{\perp}(\hat{V}^{\perp})^* = \Pi_r^{\perp}$ are orthogonal projectors onto the original subspaces in $\mathbb{C}^N$ spanned by the columns of $\hat{V}$ and $\hat{V}^{\perp}$, respectively. The columns of $\hat{V}^{\perp}$ span the kernel of X, i.e., the space of input vectors $\boldsymbol{x}$ for which $X\boldsymbol{x} = 0$.

B. SVD

In terms of the above discussion of subspaces, the singular value decomposition of the $L \times N$ matrix X, which we assume to have rank d, is obtained by making a certain well-defined choice for U and V, which then gives rise to the following decomposition [15]:

$$X = [\hat{U} \quad \hat{U}^{\perp}]\,\Sigma\begin{bmatrix}\hat{V}^* \\ (\hat{V}^{\perp})^*\end{bmatrix}$$

where Σ is an $L \times N$ diagonal matrix containing the singular values σ_i of X. These are positive numbers ordered such that

$$\sigma_1 \geq \sigma_2 \geq \cdots \geq \sigma_d > \sigma_{d+1} = \cdots = \sigma_L = 0\,.$$

Note that only d singular values are nonzero. The d columns of $\hat{U}$ corresponding to these nonzero singular values span the column space of X and are called the left singular vectors. Similarly, the d columns of $\hat{V}$ are called the right singular vectors and span the row space of X (or the column space of X^*) . In terms of these (sometimes much) smaller matrices, the SVD of X can also be written in "economy" size

$$X = \hat{U}\hat{\Sigma}\hat{V}^* \tag{22}$$

where $\hat{\Sigma}$ is a $d \times d$ diagonal matrix containing $\sigma_1, \cdots, \sigma_d$. This form of the SVD better reveals that X is actually of rank d: it is constructed from a product of rank-d matrices.

The SVD of X makes the various spaces (range and kernel) associated with X explicit. So does any decomposition of X as $X = \hat{U}E_x\hat{V}^*$, where $\hat{U}$ and $\hat{V}$ are any matrices whose columns span the column and row spaces of X, respectively, and where E_x is an invertible $d \times d$ matrix. The property that makes the SVD special is the fact that E_x is a diagonal matrix, so that a decoupling is obtained: with $\boldsymbol{u}_i$ the ith column of U, and $\boldsymbol{v}_i$ likewise for V, X can be written as a sum of rank-1 isometric matrices $\boldsymbol{u}_i\boldsymbol{v}_i^*$, scaled by σ_i:

$$X = \sum_{i=1}^{d} \sigma_i(\boldsymbol{u}_i\boldsymbol{v}_i^*)$$

and we also have

$$\sigma_i\boldsymbol{u}_i = X\boldsymbol{v}_i\,, \qquad \sigma_i\boldsymbol{v}_i = X^*\boldsymbol{u}_i\,.$$

This makes it possible to rank the vectors in the column space and row space of X: the most important direction in the column space is $\boldsymbol{u}_1$, with scale σ_1, and is reached by applying X to the vector $\boldsymbol{v}_1$. The second most important direction is $\boldsymbol{u}_2$, etc. This ranking will in turn lead to optimal low-rank approximants of X (see below). In the mapping $a \in \mathbb{C}^N \rightarrow b \in \mathbb{C}^L : b = Xa$, b will automatically be a vector in the column range of X, and will be nonzero if and only if a has a component in the row space of X; i.e., if and only if $\Pi_r a$ is nonzero. On the other hand, b will be identically zero if and only if a is orthogonal to the row space of X. Therefore, the space spanned by the vectors $\boldsymbol{v}_{d+1}, \cdots, \boldsymbol{v}_N$ in $\hat{V}^{\perp}$ is called the null space (or kernel) of X. Vectors a in this space are mapped to zero by one of the zero singular values of X. The SVD of X reveals the behavior of the map $b = Xa$: a is rotated in N-space (by V^*), then scaled (by the entries of Σ: $L-d$ components are projected to zero), and finally rotated in L-space (by U) to give b.

C. The Effect of Noise

Suppose that X is an $L \times N$ matrix with rank $d < L$. As before, denote the SVD of X as $X = U\Sigma V^* = \hat{U}\hat{\Sigma}\hat{V}^*$. In this subsection, we will briefly study the effect of adding noise to X on its SVD. The perturbation theory of the SVD is partially based on the link of the SVD with eigenvalue decompositions:

$$X = U\Sigma V^* \quad \Rightarrow \quad XX^* = U\Sigma^2U^*$$

so that the singular values of X are the positive square roots of the eigenvalues of XX^*, while its left singular values are the eigenvectors of XX^*. Suppose that X is perturbed by some noise matrix $\mathcal{V}$: $X' = X + \mathcal{V}$. We first consider the case where the entries of $\mathcal{V}$ are generated by uncorrelated, zero mean, white-noise processes with variance σ^2, so that the variance $E(\mathcal{V}\mathcal{V}^*)$ is asymptotically (for $N \to \infty$) given by $E(\mathcal{V}\mathcal{V}^*/N) = \sigma^2 I_L$. Under the same conditions

$$E(X'X'^*/N) = E(XX^*/N) + \sigma^2 I_L$$

so that, for large N, the SVD of X' is given by

$$X' \approx U(\Sigma^2 + N\sigma^2 I)^{1/2} V'^*$$

for some unitary matrix V'. This expression shows that, for large N and small σ, the singular values of X' increase by an amount approximately equal to $\sigma\sqrt{N}$, while the left singular vectors of X remain the same. X' is now of full rank, and its $L - d$ smallest singular values are no longer zero, but equal to $\sigma\sqrt{N}$. In theory, we can recover XX^* by subtracting $N\sigma^2 I$ from $(\Sigma')^2$. This should set the $(L - d)$ smallest singular values back to zero. The range space of X, as estimated from X', is spanned by $\hat{U}$, the left singular vectors corresponding to the d largest singular values of X'. It is not possible to recover $\hat{V}$ (or X), because the length of the columns of $\hat{V}$ is equal to N, and hence these vectors do not participate in the averaging effect of increasing N.

For more general $\mathcal{V}$, and in case N is not extremely large, one can show that the singular values of X are raised by an amount on the order of $\| \mathcal{V} \|$, the largest singular value of $\mathcal{V}$. However, in this case the singular vectors are also perturbed. The amount of the perturbation in the *subspace* which they span can be estimated (see, e.g., [42], [43]), and is again in the order of $\| \mathcal{V} \|$. The effect on the singular vectors themselves can be much larger if the corresponding singular values are close [42]. Summarizing, the singular values and the subspace spanned by the left singular vectors are (for reasonably large N) relatively insensitive to added perturbations on the entries of the matrix, and hence the SVD is numerically reliable and robust. The SVD thus provides a good estimate of the numerical rank of a matrix, and is useful for quantifying how "close" a matrix is to being low-rank.

The "noise threshold" depends on the smallest singular value of the original matrix. This singular value is related to the smallest vector that can be constructed with linear combinations of the columns of the matrix, or the smallest distance of one column of the matrix to the column range of the remaining columns. Obviously, it will be small when the columns are more or less "aligned," as displayed in Fig. 6. This figure shows the construction of the left singular vectors of a matrix $X = [\boldsymbol{x}_1 \ \boldsymbol{x}_2]$, whose columns $\boldsymbol{x}_1$ and $\boldsymbol{x}_2$ are of equal length. The largest singular vector $\boldsymbol{u}_1$ is in the direction of the sum of $\boldsymbol{x}_1$ and $\boldsymbol{x}_2$, i.e., the "common" direction of the two vectors, and the corresponding singular value σ_1 is equal to $\sigma_1 = \| \boldsymbol{x}_1 + \boldsymbol{x}_2 \| / \sqrt{2}$. On the other hand, the smallest singular vector $\boldsymbol{u}_2$ is dependent on the difference $\boldsymbol{x}_2 - \boldsymbol{x}_1$, as is its corresponding singular value: $\sigma_2 = \| \boldsymbol{x}_2 - \boldsymbol{x}_1 \| / \sqrt{2}$. If $\boldsymbol{x}_1$ and $\boldsymbol{x}_2$ become more aligned, then σ_2 will be smaller and X will be closer to a singular matrix. Clearly, $\boldsymbol{u}_2$ is the most sensitive direction for perturbations on $\boldsymbol{x}_1$ and $\boldsymbol{x}_2$.

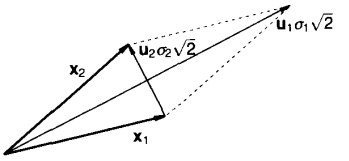


Fig. 6. Construction of the left singular vectors and values of the matrix $X = [\boldsymbol{x}_1 \ \boldsymbol{x}_2]$, where $\boldsymbol{x}_1$ and $\boldsymbol{x}_2$ have equal lengths.

The relevance of this observation is that the resolution of subspace-based parameter estimation algorithms depends on the smallest singular value of the matrix of observations, in relation to the noise level. For example, in the previous section, the observation matrix consisted of linear combinations of vectors of the form $\boldsymbol{a}(\phi) = [1 \ \phi \ \phi^2 \ \cdots \ \phi^{L-1}]^{\mathrm{T}}$. If two directions, or two poles, are close together, then $\phi_1 \approx \phi_2$ and $\boldsymbol{a}(\phi_1)$ points in about the same direction as $\boldsymbol{a}(\phi_2)$, which will be the direction of $\boldsymbol{u}_1$. The smallest singular value, σ_2, is dependent on the difference of the directions of $\boldsymbol{a}(\phi_1)$ and $\boldsymbol{a}(\phi_2)$.

With a noise matrix $\mathcal{V}$ added, detecting the presence of two signals will in general become difficult if σ_2 is approximately the same or smaller than $\| \mathcal{V} \|$, the noise level. This is because the structure of $\mathcal{V}$ determines how much σ_2 is increased: $\sigma_2^2 \leq (\sigma_2')^2 \leq \sigma_2^2 + \| \mathcal{V} \|^2$, and because the second direction is only visible if $\sigma_2' > \| \mathcal{V} \|$. Note that in the commonly assumed case where $\mathcal{V}$ is generated by independent identically distributed noise processes such that $E(\mathcal{V}\mathcal{V}^*/N) = \sigma^2 I$, then, for large enough N, all of the singular values squared increase by the same amount $\| \mathcal{V} \|^2$. In such cases, $\sigma_2' > \| \mathcal{V} \|$ automatically, and detection of the second signal is always possible. It is also important to note that the smallest singular value is strongly dependent on the value of L, the length of the observation vectors. If L is increased, then the difference between $\boldsymbol{a}(\phi_1)$ and $\boldsymbol{a}(\phi_2)$ becomes more pronounced, so that σ_2 becomes larger and the resolution increases. This effect is stronger than the effect of increasing N, the number of observation vectors. In the latter case, the purpose is to average out the noise.

For illustration, consider the following small numerical experiment. Let $\phi_1 = 1$, $\phi_2 = \exp(j\pi \cdot 0.1)$, and construct matrices $X_{L,N}$ from unitary linear combinations of the columns of $[\boldsymbol{a}(\phi_1) \ \ \boldsymbol{a}(\phi_2)]$. For $L, N \geq 2$, these matrices have rank 2. The two nonzero singular values of $X_{L,N}$ for some values of L, N are given in Table 1. It is seen that doubling L almost triples the smallest singular value, whereas doubling N only increases the singular values by a factor $\sqrt{2}$, which is because the matrices have larger size. In the latter case, the ratio between σ_2 and the noise level is not increased because the perturbation matrix would also have twice its original size, which leads to an increase in the noise level of the same factor $\sqrt{2}$.

Table 1 Singular Values of $X_{L,N}$.

$L = 3$	$\sigma_1 = 3.44$	$L = 3$	$\sigma_1 = 4.86$
$N = 3$	$\sigma_2 = 0.44$	$N = 6$	$\sigma_2 = 0.63$
$L = 6$	$\sigma_1 = 4.73$		
$N = 3$	$\sigma_2 = 1.29$		

D. Pseudo-Inverse

Consider a rank-d $L \times N$ matrix X. In general, since X may be rank-deficient or nonsquare, the inverse of X does not exist; i.e., for a given vector b, we cannot always find a vector a such that $b = Xa$. A related notion is the (Moore–Penrose) pseudo-inverse of X, denoted by X^+, which can be defined in terms of the "economy size" SVD of X (see (22)) as

$$X^+ = \hat{V}\hat{\Sigma}^{-1}\hat{U} .$$

This pseudo-inverse satisfies the properties

1. $XX^+X = X$
2. $X^+XX^+ = X^+$
3. $XX^+ = \Pi_c$
4. $X^+X = \Pi_r$

which constitute the Moore–Penrose inverse in the traditional way. These equations show that, in order to make the problem $b = Xa$ solvable, a solution can be forced to an approximate problem by projecting b onto the column space of X: $b' = \Pi_c b$, after which $b' = Xa$ has solution $a = X^+b'$. The projection can also be done implicitly by just taking $a = X^+b$: from properties 1 and 3 of the list above, we have that $a = X^+b' = X^+XX^+b = X^+b$. It can be shown that this solution a is the solution of the (least squares) minimization problem

$$\min_a \| b - Xa \|_2^2$$

where a is chosen to have minimal norm if there is more than one solution (the latter requirement translates to $a = \Pi_r a$).

E. LS and TLS Approximations

Suppose that X has full rank L. In this case, $\sigma_{d+1}, \cdots, \sigma_L$ are nonzero, and the SVD of X can be written as

$$X = U\Sigma V^* = \hat{U}\hat{\Sigma}\hat{V}^* + \hat{U}^\perp\hat{\Sigma}^\perp\hat{V}^\perp$$

where $\hat{U}$ contains the first d left singular vectors of X, corresponding to the d largest singular values which are collected in $\hat{\Sigma}$. $\hat{\Sigma}^\perp$ contains the $L - d$ remaining singular values, which are now nonzero. $\hat{U}$ contains the d "most important" vectors (directions) in the column space of X. Hence, a rank-d approximation $\hat{X}$ of X is obtained by putting

$$\hat{X} = \hat{U}\hat{\Sigma}\hat{V}^* = \Pi_{\hat{U}} X \Pi_{\hat{V}} \tag{23}$$

where $\Pi_{\hat{U}} = \hat{U}\hat{U}^*$ and $\Pi_{\hat{V}} = \hat{V}\hat{V}^*$ are the projectors onto the approximated column space and row space of X, respectively. If X' is any rank-d $L \times N$ matrix, then it can be shown that $\hat{X}$ is the rank-d approximation of X that minimizes $\| X - X' \|_F$, the Frobenius norm of the difference $E = X - X'$. The Frobenius norm of a matrix is the sum of the squares of its entries, and can be shown to be equal to the sum of the squares of its singular values (because this norm is "rotationally invariant"). $\hat{X}$ is called the rank-d Least Squares (LS) approximation to X: it retains the d most important singular values and vectors of X, and sets the remaining $L - d$ singular values to zero. Hence $\| E \|_F^2 = \sigma_{d+1}^2 + \cdots + \sigma_L^2$.

A typical LS application is the following. Suppose that a vector b is given, and we want to find a vector a such that $b = Xa$. We saw above that a (least squares min-norm) solution is obtained by setting $a = X^+b$. However, since $X^+ = V\Sigma^+U^*$, small singular values of X play an important role in X^+: the pseudo-inverse of the full-rank matrix can lead to numerical instabilities. A more reliable solution is obtained by setting the small singular values of X equal to zero, thus obtaining an LS approximation $\hat{X}$ of X. The vector a is then obtained by computing a as the least squares min-norm solution of $b = \hat{X}a$ (that is, $a = \hat{X}^+b$).

Now, suppose that instead of a single vector b we are given an $(L \times N)$-dimensional matrix Y, the columns of which are not all in the column space of the matrix $\hat{X}$. We want to force solutions to $XA = Y$. Clearly, we can use a LS approximation $\hat{Y} = \Pi_{\hat{U}} Y$ to force the columns of $\hat{Y}$ to be in the d-dimensional column space of $\hat{X}$. This is reminiscent to the LS application above, but just one way to arrive at $\hat{X}$ and $\hat{Y}$ having a common column space. There is an other way, called Total Least Squares (TLS) which is effectively described as projecting both X and Y onto some d-dimensional subspace that lies between them, and that is "closest" to the column spaces of the two matrices. To implement this method, we compute the SVD

$$\begin{aligned}[X \quad Y] &= [\hat{U} \quad \hat{U}^\perp]\,\Sigma \begin{bmatrix} \hat{V}^* \\ (\hat{V}^\perp)^* \end{bmatrix} \\ &= \hat{U}\hat{\Sigma}[\hat{V}_1^* \quad \hat{V}_2^*] + \hat{U}^\perp\hat{\Sigma}^\perp(\hat{V}^\perp)^*\end{aligned}$$

and take the TLS (column space) approximations to be $\hat{X} = \Pi_c X = \hat{U}\hat{\Sigma}\hat{V}_1^*$ and $\hat{Y} = \Pi_c Y = \hat{U}\hat{\Sigma}\hat{V}_2^*$, where $\hat{V}_1$ and $\hat{V}_2$ are the partitions of $\hat{V}$ corresponding to X and Y, respectively. $\hat{X}$ and $\hat{Y}$ are in fact solutions to

$$\min_{[\hat{X}\ \hat{Y}]_{\text{rank}\,d}} \| [X \ \ Y] - [\hat{X} \ \ \hat{Y}] \|_F^2$$

and A satisfying $\hat{X}A = \hat{Y}$ is obtained as $A = \hat{X}^+\hat{Y}$. This A is the TLS solution of $XA \approx Y$.

