A Subspace Approach to Stochastic Realization and Positive Real Sequences

Peter Van Overschee Bart De Moor *

ESAT Department of Electrical Engineering Katholieke Universiteit Leuven
Kardinaal Mercierlaan 94 B-3001 Leuven (Heverlee) BELGIUM
tel : 32/16/220931 fax : 32/16/221855
email : vanoverschee@esat.kuleuven.ac.be
ESAT-SISTA Report 1991-02

September 11, 1996

Abstract

Until recently, all linear stochastic realization algorithms were based on the construction of a (block) Hankel matrix with Markov parameters (output covariances). In this paper, we'll present a new subspace stochastic realization technique (data driven) that makes use of the principal angles and directions between subspaces and in doing so avoids estimating the output covariances explicitly.

The second part of this paper deals with a practical problem induced by the positive real lemma : the identified covariance sequence does not only have to be stable, but also positive real. Our new solution makes use of an algorithm similar to the one for the calculation of the L_∞ norm of a transfermatrix and guarantees positive realness.
Keywords : singular value decomposition, principal angles and directions, canonical correlations, level set, L_∞ norm computation, discrete Riccati equation.

1 Introduction

Obtaining mathematical models for random processes is an important problem in signal processing that has a variety of applications. One important application is the modelling of stochastic systems for control purposes. If the stochastic disturbances are known, they can be taken into account during optimal control system design. In this way, stochastic modelling becomes an integral part in the design and analysis of model based controllers.

This paper considers linear, finite dimensional, stationary stochastic processes. A mathematical model for such a process is a linear, time invariant system which, when excited with white noise, produces an output which is equivalent, up to second order statistics, to the given process. This work uses state space representations for this process, and considers only discrete time processes.

*Research Associate of the Belgian National Fund for Scientific Research and Assistant Professor at the Department of Electrical Engineering of the Katholieke Universiteit Leuven

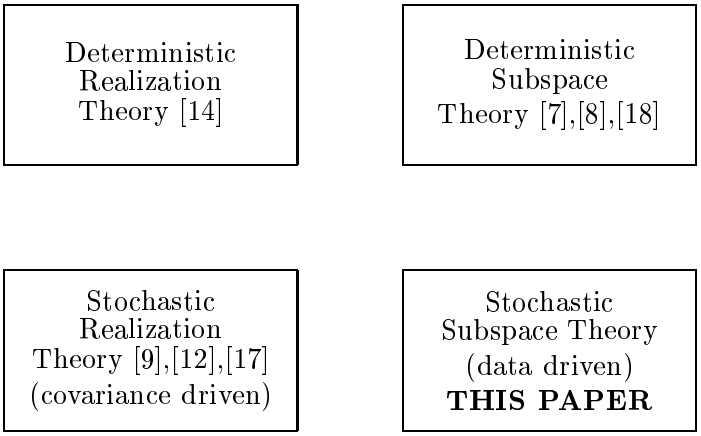


Figure 1: Overview of different deterministic and stochastic identification Schemes

1. A first contribution of this paper is the development of a subspace approach to identify the covariance sequence. For the considered type of stochastic processes, this output covariance sequence can be modelled as the impulse response of a deterministic linear time invariant system. Therefor the most popular way to identify the covariance sequence is the following two step procedure, called "stochastic realization" : estimate the covariance sequence from the given data *and* identify a model for this covariance sequence using deterministic realization theory. The subspace approach avoids estimating this covariance sequence explicitly. It identifies a state space model directly from the data (data driven) through an analysis of the principal angles and directions and the solution of a Sylvester equation. Figure 1 shows an overview of the different identification algorithms. Deterministic and stochastic realization theory have reached a state of maturity by now. Deterministic subspace techniques have been studied in [7], [8], [18]. This paper fills in the box of stochastic subspace identification.