F. The Matrix Pencil Problem

To close this section we consider the following eigenvalue problem. Let X and Y be two (full-rank) matrices of dimension $L \times N$ ($L \leq N$), and let λ be a complex scalar. The matrix pencil problem is to determine values of λ for which the rank of the matrix $Y - \lambda X$ is $L - 1$ instead of L. $Y - \lambda X$ is called a matrix pencil, and those special values of λ are called the rank-reducing numbers of

the pencil. When X and Y are not of full rank, the matrix pencil problem is to find the values of λ for which the rank of the pencil drops one in comparison to the usual rank of the pencil.

In cases where X and Y are full-rank square $L \times L$ matrices, the matrix pencil problem reduces to an ordinary eigenvalue problem. There are L rank-reducing numbers $\lambda_1, \cdots, \lambda_L$, and they are known as the generalized eigenvalues (GE's) of the matrix pair (Y, X). The GE's of (Y, X) are those values of λ for which there exists a nontrivial vector x such that $Yx = \lambda Xx$. Since, under the present assumptions, X is invertible, these are just the solutions to the ordinary eigenvalue problem $(X^{-1}Y)x = \lambda x$.

We now turn to the more general problem that will be encountered in the next section, where X and Y are rank-d $L \times N$ matrices. For convenience, we require X and Y to have the same column space and row space. These amount to conditions for the existence of nontrivial λ. If X and Y were to have disjoint column spaces, then the rank of $Y - \lambda X$ can only decrease if the rank of λX decreases, i.e., if $\lambda = 0$. A similar result holds for the row spaces. We will show that the solution of the matrix pencil problem can be given in terms of the pseudo-inverses of X and Y as introduced before. Call $\hat{U}_x\hat{\Sigma}_x\hat{V}_x^*$ and $\hat{U}_y\hat{\Sigma}_y\hat{V}_y^*$ the "economy-size" SVD's of X and Y, respectively. Since by assumption X and Y span the same column and row spaces, we can express Y in terms of $\hat{U}_x$ and $\hat{V}_x$: say $Y = \hat{U}_x E_y \hat{V}_x^*$, with $E_y = \hat{U}_x^* Y \hat{V}_x$ a $d \times d$ matrix. Hence

$$Y - \lambda X = \hat{U}_x(E_y - \lambda\hat{\Sigma}_x)\hat{V}_x^*$$

and thus the problem is reduced to the square pencil problem: the rank-reducing numbers of the pencil $X - \lambda Y$ are given by the d eigenvalues of $\hat{\Sigma}_x^{-1}E_y$. It can be shown that these solutions are precisely the nonzero entries in $\text{eig}\,(X^+Y)$ or $\text{eig}\,(YX^+)$. Indeed

$$\begin{aligned} X^+Y &= \hat{V}_x\hat{\Sigma}_x^{-1}\hat{U}_x^* \cdot \hat{U}_x E_y \hat{V}_x^* \\ &= \hat{V}_x \cdot \hat{\Sigma}_x^{-1}E_y \cdot \hat{V}_x^* . \end{aligned}$$

From the property that the nonzero eigenvalues of the product (AB) of two matrices A and B are equal to the nonzero eigenvalues of (BA), the result follows.

V. Overview of Identification Schemes

A. The Model

In the realization theory of Sections II and III, we have seen that there are two Hankel matrix decompositions that are in fact equivalent if the system poles are distinct:

$$\begin{aligned} &(1)\; H = \mathcal{O}\mathcal{C}\,, \\ &(2)\; H = \mathcal{A}(\Phi)\,\mathcal{S} \end{aligned}$$

in which $\mathcal{A}$ has a Vandermonde structure parametrized by the diagonal matrix Φ with entries ϕ_i, and in which $\mathcal{O}, \mathcal{C}, \mathcal{A}, \mathcal{S}$ are shift-invariant. In fact, the second description is a special case of the first. The purpose of identification is 1) to find the pole locations (or equivalently Φ), and 2) to determine a matching state-space model (i.e., to find the corresponding zeros of the system). In the input–output identification application, H is not a Hankel matrix but its column space is still shift-invariant. In the DOA application, the second description given above is more natural since $\mathcal{A}$ corresponds to the array response matrix and $\mathcal{S}$ to the incoming signals. $\mathcal{S}$ has a shift-invariant structure only if the sampling period is constant. The purpose in DOA estimation is 1) to find the directions of arrival (or equivalently Φ), and 2) to reconstruct the signal matrix $\mathcal{S}$ (signal copy). For the sake of discussing these applications within a unified framework, and to present algorithms that are valid for both system identification and DOA estimation, we will use the description $[H = \mathcal{A}(\Phi)\mathcal{S}]$ in most of the remainder of the paper, and focus on finding Φ. Once Φ, and hence $\mathcal{A}(\Phi)$, is known, it is a straightforward matter to determine a corresponding $\mathcal{S}$ from H, e.g., by setting $\mathcal{S} = \mathcal{A}^+H$.

The algorithms in Section II were based on noise-free conditions. In general, however, H is corrupted by noise, which is assumed to be additive:

$$H = \mathcal{A}(\Phi)\,\mathcal{S} + \mathcal{V}\,.$$

The noise incorporates all undesired components of the data. Depending on the problem at hand and on the chosen solution strategy, the noise is assumed to be either stationary zero-mean white (as in the system identification and DOA problem), or to encompass unwanted higher order components of an actual system response (modes to be neglected in model reduction problems).

The problem we will consider in the remaining part of this paper thus reads as follows: Given a matrix H which contains noise-corrupted observations of a system, determine a $d \times d$ diagonal matrix Φ using the model

$$\begin{aligned} H &= \mathcal{A}(\Phi)\,\mathcal{S} \;\; + \mathcal{V}\,, \\ \mathcal{A}(\Phi) &= [\,\boldsymbol{a}(\phi_1) \;\cdots\; \boldsymbol{a}(\phi_d)\,] \\ \boldsymbol{a}(\phi) &= [\,1 \quad \phi \quad \phi^2 \quad \cdots \quad \phi^L\,]^{\mathrm{T}} \end{aligned} \tag{24}$$

in which H is of size $(L+1) \times N$ with $N \geq L \geq d$, $\mathcal{A}$ and $\mathcal{S}$ are of full rank d, and the matrix $\mathcal{V}$ represents additive noise. In this problem statement, H need not be Hankel, and hence no shift-invariant structure in $\mathcal{S}$ is presumed. The column space of $\mathcal{A}\mathcal{S}$ is referred to as the *signal subspace* (which is the output state space in system theory), and its orthogonal complement is referred to as the *noise subspace*. The presence of the noise term means that H will actually be of full rank.

An important issue that we have not dealt with thus far is that of model order determination. With white noise present and N approaching infinity, the extra singular values due to the noise are all the same and presumably small, and d can be found by simply counting the multiplicity of the smallest singular value of H and subtracting from $L + 1$. However, with probability one, for finite N none of the singular values of H will be repeated, and hence some other method is required to estimate d. Put simply, the

strategy is to look for a break in the pattern of singular values of H, attributing the larger ones to the signal and the small ones to the noise. The detection of such a break has been well studied and a number of techniques have been developed, most of them being based on the asymptotic distribution of the covariance matrix related to H under the assumption of white Gaussian noise. These include the classical sequential hypothesis test [44]–[46], Akaike's Information Criterion (AIC) [47], Rissanen's Minimum Description Length (MDL) principle [48], [49], and the refinements of Zhao *et al.* [50]. Specific applications to DOA estimation have been studied in [51]–[55]. It is beyond the scope of this paper to study the model order determination problem in any detail, so we will just assume in what follows that d has been correctly determined by some method.

B. Solution Outline

A number of strategies for solving the identification problem in (24) have been proposed. They differ primarily in the degree of structure that is imposed on the solution.

1) *Subspace Fitting:* These methods seek to match the data with the "true" model; i.e., they minimize $\| H - \mathcal{A}(\Phi)\mathcal{S} \|_F^2$ in the Frobenius norm over all possible $\mathcal{S}$ and Vandermonde matrices $\mathcal{A}(\Phi)$ of rank d. Equivalently, they may be thought of as finding a model (with shift-invariance properties) whose column vectors are most orthogonal to the estimated noise subspace of the data. Weighted versions of this minimization have recently been proposed (Weighted Subspace Fitting, MODE) which provide minimum variance parameter estimates.
2) *Single Shift-Invariant Methods:* In contrast to the Subspace Fitting techniques, these methods impose only a single shift-invariance property on the data, in the sense that the observation matrix H is partitioned into two matrices $X = H^{(1)}$ and $Y = H^{(2)}$, with X containing the first L rows of H, and Y the last L rows. The problem is then recast as one of finding Φ from

$$\begin{aligned} X &= \mathcal{A}_L \mathcal{S}_N + \mathcal{V}_1 \\ Y &= \mathcal{A}_L \Phi \mathcal{S}_N + \mathcal{V}_2 . \end{aligned} \tag{25}$$

We will describe a number of methods (e.g., TAM, ESPRIT) that determine Φ from (X, Y) using only the above decomposition (25), hence ignoring any further (shift-invariant) structure that $\mathcal{A}$ or $\mathcal{S}$ might possess. These methods are thus valid for any application in which an X and Y which obey (25) are somehow obtained, but for which no further information on the underlying structure is known. In particular, in the ESPRIT algorithm for DOA estimation, X and Y typically contain data from two identical sensor *subarrays*. H is then obtained by stacking X and Y, thus having size $2L \times N$.
3) *Orthogonal Vector Methods:* This class of techniques is related to the above two strategies, and can be thought of as intermediary between them. These methods are also based on the shift-invariant structure of (25), but they can equivalently be described as methods that find vectors orthogonal to a particular vector selected from the noise subspace (see the discussion below).

Subspace fitting techniques are described inSection VIII. In these techniques, the problem is to determine a d-dimensional column space (range) of $\mathcal{A}$ that has the required Vandermonde-like structure and is closest to the column space of H. By ignoring any (shift-invariant) structure that $\mathcal{S}$ might possess, the minimization is linear in the parameters of $\mathcal{S}$. Consequently, the problem can be made more compact by deriving from H a rank-d signal subspace, and then finding a rank-d matrix $\mathcal{A}$ with Vandermonde structure whose column space is as close as possible to the signal subspace (or equivalently, which is as orthogonal as possible to the noise subspace). Though Subspace Fitting techniques can provide estimates of minimum variance, such techniques are more difficult to implement since in general they require a multidimensional (gradient) search over the parameter space. This drawback is mitigated by the fact that the computationally efficient Single Shift-Invariant methods can be used to obtain an accurate starting point for the search.

Approaches to the Single Shift-Invariance problem (25) are motivated by the exact relationships present in the noise-free case. They give rise to the matrix pencil techniques that we already have encountered in Section IV, in which the pencil $Y - \lambda X$ is studied for varying values of λ. Without noise, it readily follows from the structure of (25) that the diagonal entries of Φ are the rank-reducing numbers of the pencil $Y - \lambda X$, i.e., those values of λ for which the pencil drops rank. This is because $Y - \lambda X = \mathcal{A}(\Phi - \lambda I)\mathcal{S}$. A slightly more general way to describe these methods is by defining an $L \times L$ matrix F that satisfies $FX = Y$. Since in the noise-free case X and Y are of rank d, F is not unique; it can have rank anywhere from d to L. For any of the possible choices of F, it can be shown that d of the eigenvalues of F are equal to the entries of the diagonal matrix Φ. Indeed, since $\mathcal{A}$ and $\mathcal{S}$ are rectangular matrices of full rank d, they have pseudo-inverses $\mathcal{A}^+$, $\mathcal{S}^+$ such that

$$\mathcal{A}^+\mathcal{A} = I_d \qquad \mathcal{S}\mathcal{S}^+ = I_d$$

(dropping subscripts for ease of notation) and hence the equation $FX = Y$ results in

$$F\mathcal{A}\mathcal{S} = \mathcal{A}\Phi\mathcal{S} \qquad \Rightarrow \qquad \Phi = \mathcal{A}^+ F \mathcal{A} .$$

It readily follows that a subset of the eigenvalues of F form the entries of Φ. If F is taken to be rank d (e.g., the LS solution $F = YX^+$), then F has $L - d$ zero eigenvalues and Φ is equal to the d nonzero eigenvalues of F.

If there is noise, X and Y will have full rank L. We will consider two classes of solutions to solve the problem in this case. In Section IV, the algebraic structure present in (25) is exploited; i.e., these methods rely on the fact that X and Y

should ideally have the same (d-dimensional) column space and row space. By SVD analysis on X and Y, rank-d LS or TLS approximations $\hat{X}$ and $\hat{Y}$ are obtained that satisfy this property, without retaining any (Hankel) structure that might be present in X and Y. Setting $F\hat{X} = \hat{Y}$, and solving for F in a Least Squares sense, the entries of Φ are obtained as the d nonzero eigenvalues of F. These methods are also known as Principal Component methods because the column/row spaces of $\hat{X}, \hat{Y}$ contain the d strongest components in the column/row spaces of X, Y, and are obtained by projecting X, Y onto these "principal" subspaces. In many identification contexts (except DOA estimation with sensor doublets), the fact that Y has many entries in common with X is in principle not used in finding or projecting onto these subspaces. However, this fact can be exploited in the derivation of algorithms that are more computationally efficient.

Section VII describes the Orthogonal Vector methods as an intermediary between Single Shift-Invariance and Subspace Fitting techniques. They can be written in the same style as the Single Shift-Invariance methods, operating on X and Y in (25) above, and using the single shift-invariance between them to obtain a different F, now having full rank L and a special structure. Φ is obtained by selecting an appropriate set of d eigenvalues from the L eigenvalues of F. On the other hand, it can be shown that these methods compute a rank-L Vandermonde matrix $\mathcal{A}$ that is precisely orthogonal to one selected vector $\boldsymbol{u}$ in the noise subspace of H. Due to the structure of $\mathcal{A}$, this reduces the problem to finding the roots of the polynomial $u(z)$ associated with this vector. Taking $\mathcal{A}$ of rank d and maximally orthogonal (in some Least Squares sense) to *all* vectors in the noise subspace instead of just one, a connection with the Subspace Fitting class of techniques is obtained.

VI. Least Squares Single Shift-Invariant Methods

In the Single Shift-Invariant (or Principal Component) methods described in this section, the $(L+1)$ rows of the data matrix H are arranged into two matrices X and Y, with $X = H^{(1)}$ consisting of the first L rows of H, and $Y = H^{(2)}$ consisting of its last L rows. As was already stated in the previous section, the first step in this class of solutions is to find rank d approximants $\hat{X}$ and $\hat{Y}$ to X and Y, and then invoke the $(\mathcal{A}\mathcal{S}, \mathcal{A}\Phi\mathcal{S})$ structure of (25) to estimate Φ. Any additional shift-invariance structure that might be present in X and Y is not used, and also not retained by this rank reduction. The approximation is performed by projections onto subspaces spanned by the d most important singular vectors derived from SVD analysis of X and/or Y, and the approximation norm is the Frobenius norm. In LS solutions, the projection operators are constructed from either the X data or the Y data, in a way that closely follows the definitions of Π_c and Π_r in Section IV. In Total Least Squares (TLS) solutions, the subspaces, and hence the projection operators, are obtained from *both* the X and Y data [56], [15]. An outline of the procedure described in the previous section which covers almost all algorithms in this and the next section is given in the following list.

1) Using the LS or TLS approximation algorithms of Section IV, estimate from the row space of X and/or Y a "common" d-dimensional row space, i.e., the row space of $\mathcal{S}$. Let Π_r represent the orthogonal projector onto this space.
2) Estimate from the column (range) space of X and/or Y a d-dimensional "common" column space, i.e., the column space of $\mathcal{A}$. Let Π_c represent the orthogonal projector onto this space.
3) Apply these projectors to X and Y to obtain the rank-d approximants

$$\hat{X} = \Pi_c \, X \, \Pi_r$$
$$\hat{Y} = \Pi_c \, Y \, \Pi_r \, .$$

Next, find any matrix F such that $F\hat{X} = \hat{Y}$, and set the entries of the diagonal matrix Φ equal to the nonzero eigenvalues of F. These eigenvalues are the rank-reducing numbers of the pencil $(\hat{Y} - \lambda\hat{X})$.

The solution is by no means unique. Each of the projections used to obtain $\hat{X}$ and $\hat{Y}$ can be done in either LS or TLS sense, giving rise to at least four different, though closely related solutions. In addition, a matrix F such that $F\hat{X} = \hat{Y}$ cannot only be found in LS sense, in which case it will have rank d, but also in a "predictor" form of full rank L, in which the first $L-1$ rows of $\hat{X}$ are just copied by F to $\hat{Y}$, and the last row of $\hat{Y}$ is constructed by F as a linear combination of the rows of X. Although in the latter case F is of full rank L, only d eigenvalues are relevant to the solution and somehow these d eigenvalues must be separated from the rest. This fact can give rise to problems. Three of the four LS/TLS methods which lead to rank-d estimates of F have appeared in the literature, and are discussed below. Predictor methods are discussed in Section VII.

Principal Component methods were introduced by Moore in 1978 (see [10], [57]), who analyzed such methods on the Grammians of internally balanced systems. This work is related to the Principal Hankel Component analysis discussed here. Related papers are by Zeiger and McEwen [9] and by Pernebo and Silverman [58]. In the past decade, several major contributions in this field have appeared in the publications of Kung *et al.* [11], [59]–[61], in which infinite-data Principal Component algorithms and related covariance methods are discussed, with applications to state-space and harmonic retrieval problems. This research has led to a covariance-based method referred to as TAM, the direct-data variant of which is related to the LS–LS and TLS–LS cases discussed below. In another series of publications, Roy, Paulraj, and Kailath [34]–[36] have devised a comparable harmonic retrieval algorithm called ESPRIT, corresponding to the TLS–TLS case discussed below. Since then, a number of authors [62]–[65] have investigated the relationship between TAM and ESPRIT, and concluded that their statistical performance is asymptotically (i.e., for

$N \to \infty$) equivalent. Other authors have popularized the use of a pencil description for the same type of problem [66]–[68]. The classification below is both a summary and unification of the underlying concepts in the above publications, and does not precisely follow any of them in particular.

A. LS–LS Algorithm

In the LS–LS type algorithms, both Π_r, the projector onto the common row space, and Π_c, the projector onto the common column space, are determined from an SVD of X only (hence Least Squares). Following the outline above, the algorithm is in principle as follows:

1) Determine the SVD of X:

$$X = U\Sigma V^* .$$

The rank-d LS approximant $\hat{X}$ of X is $\hat{X} = \hat{U}\hat{\Sigma}\hat{V}^*$, where $\hat{U}$ and $\hat{V}$ contain the d singular vectors corresponding to the d largest singular values $\hat{\Sigma}$ in Σ. The LS projectors onto these subspaces are

$$\Pi_c = \hat{U}\hat{U}^*$$
$$\Pi_r = \hat{V}\hat{V}^*$$

and the LS–LS approximations of X and Y are

$$\hat{X} = \hat{U}\hat{U}^* X \hat{V}\hat{V}^* = \hat{U}\hat{\Sigma}\hat{V}^*$$
$$\hat{Y} = \hat{U}\hat{U}^* Y \hat{V}\hat{V}^* .$$

2) Put $F\hat{X} = \hat{Y}$, and solve for F in the LS sense:

$$F = \hat{Y}\hat{X}^+ = \hat{U}\hat{U}^* Y \hat{V}\hat{\Sigma}^{-1}\hat{U}^* .$$

3) Compute $\Phi = \text{eig}(F)$, discarding zero eigenvalues. Using the fact that the nonzero eigenvalues of a matrix product (AB) are equal to the nonzero eigenvalues of the product (BA), we can obtain Φ as

$$\Phi = \text{eig}(\hat{U}^* Y \hat{V} \cdot \hat{\Sigma}^{-1}) .$$

It is thus seen that the actual computations needed in the LS–LS case amount to 1) computing the SVD of X, and 2) computing $\Phi = \text{eig}(\hat{U}^* Y \hat{V}\hat{\Sigma}^{-1})$. This shows that the projection of Y onto the column space of $\hat{X}$ is in fact not needed ($\hat{Y}$ need not be computed) because this is a side effect of computing $(Y\hat{X}^+)$. The projection of Y onto the row space of $\hat{X}$ can also be omitted because the computation of $\text{eig}(Y\hat{X}^+)$ will implicitly project Y onto the row space and column space of $\hat{X}$ (see Section IV).

The LS–LS algorithm is akin to the "direct matrix pencil algorithm" described by Hua and Sarkar [67], [68], although in their approach $\hat{Y}$ is constructed from an SVD on Y, rather than by projections based on X. This has the conceptual advantage that X and Y are treated equally. Φ is then determined as $\Phi = \text{eig}(\hat{Y}\hat{X}^+)$ as before.

B. TLS–LS Algorithm

In the TLS–LS algorithm, a d-dimensional common row space for $\hat{X}$ and $\hat{Y}$ is obtained by SVD analysis of the full data matrix H of which $X = H^{(1)}$ and $Y = H^{(2)}$ are submatrices. This determines the projector Π_r onto the row space. The projector Π_c is the projector onto the column space of $X\Pi_r$, but is never explicitly formed because the projection is done implicitly in the computation of $\text{eig}(F)$, as in the LS–LS case. The outline of the algorithm is as follows:

1) With $X = H^{(1)}$ and $Y = H^{(2)}$, compute the SVD of the full data matrix H:

$$H = U\Sigma V^* \quad \to \quad \hat{H} = \hat{U}\hat{\Sigma}\hat{V}^*$$

where $\hat{V}^*$ represents the common d-dimensional row space of X and Y in the TLS sense, i.e., $\Pi_r = \hat{V}\hat{V}^*$. Project X and Y onto this row space (hence TLS):

$$\hat{X} = X\hat{V}\hat{V}^* = \hat{U}_1\hat{\Sigma}\hat{V}^*$$
$$\hat{Y} = Y\hat{V}\hat{V}^* = \hat{U}_2\hat{\Sigma}\hat{V}^* ,$$

where $\hat{U}_1 = \hat{U}^{(1)}$ consists of the first L rows of $\hat{U}$, and $\hat{U}_2 = \hat{U}^{(2)}$ consists of the last L rows of $\hat{U}$. Hence $\hat{U}_1$ and $\hat{U}_2$ are matrices of size $L \times d$, and in fact $\hat{X}$ and $\hat{Y}$ are just submatrices of $\hat{H}$.

2) Set $F = \hat{Y}\hat{X}^+$. Then $F = \hat{U}_2\hat{U}_1^+$, and

$$\Phi = \text{eig}(F) = \text{eig}(\hat{U}_1^+\hat{U}_2)$$

(discarding zero eigenvalues).

By construction, $\hat{X}$ and $\hat{Y}$ share the same row space. Again, the computation of $\text{eig}(F)$ implicitly projects the columns of $\hat{Y}$ onto the column space of $\hat{X}$ in the LS sense.

The above method is known in the DOA context as the LS–ESPRIT algorithm. As before, X and Y typically contain data from two identical sensor subarrays, H is obtained by stacking X and Y, thus having size $2L \times N$. The method also encompasses the "diret-data" TAM method for harmonic retrieval in [59], although the description of the computation is slightly different here. It is observed in [59] that $\hat{U}_1^+$ can be computed without inverting matrices because $\hat{U}_1$ is almost an isometry. To see this, denote by $\boldsymbol{u}_L$ the last row of $\hat{U}$, and note that $\hat{U}_1^*\hat{U}_1 + \boldsymbol{u}_L^*\boldsymbol{u}_L = I_d$. Elaborating on this formula, it follows from $\hat{U}_1^+\hat{U}_1 = I_d$ that

$$\hat{U}_1^+ = (I_d - \boldsymbol{u}_L^*\boldsymbol{u}_L)^{-1}\hat{U}_1^*$$
$$= (I_d + \frac{\boldsymbol{u}_L^*\boldsymbol{u}_L}{1 - \boldsymbol{u}_L\boldsymbol{u}_L^*})\hat{U}_1^* .$$

The TLS–LS algorithm (as well as the LS–LS algorithm) is suitable for efficient SVD updating techniques [69], [70], which can be adapted to yield on-line estimates of Φ for an increasing number of samples N.

C. TLS–TLS Algorithm

In the above two algorithms, the actual choice of F results in an implicit LS projection of the columns of $\hat{Y}$ onto the column space of $\hat{X}$ when the eigenvalues are computed.

In the TLS–TLS method, an explicit projection is done in the TLS sense by projecting the columns of $\hat{X}$ and $\hat{Y}$ onto a subspace that lies "between" the column space of X and the column space of Y. This subspace is obtained by computing the SVD of a matrix $H_1 = [X \quad Y]$ and retaining the d left singular vectors that correspond to the d largest singular values. Although this extra projection gives rise to results which are slightly different from TLS–LS, and presumably more accurate, the difference with the TLS–LS case for system identification is only marginal if N is large. The algorithm is given below.

1) With $X = H^{(1)}$ and $Y = H^{(2)}$, denote $H_1 = [X \quad Y]$, $H_2 = H$. Compute the SVD's of H_1 and H_2, and denote their rank-d approximants by $\hat{H}_1$ and $\hat{H}_2$:

$$H_1 = U_1\Sigma_1 V_1^* \rightarrow \hat{H}_1 = \hat{U}_1\hat{\Sigma}_1\hat{V}_1^*$$
$$H_2 = U_2\Sigma_2 V_2^* \rightarrow \hat{H}_2 = \hat{U}_2\hat{\Sigma}_2\hat{V}_2^* .$$

In this step, the common column and row spaces of X and Y are determined explicitly to be $\hat{U}_1$ and $\hat{V}_2^*$, and the projectors onto these subspaces are $\Pi_c = \hat{U}_1\hat{U}_1^*$ and $\Pi_r = \hat{V}_2\hat{V}_2^*$.

2) Define

$$\hat{X} = \hat{U}_1\hat{U}_1^* \cdot X \cdot \hat{V}_2\hat{V}_2^* =: \hat{U}_1 E_x \hat{V}_2^*$$
$$\hat{Y} = \hat{U}_1\hat{U}_1^* \cdot Y \cdot \hat{V}_2\hat{V}_2^* =: \hat{U}_1 E_y \hat{V}_2^*$$

where E_x and E_y are $d \times d$ the following full rank matrices:

$$E_x = \hat{U}_1^* \; X \; \hat{V}_2$$
$$E_y = \hat{U}_1^* \; Y \; \hat{V}_2 . \qquad (26)$$

With these definitions, $\hat{X}$ and $\hat{Y}$ are rank d and share common column and row spaces obtained by (TLS) projections onto both the column space spanned by $\hat{U}_1$ and the row space spanned by $\hat{V}_2^*$. They reflect the structure of the assumed model (25) in the sense that they are weighted "outer products" of rank-d rectangular matrices, the weights being the $d \times d$ matrices E_x and E_y.

3) Set $F = \hat{Y}\hat{X}^+$, then

$$\text{eig}\,(F) = \text{eig}\begin{bmatrix} E_y E_x^{-1} & \\ & 0 \end{bmatrix}$$

and

$$\Phi = \text{eig}\,(E_y E_x^{-1}) .$$

The computation of E_x and E_y in (26) can be done efficiently by defining the matrices U_{11} and U_{21} to be the first and last L rows of U_2, respectively, so that

$$X = U_{11}\Sigma_2 V_2^*$$
$$Y = U_{21}\Sigma_2 V_2^* .$$

Substituting this in the definitions of E_x and E_y in (26) and using the fact that $\Sigma_2 \; V_2^*\hat{V}_2 = [\hat{\Sigma}_2 \;\; 0]^{\mathrm{T}}$, we obtain

$$E_x = \hat{U}_1^*\hat{U}_{11}\hat{\Sigma}_2$$
$$E_y = \hat{U}_1^*\hat{U}_{21}\hat{\Sigma}_2 .$$

Multiplication by $\hat{\Sigma}_2$ can even be omitted, since this will not affect Φ.

The above algorithm only requires computation of the SVD's of $H_1$and H_2, followed by the computation of the eigenvalues of the pair $(\hat{U}_1^*\hat{U}_{11}, \hat{U}_1^*\hat{U}_{21})$. If X and Y are *Hankel* matrices (as in system identification), then X and Y have all but two columns in common. It seems in this case better to omit the duplicate columns in H_1, but in doing so H_1 and H_2 differ in only one row and one column, and for large N the difference between the TLS–TLS and TLS–LS algorithms is negligible. If the Hankel assumption is not used, then the above algorithm is a modification of the more sequential TLS–ESPRIT algorithm of [35], [36] in which the projection onto the common column space is done first, and the common row space is then determined from the *resulting* smaller matrices. In the ESPRIT context, H is obtained by stacking X and Y. Because the noise on X is now unrelated to the noise on Y, the difference between the LS and the TLS variant can be significant.

D. Pro-ESPRIT

For completeness, and to indicate that there exists a litany of algorithms that are all based on (repeated) rank-d truncations of matrices constructed from X and Y, we mention an algorithm based on Procrustes rotations [71], called Pro-ESPRIT. The algorithm can basically be formulated as follows [72]. Starting from data matrices X and Y as before, compute (independent) rank-d approximations $\hat{X}$, $\hat{Y}$ using SVD's:

$$\begin{array}{rcl} X &=& U_1\Sigma_1 V_1^* \\ Y &=& U_2\Sigma_2 V_2^* \end{array} \quad \rightarrow \quad \begin{array}{rcl} \hat{X} &=& \hat{U}_1\hat{\Sigma}_1\hat{V}_1^* \\ \hat{Y} &=& \hat{U}_2\hat{\Sigma}_2\hat{V}_2^* . \end{array} \qquad (27)$$

Then the rank reducing numbers of $\hat{Y} - \lambda\hat{X}$ are equal to those of the rank-d pencil

$$Q_u\hat{\Sigma}_2 Q_v^* - \lambda\hat{\Sigma}_1 ,$$

with $Q_u = \hat{U}_1^*\hat{U}_2$ and $Q_v = \hat{V}_1^*\hat{V}_2$. Under noise-free conditions, Q_u and Q_v are unitary matrices. With noise they are not, but can be replaced (approximated) by their closest unitary matrices $\hat{Q}_u$ and $\hat{Q}_v$. This is called a Procrustes approximation, and $\hat{Q}_u$ and $\hat{Q}_v$ can be computed via SVD's of Q_u and Q_v by setting all singular values equal to one. Hence Φ is determined from the d rank reducing numbers of the approximated pencil $\hat{Q}_u\hat{\Sigma}_2\hat{Q}_v^* - \lambda\hat{\Sigma}_1$. In [72] it is shown that, under certain conditions, this algorithm yields results identical to those that would be obtained by replacing $\hat{U}_1$ and $\hat{U}_2$ in (27) with approximating unitary matrices sharing a common d-dimensional space. This approximation is obtained via an SVD of $[\hat{U}_1 \;\; \hat{U}_2]$, and $\hat{V}_1$ and $\hat{V}_2$ are approximated in the same fashion. The resulting algorithm can be viewed as yet another (two-step) variant of the algorithms mentioned above, where a common d-dimensional subspace of the column spaces of X and Y is determined in two successive steps.

E. Discussion

It is a difficult matter to decide which of the above algorithms is to be preferred. They are all closely related, and their differences tend to disappear when N is large since they are all asymptotically (for large N) equivalent to a first-order approximation [72]. The variance of the estimated parameters is, however, smaller for the various TLS implementations. If a parallel array of processors is used, then there is not a dramatic difference in the number of operations between the LS–LS and TLS–TLS algorithm (less than a factor 2), because on a parallel array it takes about the same number of operations to compute the SVD of a matrix as it takes to apply the resulting U and V matrices to a second matrix. Pro-ESPRIT requires roughly twice the amount of computation, and is not necessarily more accurate. SVD updating techniques are very promising for an on-line (or real-time) implementation of the TLS–LS algorithm. In these techniques, estimates of Φ are calculated for increasing values of N by updating the SVD's of X and Y obtained from some previous value of N [70].

VII. Orthogonal Vector Methods

A. Introduction

As before, we assume that an $(L+1) \times N$ data matrix H is given, and let $X = H^{(1)}$ represent the matrix constructed from the first L rows of H, and $Y = H^{(2)}$ represent the matrix containing the last L rows. The last (unique) row of Y is denoted $\boldsymbol{y}_L$. In contrast to the Least Squares Single Shift-Invariant methods of the previous section, Orthogonal Vector methods exploit the fact that Y is a shifted version of X, so that, in the relation $FX = Y$, F can be chosen to be an $L \times L$ matrix with the following structure:

$$F = \begin{bmatrix} 0 & 1 & & & \\ & 0 & 1 & & \\ & & \ddots & \ddots & \\ & & & 0 & 1 \\ \hline & & \boldsymbol{g} & & \end{bmatrix} \tag{28}$$

This reflects the fact that all but the last rows of Y are just copies of rows of X. The last row $\boldsymbol{y}_L$ of Y is obtained as a linear combination of rows of X, $\boldsymbol{g}X = \boldsymbol{y}_L$, and hence $[\boldsymbol{g} \ -1]H = 0$. Consequently, $[\boldsymbol{g} \ -1]^*$ could be any vector in the left null space of H. As mentioned in the problem outline in Section V, $\Phi = \text{eig}(F)$ has L eigenvalues, only d of which are relevant. In the noise-free case, the valid eigenvalues are independent of L. The remaining $(L-d)$ eigenvalues depend on the particular choice of $\boldsymbol{g}$.

An alternative approach leading to the same result takes the analytic structure of H into account. With the definition $\boldsymbol{a}(z) = [1 \ \ z \ \ z^2 \ \ \cdots]^{\mathrm{T}}$, we can associate with the vector $\boldsymbol{u} = [u_1 \ \ u_2 \ \ \cdots]^{\mathrm{T}}$ a polynomial $u(z) = \boldsymbol{u}^*\boldsymbol{a}(z) = \bar{u}_1 + \bar{u}_2 z + \cdots$. The basic property used by all Orthogonal Vector Methods is the trivial (noise-free) relationship that $H = \mathcal{A}(\Phi)\mathcal{S}$ satisfies when $\mathcal{A}(\Phi)$ is a Vandermonde-type matrix (see (16))

$$\begin{aligned} \boldsymbol{u}^*H = 0 \quad &\Leftrightarrow \quad \boldsymbol{u}^*\mathcal{A}(\Phi) = 0 \\ &\Leftrightarrow \quad u(\phi_i) = 0 \qquad (i = 1, \cdots, d) \end{aligned} \tag{29}$$

which states that if $\boldsymbol{u}$ is in the left null space of H, then the d elements ϕ_i on the diagonal of Φ must be solutions of the equation $u(z) = 0$. Hence, the polynomials $u(z)$ derived from all possible vectors $\boldsymbol{u}$ in the left null space of H have d roots in common, and in the noise-free case the choice of $\boldsymbol{u}$ in this null space is of no particular importance. In comparison with the previous paragraph, we see that $\boldsymbol{u}$ must be proportional to $[\boldsymbol{g} \ -1]^*$. The equality between the eigenvalues of a matrix in bottom companion form (F in (28)) and the roots of the polynomial constructed from the last row of this matrix is a well-known result in linear systems theory [12].

Orthogonal Vector methods are sometimes called prediction methods. This is because when H is a Hankel matrix built from a time series h_k, the entries g_i of $\boldsymbol{g}$ can be thought of as the coefficients of a linear prediction filter (moving average filter)

$$G(z) = \sum_{1}^{L} g_{L-i+1} z^i$$

that predicts a new data sample h_{k+1} from knowledge of the preceding L samples $\{h_k, \cdots, h_{k-L+1}\}$, for $k = L$ to $k = N + L - 1$. In writing out the equations, it is seen that $\boldsymbol{g}$ predicts $\boldsymbol{y}_L$ from a linear combination of the rows of X by minimizing the error $(\boldsymbol{y}_L - \boldsymbol{g}X)$. For such a matched filter, the zeros ϕ_i of the prediction-error filter $-1 + G(z)$ are the zeros of $u(z)$ in (29), and hence equal to the poles of the system that generated the data because the inverse prediction-error filter will have the original data sequence as its impulse response. There are many variants of such linear prediction methods. For example, when the data are known to be sinusoidal, as in the harmonic retrieval problem, then doing both a forward (as above) and a backward prediction (predicting h_k from $\{h_{k+1}, \cdots, h_{k+L}\}$) yields improved accuracy. When covariance data are used instead of direct data, then the resulting relationships are known as the Yule–Walker equations; they are solved in precisely the same way as in the sequel to this section [73], [74].

If H is a noisy data matrix, then an approximation $\hat{H} = \hat{U}\hat{\Sigma}\hat{V}^*$ may be formed from the SVD representation $H = U\Sigma V^*$. In this way, the column space of H is split into a signal subspace $\hat{U}$ and a noise subspace $\hat{U}^{\perp}$ which is the left null space of $\hat{H}$. There exist a number of Orthogonal Vector methods, each of which differs from the others in the actual choice of the vector $\boldsymbol{u}$ in the noise subspace. Because, with noise, $\hat{H}$ no longer has a left factor $\mathcal{A}$ with Vandermonde structure, property (29) above is no longer valid and different selections of vectors $\boldsymbol{u}$ in the noise subspace lead to different solutions. A few of the possibilities are discussed in the subsections which follow.

One of the problems associated with these methods is that only d of the L roots of the polynomial $u(z)$ are of interest; namely, those that correspond to the system

poles. Apart from the computational overhead incurred in obtaining L roots (in comparison with the order-d eigenvalue calculations of the previous section), one is also faced with the problem of how to select these d roots. Each of the methods below has its own rationale behind its selection criteria. A few observations are indicative in this respect. One is that if H were noise-free, then the residues r_i of the underlying model $h(z)$ (in (3)) that correspond to the $L-d$ extra roots would be zero, and hence these extra poles would be unobserved in the model. If H does contain noise, one might assume that these spurious residues are still small. Another observation is that for rank-d Hankel matrices H that are corrupted by additive *white noise*, the $L-d+1$ smallest singular values of H would be equal. The theory of Adamjan, Arov, and Krein (AAK) in [75] states that if the $(d+1)$st through $(L+1)$st singular values of H are equal, then the polynomials constructed from any of the corresponding singular vectors have d roots in common. Hence, for both the white noise and noise-free cases, all polynomials constructed from vectors that are in the noise subspace of H have d roots in common, and the results of each of the methods discussed below should "asymptotically" be the same.

Below, a brief overview is given of four Orthogonal Vector methods: Padé approximation (as in [59]), Kumaresan–Tufts (KT) Min-norm method with and without rank reduction, and AAK Hankel-norm model reduction. The first two methods are included for historical reasons. The different methods are characterized by the choice of the representative vector from the noise subspace (as in [76]) since the roots of the polynomial constructed from this vector are directly related to the poles of the approximating system.

B. Padé Approximation

In this class of methods, the data matrix X is square and of full rank, so $N = L$ and $d = L$. Hence the order of the system determines the size of the data matrices to be used, and *vice versa*. The vector $\boldsymbol{g}$ is defined by

$$\boldsymbol{g}X = \boldsymbol{y}_L \qquad \Rightarrow \qquad \boldsymbol{g} = \boldsymbol{y}_L X^{-1}.$$

With F constructed from $\boldsymbol{g}$ as before (see (28)), we have $FX = Y$ and $\Phi = \mathrm{eig}(F)$. The "approximating" system which results is of degree $d = L$. Since this method uses all data without rank reduction, it is very sensitive to perturbations in X and $\boldsymbol{y}_L$ [59], and the number of measurements directly determines the degree of the approximating system. The noise subspace is defined in this case by the null space of $H^* = [X^* \quad \boldsymbol{y}_L^*]$, which has dimension one, and is spanned by the vector $[\boldsymbol{g} \quad -1]^*$. A related method is the classical method of Prony [39] for sinusoidal data.

C. Kumaresan–Tufts Method Without Rank Reduction

In the Kumaresan–Tufts method without rank reduction, it is assumed that the $L \times N$ matrices X and Y satisfy $N > L$. Since no rank reduction is done, we still have $d = L$. In comparison with the Prony method, it is seen that the restriction $N = L$ is removed. The vector $\boldsymbol{g}$ is computed by trying to solve the *overdetermined* system of equations $\boldsymbol{g}X = \boldsymbol{y}_L$ for $\boldsymbol{g}$. With noise present, the null space of $H^* = [X^* \quad \boldsymbol{y}_L^*]$ will contain no vectors at all; the row $\boldsymbol{y}_L$ is not contained in the row space of X. However, after projecting $\boldsymbol{y}_L$ onto the row space of X, resulting in $\hat{\boldsymbol{y}}_L = \boldsymbol{y}_L X^+ X$, the null space of $\hat{H}^* = [X^* \quad \hat{\boldsymbol{y}}_L^*]$ spans precisely one vector: $[\boldsymbol{g} \quad -1]^*$. This $\boldsymbol{g}$ is also the solution to the minimization problem

$$\min_{\boldsymbol{g}} \| \boldsymbol{g}X - \boldsymbol{y}_L \|_2$$

and is determined explicitly as $\boldsymbol{g} = \hat{\boldsymbol{y}}_L X^+$. Note that the LS methods of Section VI with $d = L$ yield precisely the same result since no actual rank reduction is done; i.e., $F = YX^+$ is the same as that obtained here. Pisarenko's method [77] for harmonic retrieval operates on a covariance matrix constructed on the given data but is essentially the same method (see [60]). These methods are still very sensitive to perturbations in X due to noise.

D. Kumaresan–Tufts Minimum-Norm Method

The Min-norm method proposed by Kumaresan–Tufts [78]–[83] is a modification of the above method to make it more robust for the separation of closely spaced sinusoids in the presence of noise. It amounts to the following three steps:

1) A solution to $\boldsymbol{g}X = \boldsymbol{y}_L$ is forced by reducing $H = [X^* \quad \boldsymbol{y}_L^*]^*$ to rank d. This can be done in two ways. The classical LS way would compute a rank-d approximation $\hat{X}$ from an SVD of X, and project $\boldsymbol{y}_L$ onto the row space of $\hat{X}$ to obtain an $\hat{\boldsymbol{y}}_L$ such that $\boldsymbol{g}\hat{X} = \hat{\boldsymbol{y}}_L$ has solutions $\boldsymbol{g}$:

$$X \approx U\Sigma V^* \rightarrow \hat{X} = \hat{U}\hat{\Sigma}\hat{V}^*$$
$$\hat{\boldsymbol{y}}_L = \boldsymbol{y}_L \hat{V}^* \hat{V}.$$

This is the counterpart of the LS–LS method of the previous section. A TLS method (cf. the TLS–LS method of Section VI) would compute the SVD of H and derive $\hat{X}$, $\hat{\boldsymbol{y}}_L$ from the rank-d approximation $\hat{H}$ as follows:

$$H = U\Sigma V^* \quad \rightarrow \quad \hat{H} = \hat{U}\hat{\Sigma}\hat{V}^* = \begin{bmatrix} \hat{X} \\ \hat{\boldsymbol{y}}_L \end{bmatrix}$$

which yields

$$\hat{X} = X\ \hat{V}^*\hat{V} = \hat{U}^{(1)}\hat{\Sigma}\hat{V}^*$$
$$\hat{\boldsymbol{y}}_L = \hat{\boldsymbol{y}}\ \hat{V}^*\hat{V} = (\hat{U})_L\hat{\Sigma}\hat{V}^*$$

where $(\hat{U})_L$ is the last row of $\hat{U}$.

2) The system $\boldsymbol{g}\hat{X} = \hat{\boldsymbol{y}}_L$ is now *underdetermined*, and the noise subspace of $\hat{H}$ has dimension $L-d+1$. Of the many possible vectors $[\boldsymbol{g} \quad -1]^*$ in this subspace, choose the one with minimum norm $\| \boldsymbol{g} \|_2$, i.e., choose $\boldsymbol{g} = \hat{\boldsymbol{y}}_L\hat{X}^+ = \boldsymbol{y}_L\hat{X}^+$ as in the previous case. If the TLS approach is used in the above step, then we can show that in fact

$$[\boldsymbol{g} \quad -1] \sim (\hat{U}^{\perp})_L\hat{U}^{\perp *}$$

in which $\hat{U}^{\perp}$ spans the noise subspace defined via $U = [\hat{U}\ \ \hat{U}^{\perp}]$, and $(\hat{U}^{\perp})_L$ is the last row of $\hat{U}^{\perp}$ (see also [84]). This determines precisely which vector of the noise subspace is chosen.

3) The estimated d poles are a subset of the eigenvalues of F, with F as in (28). We could also compute the roots of the polynomial associated with $[\boldsymbol{g}\ \ -1]^*$, leading to the same result.

In comparison with the previous method, the rank reduction to order d in combination with a null space vector of dimension larger than d greatly improves the previous two methods [82]. The choice of $\boldsymbol{g}$ to have minimal norm among all vectors $\boldsymbol{g}$ that satisfy $\boldsymbol{g}\hat{X} = \hat{\boldsymbol{y}}_L$ forces the extra $L-d+1$ eigenvalues of F to lie regularly spaced on a circle of minimal radius within the unit disc [78]–[80], [85]. This property can be used to select the d desired eigenvalues.

E. AAK Hankel-Norm Approximations

The ultimate goal of the methods considered in this paper is, given a (full-rank) matrix H, to find a rank d approximating structured matrix $\hat{H} = \mathcal{A}(\hat{\Phi})\mathcal{S}$ that minimizes in an appropriate norm the difference $H - \hat{H}$. In Section VI, the minimizing norm was taken at first to be the Frobenius norm:

$$\min_{\hat{H}\ \text{rank}\ d} \|H - \hat{H}\|_{\text{F}}^2. \tag{30}$$

However, the minimizing $\hat{H}$ does not have the required shift-invariance structure. By ignoring this fact, and using properties that $\hat{H}$ would have in the noise-free case, we were able to derive a reduced-order model that does possess shift-invariance structure and is presumably not too far away from $\hat{H}$. Unfortunately, to date no bound has been found to quantify this error. In Section VIII, techniques will be discussed that do solve the above minimization problem in the Frobenius norm, taking the structure of the approximant into account. This is a highly nonlinear operation, leading to complicated search techniques. Under certain conditions, however, it can be shown that a structured solution can be found when a different norm is applied. Such a norm is the Hankel norm.

In a celebrated paper, Adamjan, Arov, and Krein [75] have demonstrated that, when H is a Hankel matrix of infinite dimensions, but of finite rank and bounded L_2 norm, there exists a unique Hankel matrix $\hat{H}$ that is the solution to a related minimization problem:

$$\min_{\hat{H}\ \text{rank}\ d} \|H - \hat{H}\| \tag{31}$$

in which the matrix L_2 (operator) norm is minimized

$$\|H - \hat{H}\| = \sup_{\|\boldsymbol{x}\|_2 = 1} \|H\boldsymbol{x} - \hat{H}\boldsymbol{x}\|_2 .$$

Recall that the L_2 norm of a matrix is in fact equal to its largest singular value. The use of this norm leads to a so-called Hankel-norm approximation of the impulse response vector $\boldsymbol{h}$ on which H was built, or its polynomial $h(z)$; i.e., it is the approximation in L_2 norm of the Hankel matrix associated with $h(z)$. Unlike the Frobenius norm, the Hankel-norm approximation allows the d vectors spanning the range of $\hat{H}$ to have components outside the range spanned by the first d singular vectors of H without penalty on the norm of the error, because the norm only measures the largest singular value. This enables $\hat{H}$ to take on a Hankel structure, something that the SVD methods of Sections VI and VII were not able to achieve. We can summarize the main results [59], [75], [86]–[91] as follows, favoring vector notations over polynomial descriptions, when possible, for better comparison with the previous methods.

Given a matrix H of infinite size, representing a stable high-order system, let $H = U\Sigma V^*$ and denote the $(d+1)$st column of U by $\boldsymbol{u}_{d+1}$. With $U = [\hat{U}\ \ \hat{U}^{\perp}]$ as before, $\boldsymbol{u}_{d+1}$ is the first column of $\hat{U}^{\perp}$, the noise subspace. The corresponding singular value and right singular vector of $\boldsymbol{u}_{d+1}$ are denoted σ_{d+1} and $\boldsymbol{v}_{d+1}$.

1) The polynomial $u_{d+1}(z)$ constructed from $\boldsymbol{u}_{d+1}$ has precisely d "stable" roots ϕ_i inside the unit circle.
2) If $\hat{H}$ is a rank-d Hankel matrix approximating H according to (31), then the minimum error $\|H - \hat{H}\| = \sup_{\boldsymbol{u}} \|\boldsymbol{u}^*H - \boldsymbol{u}^*\hat{H}\|$ equals σ_{d+1} and is attained by the corresponding left singular vector $\boldsymbol{u}_{d+1}$ of H:

$$\boldsymbol{u}_{d+1}^*(H - \hat{H}) = \sigma_{d+1}\boldsymbol{v}_{d+1}^*$$

where $\boldsymbol{v}_{d+1}$ is the $(d+1)$st right singular vector. Since $\boldsymbol{u}_{d+1}^*H = \sigma_{d+1}\boldsymbol{v}_{d+1}^*$, we must have $\boldsymbol{u}_{d+1}^*\hat{H} = 0$. Hence the columns of $\hat{H}$ are all orthogonal to $\boldsymbol{u}_{d+1}$, or, in the context of the previous section, $\boldsymbol{u}_{d+1}$ is in the noise subspace associated with H.

3) Combining the above two properties, it is concluded that the d stable roots of $u_{d+1}(z)$ define the best rank-d Hankel approximant in the L_2 norm.

The above properties are derived only for infinite-dimensional Hankel matrices. If a high-order (stable) model of $h(z) = b(z)/a(z)$ is known, for example in the form of a high-order state-space model, then the theory can be extended to operate on Hankel matrices of finite size that are larger than or equal to the model order, thereby obtaining the same results [59], [86], [88], [89], [92]. The singular values and vectors of the infinite-dimensional Hankel matrix can then also be easily computed [93]. If operating on Hankel matrices that are windowed (finite) versions of infinite Hankel matrices (as is the case throughout this paper), then the above theory is no longer applicable, although the solution is continuous when the rank of the matrix is finite and the dimension is larger than the rank. However, in general it can easily happen that $u_{d+1}(z)$ has more or fewer roots than d in the open unit disc, especially if the data are corrupted by noise, and hence the rank of the underlying infinite matrix is not finite. One way to avoid these problems is first to derive a high-order stable model using other techniques, and then use an extension of the AAK theory that works on finite-size state-space models to

obtain a rank-d reduced-order model $\hat{H}$. Precise formulas appear in [94], [95, p. 452]. Since this $\hat{H}$ is obtained by a two-step process, it will be a suboptimal solution to (31). However, it will approach the optimal solution as $N, L \to \infty$. AAK Hankel-norm model reduction methods can also be extended to the time-varying context [96].

F. Root MUSIC: A Link to Subspace Fitting Techniques

To link the methods of this section with the Subspace Fitting techniques of the next section, we briefly discuss here a derivation of (root)-MUSIC. In the introduction to this section, we noticed that the basic form of the Orthogonal Vector methods is simply

$$\boldsymbol{u}^* \hat{H} = 0 \quad \Leftrightarrow \quad u(\phi_i) = 0 \tag{32}$$

which means that for a selected $\boldsymbol{u}$ in the left null space of $\hat{H}$, the roots of $u(z)$ are viable estimates of ϕ_i. However, as $\hat{H}$ does not have the Vandermonde structure, different choices of $\boldsymbol{u}$ in this null space will lead to different estimates $\{\phi_i\}$. Because the left null space of $\hat{H}$ is, by definition, spanned by $\hat{U}^\perp$, we can write $\boldsymbol{u}^* = \boldsymbol{w}\hat{U}^{\perp *}$ for some row vector $\boldsymbol{w}$ of dimension $L-d+1$. For example, in the AAK approach $\boldsymbol{w} = [1\ 0 \cdots 0]$ selects the first vector in the noise subspace, while for for the Kumaresan–Tufts TLS method, $\boldsymbol{w} = (\hat{U}^\perp)_L$ is the last row in $\hat{U}^\perp$. Now, using the notation $\boldsymbol{a}(\phi) := [1 \ \ \phi \ \ \phi^2 \ \ \cdots \ \ \phi^L]^{\mathrm{T}}$, (32) is equivalent to the polynomial equation in ϕ

$$\boldsymbol{w}\hat{U}^{\perp *}\boldsymbol{a}(\phi) = 0\,.$$

Orthogonal Vector methods select one specific vector $\boldsymbol{w}$, and search for the roots of the above expression. In this context, the idea behind the well-known DOA estimation algorithm MUSIC is not to select a single $\boldsymbol{w}$, but instead to work with the full polynomial null space $\hat{U}^{\perp *}\boldsymbol{a}$. In particular, root-MUSIC exploits the fact that in the noise-free case, as well as in the infinite-data white-noise case, all entries of the column *vector* of polynomials $\hat{U}^{\perp *}\boldsymbol{a}(z) \equiv \hat{U}^\perp(z)$ have d roots in common. The root-MUSIC algorithm, as a spectral estimation method, then makes the assumption that these roots lie on the unit circle, and estimates them by rooting the sum of squared polynomials $\hat{U}^{\perp *}(z^{-1})\hat{U}^\perp(z)$, retaining only the d roots in the unit disc with modulus nearest unity (only roots inside the unit circle need be considered since the squaring operation forces conjugate reciprocal roots).

To connect this Orthogonal Vector method with the Subspace Fitting methods of the next section, note that the root-MUSIC technique was derived from the MUSIC algorithm, which obtains parameter estimates by minimizing the so-called MUSIC null-spectrum:

$$\min_{\phi_i} \| \hat{U}^{\perp *}\, \boldsymbol{a}(\phi_i) \|_{\mathrm{F}}^2 = \min_{\phi_i} \boldsymbol{a}^*(\phi_i)\, \hat{U}^\perp \hat{U}^{\perp *}\, \boldsymbol{a}(\phi_i)\,, \quad i = 1, \ldots, d \tag{33}$$

for ϕ_i on the unit disc. It can be seen that MUSIC attempts to find, one at a time, d vectors $\boldsymbol{a}(\phi_i)$ from the array manifold which most closely fit the signal subspace, or which are most orthogonal to the noise subspace. Note that MUSIC cannot force the null spectrum to be zero since it only uses vectors $\boldsymbol{a}(\phi)$ from the array manifold in its search; i.e., instead of rooting a polynomial as above, it finds points on the unit circle where the sum of squared polynomials is minimized. On the other hand, root-MUSIC finds the exact roots of this polynomial, and then estimates ϕ_i, $i = 1, \ldots, d$ by projecting these roots onto the unit circle.

VIII. Subspace Fitting Techniques

In this section, the class of Subspace Fitting techniques for solving the direction-of-arrival estimation problem is considered. The discussion follows the framework of Viberg and Ottersten [97] and Stoica *et al.* [98], [99], whose recent work provides an enlightening overview of the DOA estimation problem and new results on the asymptotic behavior of the estimate errors. The generic subspace fitting problem considered in [97] is the following: given some representation of the data M, find $\hat{\Phi}$ and $\hat{T}$ such that

$$\hat{\Phi}, \hat{T} = \arg\min_{\Phi, T} \| M - \mathcal{A}(\Phi)T \|_F^2 \tag{34}$$

for T of suitable size, and with $\mathcal{A}(\Phi)$ and T of rank d. In the sequel, we will often write just $\mathcal{A}$ instead of $\mathcal{A}(\Phi)$. Due to the special structure of $\mathcal{A}$, this is a nonlinear optimization problem, separable however into a linear part in T and a nonlinear part in $\mathcal{A}$. Substituting the solution of the linear part, $\hat{T} = \mathcal{A}^+ M$, back into (34) gives

$$\begin{aligned} \hat{\Phi} &= \arg\min_{\Phi} \| (I - \Pi_{\mathcal{A}})M \|_F^2 \\ &= \arg\max_{\Phi} \mathrm{Tr}(\, \Pi_{\mathcal{A}}(\Phi)\, MM^*\,) \end{aligned} \tag{35}$$

in which $\Pi_{\mathcal{A}}(\Phi) = \mathcal{A}\mathcal{A}^+$ is the LS projector onto the column space of $\mathcal{A}(\Phi)$, and Tr denotes the trace operator.[2]

Several popular DOA estimation algorithms may be cast in the form of (35). These include the deterministic maximum-likelihood method [79], [98], [100]–[104], multidimensional MUSIC [35], [105], as well as Weighted Subspace Fitting (WSF) [97]. The MODE algorithm of Stoica *et al.* also has a related interpretation [98], [99].

In our discussion of identification methods so far, we have been able to avoid the notion of covariance matrices. However, the Subspace Fitting techniques have been introduced in the literature in a statistical framework, and hence the analysis is traditionally not done directly on the data, but rather on the covariance matrix of the data. There are strong links between the two, and it is possible to avoid the notion of covariance altogether (as we have done in the preceding sections), but in the discussion of the present section the use of covariance matrices avoids certain complications. Denoting the $((L+1)\times N)$-dimensional output data matrix H of (24) by $H = H_N = \mathcal{A}(\Phi)\mathcal{S}_N + \mathcal{V}_N$, the relevant

[2] Recall that the trace of a matrix is defined as the sum of its diagonal entries. We will use some of its properties: 1) the trace of a projection operator is equal to the dimension of the subspace on which it projects, 2) $\mathrm{Tr}(AB) = \mathrm{Tr}(BA)$, for matrices A and B of compatible size, 3) $\| A \|_{\mathrm{F}}^2 = \mathrm{Tr}(A^*A)$.

covariance matrices are defined as

$$\text{Signal covariance: } P = \lim_{N\to\infty} \frac{1}{N} \mathcal{S}_N \mathcal{S}_N^*$$

$$\text{Output covariance: } R = \lim_{N\to\infty} \frac{1}{N} H_N H_N^* . \quad (36)$$

With the assumption that the additive noise matrix is a realization of a stationary, zero-mean white Gaussian process with spatial covariance $\sigma^2 I$, we have

$$R = \mathcal{A}(\Phi)\ P\ \mathcal{A}^*(\Phi) + \sigma^2 I$$

if the noise and signals are uncorrelated. If $\mathcal{A}(\Phi)P$ is full rank d, it is easily seen that the $L + 1 - d$ smallest eigenvalues of R are all equal to σ^2. This fact is reflected in our notation for the eigenvalue decomposition of R

$$R = E_s \Lambda_s E_s^* + E_n \Lambda_n E_n^* = E_s \tilde{\Lambda} E_s^* + \sigma^2 I$$

where $E = [E_s \quad E_n]$ is unitary

$$\Lambda_n = \sigma^2 I$$
$$\tilde{\Lambda} = \Lambda_s - \sigma^2 I$$

and E_s and E_n are isometries of rank d and $(L + 1 - d)$, respectively. Since the column space of E_s is equal to that of $\mathcal{A}(\Phi)P$, it is referred to (as above) as the signal subspace. The column space of E_n is correspondingly referred to as the noise subspace.

Since in practical applications we cannot allow $N \to \infty$, the above quantities must be estimated using finite sample averages. Thus the sample covariance R_N of the data is computed as in (36) by removing the limit statement. Estimates of the signal and noise subspaces are then simply obtained by performing an eigenvalue decomposition on R_N, and these estimates will be denoted as $\hat{E}_s$ and $\hat{E}_n$. Comparing R_N with $H_N = U\Sigma V^*$ and its rank-d approximation $\hat{H}_N = \hat{U}\hat{\Sigma}\hat{V}^*$, where $U = [\hat{U} \quad \hat{U}^\perp]$ as usual, we can identify $\hat{U} = \hat{E}_s$, $\hat{U}^\perp = \hat{E}_n$, and $\hat{\Sigma}^2/N = \hat{\Lambda}_s$. This provides the link between the SVD of a data matrix and the eigenvalue decomposition of the estimate of its covariance matrix. An estimate of σ^2 can be obtained by simply averaging the $L+1-d$ smallest eigenvalues of R_N.

The remainder of this section is devoted to a brief overview of the various Subspace Fitting methods, based on specific choices of M in (34) and (35).

A. Deterministic Maximum Likelihood

If we assume that the columns of $\mathcal{V}_N$ are stationary, independent, zero-mean, circular, complex Gaussian random vectors, and that the signals corresponding to the matrix $\mathcal{S}_N$ are deterministic (as in the pole estimation problem), then maximizing the log likelihood of the data H_N with respect to Φ and $\mathcal{S}_N$ can be shown [102], [103] to be equivalent to the following minimization problem:

$$\hat{\Phi}, \hat{\mathcal{S}_N} = \arg\min_{\Phi, \mathcal{S}_N} \| H_N - \mathcal{A}(\Phi)\mathcal{S}_N \|_F^2 . \quad (37).$$

The solution of the linear part gives $\hat{\mathcal{S}}_N = \mathcal{A}^+ H_N$, and substitution into (37) reduces the minimization problem to

$$\hat{\Phi} = \arg\max_\Phi \operatorname{Tr}(\Pi_{\mathcal{A}}(\Phi)\ R_N) . \quad (38)$$

The algorithm resulting from implementation of either of the two above extremization problems is referred to as deterministic, or conditional, Maximum Likelihood (ML) [79], [98], [100]–[104]. Since $\mathcal{A}(\Phi)$ is nonlinear in the entries of Φ, its computation requires in general a complicated multidimensional search over the parameter space. Asymptotic properties of the deterministic ML method are given in [97]–[99].

Note that deterministic ML can be cast in the Subspace Fitting framework of (34) if the matrices M and T are chosen to be H_N and $\mathcal{S}_N$, respectively. Using asymptotic arguments, another connection with Subspace Fitting can be made [97]. For large N, we have $\hat{\Lambda}_n \to \sigma^2 I$ and $R_N \to \hat{E}_s \tilde{\Lambda} \hat{E}_s + \sigma^2 I$. As the trace of $\sigma^2 \Pi_{\mathcal{A}}$ is a constant, it can be omitted from the optimization and, from (38), it then follows that the ML solution is asymptotically (for large N) equivalent to the solution obtained from

$$\hat{\Phi} = \arg\min_{\Phi, T} \| \hat{E}_s \tilde{\Lambda}^{1/2} - \mathcal{A}(\Phi) T \|_F^2$$
$$= \arg\max_\Phi \operatorname{Tr}(\Pi_{\mathcal{A}}(\Phi) \hat{E}_s \tilde{\Lambda} \hat{E}_s^*) . \quad (39)$$

This is again an instance of the generic Subspace Fitting problem in (34) and (35) for $M = \hat{E}_s \tilde{\Lambda}^{1/2}$ and T of dimension $d \times d$. Using the weighting $\tilde{\Lambda}$, (39) minimizes the distance of the d-dimensional shift-invariant subspace of $\mathcal{A}$ to the signal subspace $\hat{E}_s$. In going from the formulation of (37) to that of (39), we see that the minimization has been made more compact; i.e., it involves d columns of data instead of N.

B. ESPRIT and MI-ESPRIT

In Section VI, it was mentioned that the (TLS) ESPRIT algorithm [35] was a special case of the TLS–TLS principal component approach. It has recently been noted [97], [106] that ESPRIT also has a Subspace Fitting interpretation. In particular, it can be shown that the ESPRIT algorithm is equivalent to the following least squares minimization problem:

$$\hat{\Phi} = \arg\min_{\Phi, \mathcal{A}_1} \left\| \begin{bmatrix} \hat{E}_1 \\ \hat{E}_2 \end{bmatrix} - \begin{bmatrix} \mathcal{A}_1 \\ \mathcal{A}_1 \Phi \end{bmatrix} T \right\|_F^2 \quad (40)$$

where $\hat{E}_1$ and $\hat{E}_2$ contain the rows of $\hat{E}_s$ corresponding to the two identical subarrays; e.g., for the uniform linear array described in (16), two *maximally overlapped* subarrays will yield $\hat{E}_1 = \hat{E}^{(1)}$ equal to the first L rows of $\hat{E}$, and $\hat{E}_2 = \hat{E}^{(2)}$ containing the last L rows of $\hat{E}$. The obvious connection with (34) is made by describing $\mathcal{A}(\Phi)$ as in (18) and letting

$$M = \begin{bmatrix} \hat{E}_1 \\ \hat{E}_2 \end{bmatrix} .$$

If instead of just two subarrays, the array is composed of *multiple* identical subarrays, a similar Subspace Fitting

approach may be formulated. Letting $\hat{E}_i$ represent the rows of $\hat{E}_s$ corresponding to the ith subarray, and Φ_i the diagonal matrix of phase delay factors due to propagation of the (plane) wave from the reference to the ith subarray, the most natural extension of ESPRIT is given by the following minimization problem:

$$\hat{\Phi} = \arg \min_{\Phi_1,\cdots,\Phi_p,\mathcal{A}_0} \left\| \begin{bmatrix} \hat{E}_0 \\ \hat{E}_1 \\ \hat{E}_2 \\ \vdots \\ \hat{E}_p \end{bmatrix} - \begin{bmatrix} \mathcal{A}_0 \\ \mathcal{A}_0\Phi_1 \\ \mathcal{A}_0\Phi_2 \\ \vdots \\ \mathcal{A}_0\Phi_p \end{bmatrix} T \right\|_{\mathrm{F}}^2 \quad (41)$$

where we have assumed a total of $p+1$ identical subarrays. Algorithms based on this approach have been developed in [107], [106] for the case where $\Phi_i = \Phi^i$, and in [108] for the two-dimensional (azimuth/elevation) case.

When $\Phi_i = \Phi^i$ above, a generalized Vandermonde structure results as evidenced by the multiple-shift structure in the signal subspace. The algorithm for this case is referred to as Multiple Invariance (MI) ESPRIT. One drawback relative to (41) that should be mentioned is that the elegant "closed-form" SVD solution of ESPRIT is not applicable; minimizing (41) requires a nonlinear multidimensional search when $p > 1$.

C. MUSIC

Although the Subspace Fitting paradigms of (34) and (35) are inherently multidimensional, similar one-dimensional formulations are also possible. For example, if the MUSIC [51], [109] cost function introduced in (33) is normalized by dividing by $\boldsymbol{a}^*(\phi_i)\boldsymbol{a}(\phi_i)$, it may be re-written as

$$\phi_i = \arg \max_{\phi} \mathrm{Tr}\left(\Pi_{\boldsymbol{a}}(\phi)\, \hat{E}_s \hat{E}_s^* \right), \quad (42)$$

where $\Pi_{\boldsymbol{a}}(\phi)$ is the projection onto the vector $\boldsymbol{a}(\phi)$. The only difference between (42) and (35) above with $M = \hat{E}_s$ is that while (35) implements a search for all of the parameters simultaneously, MUSIC searches for them one at a time. Thus MUSIC can be classified as a one-dimensional Subspace Fitting technique.

The asymptotic properties of MUSIC have been studied, a.o., in [98], [110]–112]. One of the interesting results of these studies is that deterministic ML and MUSIC have equivalent asymptotic performance if the sources are uncorrelated and of equal power.

D. Multidimensional MUSIC

Although relatively simple to compute, MUSIC does not give accurate results if the signals are highly correlated. This is primarily because the parameter search is done one dimension at a time. Schmidt [51] hinted at a *multi-dimensional* (MD) counterpart to MUSIC that would overcome this difficulty, and Cadzow independently developed such an algorithm [105]. The resulting algorithm, which has been referred to by several authors as MD–MUSIC, can be described by replacing M with $\hat{E}_s$ in (34):

$$\begin{aligned} \hat{\Phi} &= \arg \min_{\Phi} \| \hat{E}_s - \mathcal{A}(\Phi)T \|_{\mathrm{F}}^2 \\ &= \arg \max_{\Phi} \mathrm{Tr}\left(\Pi_{\mathcal{A}}(\Phi)\, \hat{E}_s \hat{E}_s^* \right). \end{aligned} \quad (43)$$

The motivation for the terminology "one-dimensional" and "multi-dimensional" MUSIC becomes clear when comparing (43) and (42).

E. Weighted Subspace Fitting (WSF)

In the Weighted Subspace Fitting method of Viberg and Ottersten [97], the optimality criterion is defined as (cf. (34) and (39))

$$\begin{aligned} \hat{\Phi} &= \arg \min_{\Phi} \| \hat{E}_s W^{1/2} - \mathcal{A}(\Phi)T \|_F^2 \\ &= \arg \max_{\Phi} \mathrm{Tr}\left(\Pi_{\mathcal{A}}(\Phi)\, \hat{E}_s W \hat{E}_s^* \right). \end{aligned} \quad (44)$$

In this method, a positive definite weighting matrix W is introduced. We showed earlier that the deterministic ML method corresponds to the case where $W = \tilde{\Lambda}^{1/2}$. Viberg and Ottersten have shown [97] that W can be chosen to asymptotically (for large N) minimize the estimation error variance of the parameters ϕ_i, and that the optimal choice for W is $W_{\mathrm{opt}} = \tilde{\Lambda}^2\Lambda_s^{-1}$, or a consistent estimate thereof. This choice for W has also been shown to make WSF statistically *efficient*; i.e., the WSF estimates asymptotically achieve the Cramér–Rao lower bound on the variance of the estimation error under the assumption that the signal waveforms are Gaussian random processes [113].

F. Method of Direction of Arrival Estimation (MODE)

Using the orthogonality of the estimated signal and noise subspaces defined by $\hat{E}_s$ and $\hat{E}_n$, an algorithm that is in some sense a dual of the Subspace Fitting approach in (43) can be developed. In this approach, one estimates the parameters Φ as those for which $\mathcal{A}(\Phi)$ provides the *worst* fit (i.e., most orthogonal) to the estimated noise subspace. Such an approach has been formulated in [99] by considering a criterion function of the form

$$\begin{aligned} \hat{\Phi} &= \arg \min_{\Phi} \| \hat{E}_n^* \mathcal{A}(\Phi) W_1^{1/2} \|_F^2 \\ &= \arg \min_{\Phi} \mathrm{Tr}\left(\mathcal{A}^* \hat{E}_n\, \hat{E}_n^* \mathcal{A}\, W_1 \right). \end{aligned} \quad (45)$$

The estimation error covariance is shown in [99], [114] to be minimized by the weighting $W_{1,\mathrm{opt}} = (\mathcal{A}^* U \mathcal{A})^{-1}$, where $U = E_s\, \tilde{\Lambda}^2\Lambda_s^{-1}\, E_s^*$, and the resulting algorithm using this weighting is referred to as MODE. It can easily be shown that both WSF and MODE yield results with identical asymptotic second-order error statistics [115]. Note also that the MUSIC algorithm is equivalent to (45) when $W_1 = I$, and that deterministic ML is asymptotically equivalent to (45) when $W_1^{1/2} = \mathcal{S}$ or $W_1 = P$ [114].

G. Identification via Subspace Fitting

While the description of the above algorithms has been couched in the problem of DOA estimation, the subspace fitting concept may also be directly applied to the pole estimation (i.e., system identification) problem. To see this,

recall (12), where it is shown that the column space of the matrix $R_{22}Q_2^*$ is equivalent to that of the observability matrix $\mathcal{O}$. Without measurement noise, there will exist a full rank $d \times d$ matrix T satisfying

$$E = \mathcal{O}T$$

where E represents the d principal components of $R_{22}Q_2^*$. With noise, $R_{22}Q_2^*$ is full-rank, and we are led by the subspace fitting results above to consider the minimization problem [116]

$$\begin{aligned}\hat{A}, \hat{C} &= \arg\min_{A,C} \| \hat{E}W^{1/2} - \mathcal{O}T \|_F^2 \\ &= \arg\max_{A,C} \operatorname{Tr}\left(\Pi_{\mathcal{O}}(A,C)\, \hat{E}W\hat{E}^*\right) \end{aligned} \quad (46)$$

where A and C are the matrices of the state-space model upon which $\mathcal{O}$ depends. Because of the special shift structure inherent in $\mathcal{O}$, we see that the minimization problem of (46) is isomorphic to that of the MI–ESPRIT algorithm described by (41).

As with MI–ESPRIT, implementation of (46) is somewhat more difficult than for the single-shift invariance methods of Section VI. Whereas in the latter case the estimates are obtained directly via one or two SVD's, solving (46) requires some type of search technique. However, since single-shift methods can be used to efficiently obtain an accurate initial estimate, a Newton-like gradient search will rapidly converge to the desired solution. Details of a Gauss–Newton implementation can be found in [106].

One might immediately assume that the weighting matrix W could be chosen to minimize the variance of the parameter estimates, as does the WSF algorithm. Strictly speaking, however, the optimality of W_{opt} has only been derived for the case where the observations (columns) in H are independent (as is the case in the DOA estimation problem). In the pole estimation problem, the Hankel structure of H violates this assumption. However, simulations indicate that the weighting nonetheless has the desired effect of reducing the variance of the pole estimates.

IX. Properties of the Identification Methods

The previous three sections have introduced perhaps an overwhelming number of algorithms and methods, all computing approximately the same quantity. How does one go about selecting an appropriate algorithm for a given application? Usually, the tradeoff that must be addressed in answering this question comes down to estimation accuracy versus ease of implementation and computational complexity. As a general rule, recent literature conveys that among the identification methods mentioned in the previous sections the best estimation performance is obtained by the optimal Subspace Fitting methods (e.g., WSF, MODE), whereas the most computationally efficient solutions are obtained by the Single Shift-Invariant methods of Section VI.

However, since there are often other variables and tradeoffs to consider, the question above is often not so easily answered. For example, in the array processing context, if the source signals are highly correlated (e.g., due to specular multipath: the same source is observed directly as well as via reflections), then one of the multidimensional Subspace Fitting methods must be selected. On the other hand, these methods require full knowledge of the sensor array geometry and sensor properties (i.e., the array must be calibrated), while ESPRIT exploits the special doublet structure of the array and does not require precise locations and response properties of the sensors.

To conclude the paper, we will briefly describe these tradeoffs in more detail. Our focus will be on the DOA estimation problem, since this is where most of the research in this area has been conducted. Because of the large number of techniques discussed, it is impractical to conduct and present the results of extensive simulation studies in this paper. Instead, we choose to qualitatively describe the results that others have obtained in various performance analyses. We refer the interested reader to the papers cited in this section for the actual numerical results of such simulation studies.

A. Performance Analyses

In the past several years, there has been considerable interest in investigating the statistical properties of the various methods mentioned in the previous three sections. In particular, the goal of this work has most often been to derive theoretical expressions for the variance of the pole or DOA estimates obtained by these algorithms. Since this is very difficult to do in general, the theoretical studies are usually limited to the large sample case (i.e., large N), and hence can be considered to hold only asymptotically. The picture can be completed by numerical examples for finite N. It should be noted that since these studies have concentrated on the DOA estimation problem and its corresponding assumption of independent noise samples, their results are not directly applicable to the system identification problem since the additive noise has (by construction) a Hankel structure that cannot be regarded as a set of $L \times N$ truly independent random variables.

For the DOA application, most of the algorithms mentioned in this paper have been investigated, and more or less final results have been published [97], [98], [117], [118], which we summarize below. The results have been obtained for signals modeled as stationary stochastic processes, with temporally uncorrelated zero-mean jointly Gaussian distributions. The noise is assumed to be a zero-mean temporally uncorrelated white Gaussian process that is also uncorrelated with the signals (there are a few other more minor conditions). It has been shown that ESPRIT, Deterministic Maximum Likelihood, MUSIC, WSF, etc., are all asymptotically unbiased; that is, the estimated parameters converge to the true parameters as $N \to \infty$ with probability one. However, the second-order performance (estimation error variance) of these algorithms can be very different, and it is usually this second-order performance that is used to evaluate them. This evaluation is often conducted with respect to the so-called Cramér–Rao Bound (CRB), which provides a lower bound on the estimation error variance of any unbiased estimator.

For historical reasons, the MUSIC algorithm was the first to have its performance extensively analyzed [98], [110]–[112]. Among other results obtained in these papers, it has been shown that MUSIC is a large sample realization of the deterministic Maximum Likelihood (ML) method if the signals are uncorrelated (P diagonal) [98]. Under this condition, both algorithms asymptotically achieve the deterministic signal CRB, where by asymptotic we mean for *both* large N and L. For finite L, however, neither method is statistically efficient.

For correlated signals, deterministic ML will generally outperform MUSIC. In cases where the signals are highly correlated, MUSIC will often fail to resolve all d of the signals; that is, there will be fewer than d local maxima in the MUSIC spectrum of (42). This loss of resolution can also occur when the signal-to-noise ratio is very low, or if the signals arrive from nearly coincident directions. One of the advantages of the Orthogonal Vector formulation of MUSIC, i.e., root-MUSIC, is that it does not exhibit this loss-of-resolution threshold effect[3]. Above the threshold, however, both MUSIC and root-MUSIC yield estimates with identical asymptotic variance [112]. When compared with the other Orthogonal Vector methods, root-MUSIC has the lowest estimation error variance that can be achieved by selecting only one orthogonal vector [117] from the noise subspace; in particular, it has a lower variance than Pisarenko, Min-Norm, and AAK. The same has been observed in [119] via other methods.

As with Orthogonal Vector methods, the Single Shift-Invariant techniques of Section VI are guaranteed by construction to always produce the correct number of parameter estimates. However, these algorithms will also fail when the signals are perfectly coherent, or nearly so. In this case, a failure is manifest by one of the estimates taking on what is essentially a random value. Among other results obtained for the Single Shift-Invariant methods of Section VI, it has been shown that TLS–ESPRIT and LS–ESPRIT are asymptotically equivalent [64], [118], although for small N TLS–ESPRIT has slightly better empirical performance. It has also been shown that TLS–ESPRIT is in general asymptotically less accurate that MUSIC [120], although comparing the two algorithms is somewhat unfair since they rely on a different set of assumptions about the sensor array. In particular, MUSIC requires much more information about the array, and hence its superior performance is to be expected. A recent nonasymptotic comparison between Orthogonal Vector methods (MUSIC, Min-Norm) and Single Shift-Invariant techniques (TAM, ESPRIT) has appeared in [43], [121], and supports the above asymptotic results. In these papers, closed-form expressions for first-order approximations of the perturbation of the signal and noise subspaces are derived.

One of the greatest advantages of the multidimensional Subspace Fitting methods of Section VIII is their ability to provide accurate parameter estimates in the presence of perfectly coherent signals. Of these methods, WSF and MODE possess the smallest estimation error, and in fact both methods asymptotically achieve the CRB under the Gaussian signal and noise model [97]. Thus both WSF and MODE can be thought of as large sample realizations of the Maximum Likelihood method for stochastic signals [102], [122]. An important result derived in [113], [114] states that asymptotically, deterministic ML is statistically less efficient than WSF, MODE, and stochastic ML, *independent* of whether one assumes the signals are random or not. The performance difference between these algorithms and deterministic ML can be quite large in difficult cases involving highly correlated, closely spaced signals at low signal-to-noise ratios.

Our discussion thus far in this section has implicitly focused on algorithm performance degradations due to additive noise. Another important practical consideration is the sensitivity of the algorithms to various modeling assumptions, the most important of which is the assumption of a perfectly uniform linear array of identical sensors (or, in the general case, a perfectly calibrated array response). Such analyses have been carried out for many of the algorithms discussed thus far, including MUSIC [123]–[125], ESPRIT [126], [127], deterministic ML [128], and Subspace Fitting algorithms in general [129]–[132]. One of the surprising results to come out of these studies is the fact that, under the assumption of simple Gaussian perturbations to the array response and infinite data ($N \rightarrow \infty$), MUSIC yields lower variance estimates than MODE, WSF, MD–MUSIC, and deterministic ML [125], [132]. A Subspace Fitting minimization of the form (44) can yield performance equivalent to MUSIC in such cases, but it requires a weighting matrix W quite different from that of WSF.

In the context of system identification, theoretical studies comparing several Matrix Pencil and Orthogonal Vector methods have been carried out in [68], [72], [81], [120], [133], for the harmonic retrieval problem. As already noted above, in this problem the noise matrix has a Hankel structure, and its columns cannot be regarded as being independent. This fact makes the analysis somewhat more difficult, although some results have been obtained. For example, the conclusion of the study in [120] is that MUSIC and ESPRIT perform almost equally, although usually ESPRIT is slightly better (this contrasts with the DOA problem). For signals with unknown damping factors, the Single Shift-Invariant methods of Section VI are less sensitive to noise than the Orthogonal Vector methods [68], [72]. A significant increase in accuracy for these methods is obtained by increasing L, because the error variance is proportional to $1/(L^3N)$ [120]. This is interesting because for a given set of data, one is free to choose the "blocking factor" N/L of the Hankel matrix constructed on the data, as long as $d \leq N, L$. Note, however, that the computational complexity is also proportional to L^3, and that we still require $N \gg L$. For the special case of only one signal, it has been derived [68] that the best choice for the pencil method is $(N - L)/3 \leq L \leq 2(N - L)/3$. For model

[3] Strictly speaking, root-MUSIC does have a performance threshold that results when the algorithm chooses a spurious root from its polynomial. However, this effect is manifest well beyond the MUSIC threshold.

reduction, the "noise" due to unwanted high-order modes is actually deterministic, and cannot be modeled as white noise; hence, the statistical results obtained in the DOA context are not necessarily valid. In fact, one wants to have a bound on the modeling error $\| h(z) - \hat{h}(z) \|$ in some suitable norm. At present, only the AAK method provides such a bound (in terms of the Hankel norm).

B. Computational Aspects

Although WSF, MODE, and stochastic ML are optimal in the sense of minimum asymptotic estimation error variance, the minimization of their various error criteria can only be achieved by iterative, nonlinear optimization procedures. These procedures are necessarily complex, and must be given initial estimates of reasonable quality to guarantee convergence. In [53], a Gauss–Newton descent method is proposed that can be used for both WSF and other ML techniques, and that requires $\mathcal{O}(Ld^2)$ operations per iteration. Compared with the fact that the computation of the SVD for an $(L \times N)$ matrix requires $\mathcal{O}(L^2(N + 20L))$ operations, the cost of each Gauss–Newton iteration is relatively small. The number of iterations required for convergence depends of course on the quality of the initial estimates. When ESPRIT is used to obtain the starting point, adequate convergence can be expected in two to three iterations. A number of empirical studies [53], [115] have indicated that WSF has better convergence properties than both deterministic and stochastic ML.

In comparison with Subspace Fitting and Orthogonal Vector methods (OVM), Single Shift-Invariant methods (such as ESPRIT) are computationally more attractive. The number of operations required for the SVD part of these algorithms is the same as for Subspace Fitting and OVM, but the eigenvalue computations can be done on $d \times d$ matrices in the SSI class, while the OVM requires the solution of a larger $L \times L$ eigenvalue problem, after which the d "valid" eigenvalues must be selected. Because of the regularity of the operations, the Single Shift-Invariant methods are amenable to implementation on parallel arrays of processors, of which the basic operation is a Jacobi (plain) rotation [134].

In many signal processing applications, the identification problem is solved several times, using new data as it becomes available, and discarding the older data. There is recent interest in developing efficient updating techniques, which will result in an "on-line" processor array that can update the pole or angle estimates each time a new sample vector is received ("updating") and an old vector is discarded ("downdating"). One such updating scheme, based on an approximate SVD that will converge for stationary signals, is reported in [70].

To alleviate the cost of computing the SVD, alternative but computationally less demanding decompositions of the form $X = UE_xV^*$, where E_x is not diagonal any more, are gaining interest. Recent developments are the rank-revealing QR factorization [135] which can be updated [136], and the rank-revealing URV decomposition [137], where $E_x =: R$ is upper-triangular. In this decomposition, R has a block decomposition into four blocks, such that R_{12} and R_{22} both have small Frobenius norms, and the smallest singular value of R_{11} is of the order of the smallest singular value of X that one does not want to neglect. In this way, one still obtains a decomposition of the range space of X into a signal subspace and a noise subspace. The URV decomposition can be updated and downdated at lower computational cost than the SVD, which makes it a useful tool for adaptive subspace tracking algorithms.

REFERENCES

[1] L. Ljung, *System Identification — Theory for the User.* Englewood Cliffs, NJ: Prentice Hall, 1987.

[2] T. Söderström and P. Stoica, *System Identification.* Englewood Cliffs, NJ: Prentice Hall, Inc., 1989.

[3] L. Ljung and T. Söderström, *Theory and Practice of Recursive Identification.* Cambridge, MA: MIT Press, 1983.

[4] L. L. Scharf, "The SVD and reduced-rank signal processing," in *SVD and Signal Processing, II: Algorithms, Analysis and Applications*, R. J. Vaccaro, Ed. New York: Elsevier, 1991, pp. 3–31.

[5] S. Haykin, *Adaptive Filter Theory.* Englewood Cliffs, NJ: Prentice-Hall, 1991.

[6] D.V. Bhaskar Rao and K.S. Arun, "Model based processing of signals: A state space approach," *Proc. IEEE*, vol. 80, pp. 283–309, Feb. 1992.

[7] A. Nerode, "Linear automaton transformations," *Proc. Amer. Math. Soc.*, vol. 9, pp. 541–544, 1958.

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

[9] H. P. Zeiger and A. J. McEwen, "Approximate linear realizations of given dimension via Ho's algorithm," *IEEE Trans. Automat. Cont.*, vol. AC-19, p. 153, Apr. 1974.

[10] B. C. Moore, "Singular value analysis of linear systems," in *Proc. IEEE Conf. Dec. Control*, 1979, pp. 66–73.

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

[12] T. Kailath, *Linear Systems.* Englewood Cliffs, NJ: Prentice Hall, 1980.

[13] W. J. Rugh, *Linear System Theory. A Graduate Course.* Englewood Cliffs, NJ: Prentice Hall, 1992.

[14] P. R. Halmos, *Introduction to Hilbert Space.* New York: Chelsea Pub. Co., 1951.

[15] G. Golub and C. F. Van Loan, *Matrix Computations.* Baltimore, MD: Johns Hopkins Univ. Press, 1984.

[16] R. E. Kalman, P. L. Falb, and M. A. Arbib, *Topics in Mathematical System Theory* (Int. Series in Pure and Applied Mathematics). New York: McGraw-Hill, 1970.

[17] P.A. Fuhrmann, *Linear Systems and Operators in Hilbert Space.* New York: McGraw-Hill, 1981.

[18] A. J. van der Veen and P. M. Dewilde, "Time-varying system theory for computational networks," in *Algorithms and Parallel VLSI Architectures, II*, P. Quinton and Y. Robert, Eds. New York: Elsevier, 1991, pp. 103–127.

[19] ——, "Time-varying computational networks: Realization, orthogonal embedding and structural factorization," in *Proc. SPIE, "Advanced Signal Processing Algorithms, Architectures, and Implementations," III*, F. T. Luk, Ed., vol. 1770, pp. 164–177, July 1992.

[20] A. Feintuch and R. Saeks, *System Theory: A Hilbert Space Approach.* New York: Academic Press, 1982.

[21] E. W. Kamen, P. P. Khargonekar, and K. R. Poolla, "A transfer-function approach to linear time-varying discrete-time systems," *SIAM J. Contr. Optimization*, vol. 23, no. 4, pp. 550–565, 1985.

[22] E. W. Kamen, "The poles and zeros of a linear time-varying system," *Lin. Alg. Applications*, vol. 98, pp. 263–289, 1988.

[23] B. D. O. Anderson and J. B. Moore, "Detectability and stabilizability of time-varying discrete-time linear systems," *SIAM J. Contr. Optimization*, vol. 19, no. 1, pp. 20–32, 1981.

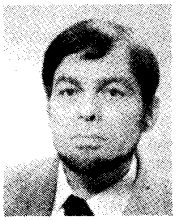
[24] I. Gohberg, M. A. Kaashoek, and L. Lerer, "Minimality and realization of discrete time-varying systems," in *Time Variant Systems and Interpolation*, I. Gohberg, Ed., vol. OT 56. Basel, Switzerland: Birkhäuser Verlag, 1992, pp. 261–296.
[25] B. De Moor, "Mathematical concepts and techniques for modeling of static and dynamic systems," Ph.D. dissertation, Kath. Univ. Leuven, Belgium, 1988.
[26] M. Verhaegen and E. F. Deprettere, "Subspace model identification," in *Algorithms and Parallel VLSI Architectures*, vol. B, E. F. Deprettere and A. J. van der Veen, Eds. New York: Elsevier, 1991, pp. 13–32.
[27] M. Verhaegen and P. M. Dewilde, "Subspace model identification. Part 1: The output error state space model identification class of algorithms," *Int. J. Contr.*, vol. 56, no. 5, pp. 1187–1210, 1992.
[28] ——, "Subspace model identification. Part 2: Analysis of the elementary output-error state-space model identification algorithm," *Int. J. Contr.*, vol. 56, no. 5, pp. 1211–1241, 1992.
[29] B. De Moor, M. Moonen, L. Vandenberghe, and J. Vandewalle, "A geometrical approach for the identification of state space models with singular value decomposition," in *Proc. IEEE ICASSP* (New York, NY), vol. 4, 1988, pp. 2244–2247.
[30] M. Moonen, B. De Moor, L. Vandenberghe, and J. Vandewalle, "On- and off-line identification of linear state-space models," *Int. J. Contr.*, vol. 49, no. 1, pp. 219–232, 1989.
[31] M. Moonen and J. Vandewalle, "QSVD approach to on- and off-line state space identification," *Int. J. Contr.*, vol. 50, no. 1, pp. 1133–1146, 1990.
[32] P. Van Overschee, B. De Moor, and J. Suykens, "Subspace algorithms for system identification and stochastic realization," in *Recent Advances in Mathematical Theory of Systems, Control, Networks and Signal Processing (Proc. Int. Symp. MTNS-91)*, vol. I, H. Kimura and S. Kodama, Eds. MITA Press, Japan, 1992, pp. 589–594.
[33] P. Van Overschee and B. De Moor, "Two subspace algorithms for the identification of combined deterministic and deterministic-stochastic systems," in *Proc. IEEE Conf. on Decision Control* (Tucson, AZ), Dec. 1992, pp. 511–516. (Also to appear in *Automatica (Special Issue on Statistical Signal Processing and Control)*. .
[34] R. Roy, A. Paulraj, and T. Kailath, "ESPRIT—A subspace rotation approach to estimation of parameters of cisoids in noise," *IEEE Trans. Acoust., Speech, Signal Processing*, vol. ASSP-34, no. 5, pp. 1340–1342, 1986.
[35] R. Roy, "ESPRIT," Ph.D. dissertation, Stanford Univ., Stanford, CA, 1987.
[36] R. Roy and T. Kailath, "ESPRIT— Estimation of signal parameters via rotational invariance techniques," *IEEE Trans. Acoust., Speech, Signal Processing*, vol. 37, pp. 984–995, July 1989.
[37] Y. Bresler and A. Macovski, "On the number of signals resolvable by a uniform linear array," *IEEE Trans. Acoust., Speech, Signal Processing*, vol. ASSP-34, pp. 1361–1375, Dec. 1986.
[38] M. Wax and I. Ziskind, "On unique localization of multiple sources by passive sensor arrays," *IEEE Trans. Acoust., Speech, Signal Processing*, vol. 37, pp. 996–1000, July 1989.
[39] R. Prony, "Essai experimental et analytique sur les lois de la dilabilité des fluides elastiques ...," *J. de l'Ecole Polytechnique*, vol. 1, no. 2, pp. 24–76, 1795.
[40] E. F. Deprettere, Ed., *SVD and Signal Processing: Algorithms, Applications and Architectures*. Amsterdam, The Netherlands: North-Holland, 1988.
[41] R. J. Vaccaro, Ed., *SVD and Signal Processing, II: Algorithms, Analysis and Applications*. New York: Elsevier, 1991.
[42] G. W. Stewart, "Perturbation theory for the singular value decomposition," in *SVD and Signal Processing, II: Algorithms, Analysis and Applications*, R. J. Vaccaro, Ed. New York: Elsevier, 1991, pp. 99–109.
[43] F. Li and R. J. Vaccaro, "Performance degradation of DOA estimators due to unknown noise fields," *IEEE Trans. Signal Processing*, vol. 40, pp. 686–690, Mar. 1992.
[44] M. S. Bartlett, "Tests of significance in factor analysis," *British J. Psych. (Statist. Sect.)*, vol. 3, pp. 77–85, 1950.
[45] ——, "A note on the multiplying factors for various χ^2 approximations," *J. Roy. Statist. Soc., Ser. B*, vol. 16, pp. 296–298, 1954.
[46] T. W. Anderson, "Asymptotic theory for principal component analysis," *Ann. Math. Statist.*, vol. 34, pp. 122–148, 1963.
[47] H. Akaike, "Information theory and an extension of the Maximum Likelihood principle," in *Proc. 2nd Int. Symp. on Information Theory*, 1973, pp. 267–281.
[48] J. Rissanen, "Modeling by shortest data description," *Automatica*, vol. 14, pp. 465–471, 1978.
[49] ——, "A universal prior for the integers and estimation by Minimum Description Length," *Annals Statist.*, vol. 11, pp. 417–431, 1983.
[50] L. C. Zhao, P. R. Krishnaiah, and Z. D. Bai, "On the detection of the number of signals when the noise covariance matrix is arbitrary," *J. Multivariate Anal.*, vol. 20, no. 1, pp. 1–25, 1986.
[51] R. O. Schmidt, "A signal subspace approach to multiple emitter location and spectral estimation," Ph.D. dissertation, Stanford University, Stanford, CA, 1981.
[52] M. Wax and T. Kailath, "Detection of signals by information theoretic criteria," *IEEE Trans. Acoust., Speech, Signal Processing* , vol. ASSP-33, pp. 387–392, Apr. 1985.
[53] M. Viberg, B. Ottersten, and T. Kailath, "Detection and estimation in sensor arrays using weighted subspace fitting," *IEEE Trans. Signal Processing*, vol. 39, pp. 2436–2449, Nov. 1991.
[54] M. Wax and I. Ziskind, "Detection of the number of coherent signals by the MDL principle," *IEEE Trans. Acoust., Speech, Signal Processing*, vol. 37, pp. 1190–1196, Aug. 1989.
[55] M. Wax, "Detection and localization of multiple sources via the stochastic signals model," *IEEE Trans. Signal Processing*, vol. 39, pp. 2450–2456, Nov. 1991.
[56] S. Van Huffel and J. Vandewalle, *The Total Least Squares problem: Computational Aspects and Analysis*. Philadelphia, PA: SIAM, 1991.
[57] B. C. Moore, "Principal component analysis in linear systems: Controllability, observability, and model reduction," *IEEE Trans. Automat. Contr.*, vol. AC-26, pp. 17–32, Feb. 1981.
[58] L. Pernebo and L. M. Silverman, "Balanced systems and model reduction," in *Proc. IEEE Conf. Dec. Control*, 1979, pp. 865–867.
[59] S. Y. Kung and D. W. Lin, "Recent progress in linear system model-reduction via Hankel matrix approximation," in *Proc. ECCTD Circuit Theory and Design*, (The Hague), 1981, pp. 222–233.
[60] S. Y. Kung, K. S. Arun, and D. V. Bhaskar Rao, "State-space and singular-value decomposition-based approximation methods for the harmonic retrieval problem," *J. Opt. Soc. Amer.*, vol. 73, pp. 1799–1811, Dec. 1983.
[61] S. Y. Kung, C. K. Lo, and R. Foka, "A Toeplitz approximation approach to coherent source direction finding," in *Proc. IEEE ICASSP*, 1986, pp. 193–196.
[62] S. Mayrargue, "ESPRIT and TAM (Toeplitz approximation method) are theoretically equivalent," in *Proc. IEEE ICASSP*, vol. 4 (New York), 1988, pp. 2456–2459.
[63] S. Mayrargue and J. P. Jouveau, "A new application of SVD to harmonic retrieval," in *SVD and Signal Processing*, E. F. Deprettere, Ed. Amsterdam, The Netherlands: North-Holland, 1988, pp. 467–472.
[64] D. V. Bhaskar Rao and K. V. S. Hari, "Performance analysis of ESPRIT and TAM in determining the direction of arrival of plane waves in noise," *IEEE Trans. Acoust., Speech, Signal Processing*, vol. 37, pp. 1990–1995, Dec. 1989.
[65] D. V. Bhaskar Rao, "Relationship between matrix pencil and state space based harmonic retrieval methods," *IEEE Trans. Acoust., Speech, Signal Processing*, vol. 38, pp. 177–179, Jan. 1990.
[66] H. Ouibrahim, D. D. Weiner, and T. K. Sarkar, "A general approach to direction finding," in *Proc. IEEE MILCON*, 1986, pp. 41.4.1–41.4.5.
[67] Y. Hua and T. K. Sarkar, "Matrix pencil method and its performance," in *Proc. IEEE ICASSP*, 1988, pp. 2476–2479.
[68] ——, "Matrix pencil method for estimating parameters of exponentially damped/undamped sinusoids in noise," *IEEE Trans. Acoust., Speech, Signal Processing*, vol. 38, pp. 814–824, May 1990.
[69] M. Moonen, P. Van Dooren, and J. Vandewalle, "An SVD updating algorithm for subspace tracking," *SIAM J. Matrix Anal. Appl.*, vol. 13, no. 4, 1992.
[70] M. Moonen, F. Van Poucke, and E. Deprettere, "Parallel and adaptive high resolution direction finding," in *Proc. SPIE: Advanced Signal Processing Algorithms, Architectures and Implementations III*, F. Luk, Ed., 1992, pp. 219–230.
[71] M. D. Zoltowski and D. Stavrinides, "Sensor array signal

processing via a Procrustus rotations based eigenanalysis of the ESPRIT data pencil," *IEEE Trans. Acoust., Speech, Signal Processing*, vol. 37, pp. 832–861, June 1989.
[72] Y. Hua and T. K. Sarkar, "On SVD for estimating generalized eigenvalues of singular matrix pencil in noise," *IEEE Trans. Signal Processing*, vol. 39, pp. 892–900, Apr. 1991.
[73] Y. T. Chan and R. P. Langford, "Spectral estimation via the high-order Yule-Walker equations," *IEEE Trans. Acoust., Speech, Signal Processing*, vol. ASSP-30, pp. 689–698, Oct. 1982.
[74] P. Stoica, R. L. Moses, T. Söderström, and J. Li, "Optimal high-order Yule-Walker estimation of sinusoidal frequencies," *IEEE Trans. Signal Processing*, vol. 39, pp. 1360–1368, June 1991.
[75] V. M. Adamjan, D. Z. Arov, and M. G. Krein, "Analytic properties of Schmidt pairs for a Hankel operator and the generalized Schur-Takagi problem," *Mat. USSR Sbornik*, vol. 15, no. 1, pp. 31–73, 1971.
[76] M. Bouvet and H. Clergeot, "Eigen- and singular value decomposition techniques for the solution of harmonic retrieval problems," in *SVD and Signal Processing*, E. F. Deprettere, Ed. Amsterdam, The Netherlands: North-Holland, 1988.
[77] V. Pisarenko, "The retrieval of harmonics from a covariance function," *Geophys. J. Roy. Astron. Soc.*, vol. 33, pp. 347–366, 1973.
[78] R. Kumaresan and D. W. Tufts, "Estimating the parameters of exponentially damped sinusoids and pole-zero modeling in noise," *IEEE Trans. Acoust., Speech, Signal Processing*, vol. ASSP-30, pp. 833–840, Dec. 1982.
[79] D. W. Tufts and R. Kumaresan, "Estimation of frequencies of multiple sinusoids: Making Linear Prediction perform like Maximum Likelihood," *Proc. IEEE*, vol. 70, pp. 975–989, Sept. 1982.
[80] R. Kumaresan, "On the zeros of the Linear Prediction-error filter for deterministic signals," *IEEE Trans. Acoust., Speech, Signal Processing*, vol. ASSP-31, pp. 217–220, Feb. 1983.
[81] B. Porat and B. Friedlander, "On the accuracy of the Kumarsean-Tufts method for estimating complex damped exponentials," *IEEE Trans. Acoust., Speech, Signal Processing*, vol. ASSP-35, pp. 231–235, Feb. 1987.
[82] F. Li, R. J. Vaccaro, and D. W. Tufts, "Min-Norm Linear Prediction for arbitrary sensor arrays," in *Proc. IEEE ICASSP* (Glasgow), May 1989, pp. 2613–2616.
[83] D. W. Tufts, R. J. Vaccaro, and A. C. Kot, "Analysis of estimation of signal parameters by Linear Prediction at high SNR using matrix approximations," in *Proc. IEEE ICASSP* (Glasgow), May 1989, pp. 2194–2197.
[84] E. M. Dowling and R. D. DeGroat, "The equivalence of the Total Least Squares and Minimum Norm methods," *IEEE Trans. Signal Processing*, vol. 39, pp. 1891–1892, Aug. 1991.
[85] G. Cybenko, "Locations of zeros of predictor polynomials," *IEEE Trans. Acoust., Speech, Signal Processing*, vol. ASSP-27, no. 1, pp. 235–237, 1982.
[86] A. Bultheel, "Recursive rational approximation," Ph.D. dissertation, Catholic University Louvain, Louvain, Belgium, 1979.
[87] K. Glover, "All optimal Hankel-norm approximations of linear multivariable systems," *Int. J. Contr.*, vol. 39, no. 6, pp. 1115–1193, 1984.
[88] Y. V. Genin and S. Y. Kung, "A two-variable approach to the model reduction problem with Hankel norm criterion," *IEEE Trans. Circuits Syst.*, vol. CAS-28, pp. 912–924, Sept. 1981.
[89] Ph. Delsarte, Y. Genin, and Y. Kamp, "On the role of the Nevanlinna-Pick problem in circuit theory and design," *Circuit Theory Appl.*, vol. 9, pp. 177–187, 1981.
[90] M. H. Gutknecht, J. O. Smith, and L. N. Trefethen, "The Carathéodory-Fejér method for recursive digital filter design," *IEEE Trans. Acoust., Speech, Signal Processing*, vol. ASSP-31, pp. 1417–1426, Dec. 1983.
[91] P. M. Dewilde and E. F. Deprettere, "Singular value decomposition: an introduction," in *SVD and Signal Processing*, E. F. Deprettere, Ed. Amsterdam, The Netherlands: North-Holland, 1988, pp. 3–41.
[92] L. M. Silverman and M. Bettayeb, "Optimal approximation of linear systems," in *Proc. 1980 Joint Autom. Control Conf.*, 1980.
[93] H. Ozbay, "Computing the singular values and vectors of a Hankel operator," in *SVD and Signal Processing, II: Algorithms, Analysis and Applications*, R. J. Vaccaro, Ed. New York: Elsevier, 1991, pp. 455–469.
[94] S. Y. Kung and D. W. Lin, "A state-space formulation for optimal Hankel-norm approximations," *IEEE Trans. Automat. Contr.*, vol. AC-26, pp. 942–946, Aug. 1981.
[95] J. A. Ball, I. Gohberg, and L. Rodman, *Interpolation of Rational Matrix Functions*, vol. OT 45 of *Operator Theory: Advances and Applications*. Basel, Switzerland: Birkhäuser Verlag, 1990.
[96] P. M. Dewilde and A. J. van der Veen, "On the Hankel-norm approximation of upper-triangular operators and matrices," *Integral Equations and Operator Theory*, vol. 17, no. 1, pp. 1–45, 1993.
[97] M. Viberg and B. Ottersten, "Sensor array processing based on subspace fitting," *IEEE Trans. Signal Processing*, vol. 39, pp. 1110–1121, May 1991.
[98] P. Stoica and A. Nehorai, "MUSIC, Maximum Likelihood and Cramér-Rao bound," *IEEE Trans. Acoust., Speech, Signal Processing*, vol. 37, pp. 720–741, May 1989.
[99] P. Stoica and K. C. Sharman, "Maximum Likelihood methods for direction-of-arrival estimation," *IEEE Trans. Acoust., Speech, Signal Processing*, vol. 38, pp. 1132–1143, July 1990.
[100] D. W. Tufts and R. Kumaresan, "Improved spectral resolution II," in *Proc. IEEE ICASSP* (Denver, CO), Apr. 1980, pp. 592–597.
[101] R. Kumaresan, L. L. Scharf, and A. K. Shaw, "An algorithm for pole-zero modeling and spectral analysis," *IEEE Trans. Acoust., Speech, Signal Processing*, vol. ASSP-34, pp. 637–640, June 1986.
[102] J. F. Böhme, "Estimation of spectral parameters of correlated signals in wavefields," *Signal Processing*, vol. 11, pp. 329–337, Dec. 1986.
[103] M. Wax, "Detection and estimation of superimposed signals," Ph.D. dissertation, Stanford University, Stanford, CA, 1985.
[104] Y. Bresler and A. Macovski, "Exact Maximum Likelihood parameter estimation of superimposed exponential signals in noise," *IEEE Trans. Acoust., Speech, Signal Processing*, vol. ASSP-34, pp. 1081–1089, Oct. 1986.
[105] J. A. Cadzow, "A high resolution direction-of-arrival algorithm for narrow-band coherent and incoherent sources," *IEEE Trans. Acoust., Speech, Signal Processing*, vol. 36, pp. 965–979, July 1988.
[106] A. Swindlehurst, B. Ottersten, R. Roy, and T. Kailath, "Multiple invariance ESPRIT," *IEEE Trans. Signal Processing*, vol. 40, pp. 867–881, Apr. 1992.
[107] R. Roy, B. Ottersten, A. L. Swindlehurst, and T. Kailath, "Multiple invariance ESPRIT," in *Proc. 22-nd Asilomar Conf. Sign., Syst., Computing*, 1988, pp. 583–587.
[108] A. Swindlehurst and T. Kailath, "Azimuth/elevation direction finding using regular array geometries," *IEEE Trans. Aerosp. Electron. Syst.*, vol. 29, pp. 145–156, Jan. 1993.
[109] G. Bienvenue and L. Kopp, "Optimality of high resolution array processing using the eigensystem approach," *IEEE Trans. Acoust., Speech, Signal Processing*, vol. ASSP-31, no. 5, pp. 1235–1247, 1983.
[110] A. J. Barabell, "Improving the resolution performance of eigenstructure-based direction finding algorithms," in *Proc. IEEE ICASSP*, 1983, pp. 336–339.
[111] M. Kaveh and A. J. Barabell, "The statistical performance of the MUSIC and minimum-norm algorithms in resolving plane waves in noise," *IEEE Trans. Acoust., Speech, Signal Processing*, vol. ASSP-34, pp. 331–341, Apr. 1986. (Corrections in vol. ASSP-34, no. 6, 1986.)
[112] D. V. Bhaskar Rao and K. V. S. Hari, "Performance analysis of root-MUSIC," *IEEE Trans. Acoust., Speech, Signal Processing*, vol. 37, pp. 1939–1949, Dec. 1989.
[113] B. Ottersten, M. Viberg, and T. Kailath, "Analysis of subspace fitting and ML techniques for parameter estimation from sensor array data," *IEEE Trans. Signal Processing*, vol. 40, pp. 590–600, Mar. 1992.
[114] P. Stoica and A. Nehorai, "Performance study of conditional and unconditional direction-of-arrival estimation," *IEEE Trans. Acoust., Speech, Signal Processing*, vol. 38, pp. 1783–1795, Oct. 1990.
[115] B. Ottersten, M. Viberg, P. Stoica, and A. Nehorai, "Exact and large sample ML techniques for parameter estimation and detection in array processing," in *Radar Array Processing*, S. Haykin, Ed. New York: Springer-Verlag, 1991, ch. 4.
[116] A. Swindlehurst, R. Roy, B. Ottersten, and T. Kailath, "A subspace fitting approach for identification of linear state space models," submitted to *IEEE Trans. Automat. Contr.*, 1993.

[117] P. Stoica and A. Nehorai, "MUSIC, Maximum Likelihood, and Cramér-Rao bound: Further results and comparisons," *IEEE Trans. Acoust., Speech, Signal Processing*, vol. 38, pp. 2140–2150, Dec. 1990.
[118] B. Ottersten, M. Viberg, and T. Kailath, "Performance analysis of the Total Least Squares ESPRIT algorithm," *IEEE Trans. Signal Processing*, vol. 39, pp. 1122–1135, May 1991.
[119] H. Krim, P. Forster, and J. G. Proakis, "Operator approach to performance analysis of root-MUSIC and root-Min-Norm," *IEEE Trans. Signal Processing*, vol. 40, pp. 1687–1696, July 1992.
[120] P. Stoica and T. Söderström, "Statistical analysis of MUSIC and subspace rotation estimates of sinusoidal frequencies," *IEEE Trans. Signal Processing*, vol. 39, pp. 1836–1847, Aug. 1991.
[121] F. Li and R. J. Vaccaro, "Performance analysis of state-space realization (TAM) and ESPRIT algorithms for DOA estimation," *IEEE Trans. Antennas Propagat.*, vol. 39, pp. 418–423, Mar. 1991.
[122] W. J. Bangs, "Array processing with generalized beamformers," Ph.D. dissertation, Yale University, New Haven, CT, 1971.
[123] K. M. Wong, R. S. Walker, and G. Niezgoda, "Effects of random sensor motion on bearing estimation by the MUSIC algorithm," *Proc. Inst. Elec. Eng.*, vol. 135, pt. F, pp. 233–250, June 1988.
[124] B. Friedlander, "A sensitivity analysis of the MUSIC algorithm," *IEEE Trans. Acoust., Speech, Signal Processing*, vol. 38, pp. 1740–1751, Oct. 1990.
[125] A. L. Swindlehurst and T. Kailath, "A performance analysis of subspace-based methods in the precence of model errors —Part 1: The MUSIC algorithm,"*IEEE Trans. Signal Processing*, vol. 40, pp. 1758–1774, July 1992.
[126] F. Li, R. Vaccaro, and D. Tufts, "Unified performance analysis of subspace-based estimation algorithms," in *Proc. IEEE ICASSP* (Albuquerque, NM), 1990, vol. 5, pp. 2575–2578.
[127] A. Swindlehurst and T. Kailath, "On the sensitivity of the ESPRIT algorithm to non-identical subarrays," *Sādhanā, Academy Proc. in Eng. Sciences*, vol. 15, pp. 197–212, Nov. 1990.
[128] B. Friedlander, "Sensitivity of the Maximum Likelihood direction finding algorithm," *IEEE Trans. Aerosp. Electron. Syst.*, vol. 26, pp. 953–968, Nov. 1990.
[129] F. Li and R. Vaccaro, "Statistical comparison of subspace based DOA estimation algorithms in the presence of sensor errors," in *Proc. 5th Acoust., Speech, Signal Processing Spectral Estimation Workshop* (Rochester, NY), Oct. 1990, pp. 327–331.
[130] A. Swindlehurst and T. Kailath, "An analysis of subspace fitting algorithms in the presence of sensor errors," in *Proc. IEEE ICASSP* (Albuquerque, NM), 1990, vol. 5, pp. 2647–2650.
[131] A. Swindlehurst, "Robust algorithms for direction-finding in the presence of model errors," in *Proc. 5th Acoust., Speech, Signal Processing Workshop on Spectral Estimation and Modeling* (Rochester, NY), Oct. 1990, pp. 362–366 .
[132] A. Swindlehurst and T. Kailath, "A performance analysis of subspace-based methods in the presence of model errors—Part 2: Multidimensional algorithms," *IEEE Trans. Signal Processing*, vol. 41, pp. 2882–2890, Sept. 1993.
[133] Y. Hua and T. K. Sarkar, "Perturbation analysis of TK method for harmonic retrieval problems," *IEEE Trans. Acoust., Speech, Signal Processing*, vol. 36, pp. 228–240, Feb. 1988.
[134] A. J. van der Veen and E. F. Deprettere, "Parallel VLSI matrix pencil algorithm for high resolution direction finding," *IEEE Trans. Acoust., Speech, Signal Processing*, vol. 39, pp. 383–394, Feb. 1991.
[135] T. F. Chan, "Rank revealing QR factorizations," *Lin. Alg. Appl.*, vol. 88/89, pp. 67–82, 1987.
[136] C. H. Bischof and G. M. Schroff, "On updating signal subspaces," *IEEE Trans. Signal Processing*, vol. 40, pp. 96–105, Jan. 1992.
[137] G. W. Stewart, "An updating algorithm for subspace tracking," *IEEE Trans. Signal Processing*, vol. 40, pp. 1535–1541, June 1992.

Alle-Jan van der Veen (Student Member, IEEE) was born in The Netherlands in 1966. He graduated (*cum laude*) from the Department of Electrical Engineering, Delft University of Technology, in 1988.

He is currently with the Network Theory Section at the Delft University, where he has recently received the Ph.D. degree (*cum laude*). His research interests are in the areas of system theory, in particular system identification, model reduction, and time-varying system theory, and in parallel algorithms for linear algebra. He has organized two workshops in the area of signal processing, and is the co-editor of the book *Algorithms and Parallel VLSI Architectures.*



Ed F. Deprettere (Senior Member, IEEE) was born in Roeselare, Belgium, on August 10, 1944. He received the M.S. degree from Ghent State University, Ghent, Belgium, in 1968, and the Ph.D. degree from Delft University of Technology (DUT), Delft, The Netherlands, in 1981.

In 1970, he became a Research Assistent and Lecturer at the DUT, where he is now Associate Professor in the Department of Electrical Engineering, Network Theory Section, Signal Processing Group. His current research interests are in modern signal processing: algorithms, VLSI architectures, and applications, and in methodologies for the mapping of parallel signal processing algorithms, network graphs, and numerical computations onto silicon. He is the editor of the books *SVD and Signal Processing: Algorithms, Architectures and Applications* and *Algorithms and Parallel VLSI Architectures.* He is on the editorial board of the *IEEE Transactions on Signal Processing*, the *Journal of VLSI Signal Processing*, and *Integration, the VLSI Journal.* He coauthored a paper that received a 1989 IEEE SP award.

A. Lee Swindlehurst was born on March 10, 1960 in Boulder City, NV. He received the B.S. and M.S. degrees in electrical engineering from Brigham Young University, Provo, UT, in 1985 and 1986, respectively, and the Ph.D. degree also in electrical engineering from Stanford University, Stanford, CA, in 1991.

He is currently an Assistant Professor in the Electrical and Computer Engineering Department at Brigham Young University. His research interests are in the general areas of signal processing, estimation, and control theory. In particular, his published research has focussed on problems in sensor array signal processing, state-space system identification, and bispectral estimation.