2. This paper also highlights a deep problem that occurs when working with real data, that satisfy certain assunptions only in an approximate way or assymptotically. This problem is often overlooked in theoretical identification algorithms where 'idealized' assumptions are necessary to be able to invoke sophisticated mathematical machinery. In this paper we are confronted with such a problem in the form of the positive real lemma. This problem is encountered during the second step of the stochastic identification procedure : identifying a model that, if driven by white noise, would generate the identified covariance sequence. Faurre [12] showed with his "positive real lemma", that this model only exists if the covariance sequence is positive real. When working with real data and MIMO systems, it very often turns out that the covariance identification algorithm (e.g. subspace approach,[3], [9], [12], [17], [19]) returns a sequence that does not satisfy this lemma. The basic problem is thus to perturb the identified covariance sequence, until it becomes positive real, and can be split in its spectral factors.

   In this paper, we'll present a new algorithm that enables us to adjust the identified covariance sequence until it is positive real. It is based on a perturbation of the lag zero covariance Λ_0. This heuristically inspired algorithm is guaranteed to return a positive

real sequence in all cases. It makes use of an algorithm that is similar to the one by Boyd et al. [5] for calculating the L_∞ norm. Our algorithm calculates the minimum, over the unit circle, of the minimum eigenvalue of the real part of a transfer matrix.

3. Finally, we confront the performance of 3 different stochastic realization techniques, with our new stochastic subspace approach. On the same algorithm we also illustrate the positive realness enforcement algorithm.

This paper is organized as follows : In section 2, we introduce stochastic state space models and state the general stochastic identification problem. In section 3, the solution for the Markovian representation problem derived by Faurre [12] is restated. Section 4 looks at a realization approach for the covariance identifiaction problem, but in way that is different from the classical presentation. In section 5, a new subspace technique for identifying the covariance sequence is presented. In section 6, we describe the problems that are encountered with the positive real lemma. Section 7 presents a new algorithm for approximating the covariance sequence with a positive real sequence. An example will sustain the validity of the new algorithms in section 8. Finally, the conclusions can be found in section 9.

2 Stochastic State Space Models and the General Stochastic Identification Problem

In this section we'll describe the general linear time invariant stochastic state space model together with two special forms of this model (forward and backward innovation). We also state the general stochastic identification problem.

2.1 The General Model

A general state space model for a linear stochastic process is given by [12] :

$$x_{k+1} = Ax_k + w_k \tag{1}$$
$$y_k = Cx_k + v_k \tag{2}$$
$$\mathbf{E}\left[\begin{pmatrix} w_k \\ v_k \end{pmatrix}\begin{pmatrix} w_l^t & v_l^t \end{pmatrix}\right] = \begin{pmatrix} Q & S \\ S^t & R \end{pmatrix}\delta_{kl} \tag{3}$$

With $A \in \Re^{n\times n}$ and $C \in \Re^{l\times n}$, $v_k \in \Re^l$ and $w_k \in \Re^l$, $\mathbf{E}$ the expected value operator and δ_{kl} the Kronecker delta : $\delta_{kl} = 1$ iff $k = l$ and zero otherwise. Formula (3) expresses that w_k and v_k are white noise sequences, which we will asume to be zero mean : $\mathbf{E}(v_k) = \mathbf{E}(w_k) = 0$.

Statistics of the General Model

Assume that the process is stationary i.e. A is stable, $\mathbf{E}(x_k) = 0$ and the state covariance matrix is independant of time : $\Sigma \stackrel{\text{def}}{=} \mathbf{E}(x_k x_k^t) \in \Re^{n\times n}$ for all k. Furthermore, we also make the standard assumption that : $\mathbf{E}(w_k x_k^t) = 0$, $\mathbf{E}(v_k x_k^t) = 0$. With $\Lambda_0 \stackrel{\text{def}}{=} \mathbf{E}(y_k y_k^t) \in \Re^{l\times l}$ and $G \stackrel{\text{def}}{=} \mathbf{E}(x_{k+1} y_k^t) \in \Re^{n\times l}$, it can be proven that [12] :

$$\Sigma = A\Sigma A^t + Q \quad (4)$$
$$\Lambda_0 = C\Sigma C^t + R \quad (5)$$
$$G = A\Sigma C^t + S \quad (6)$$
$$Q > 0 \ , \ R > 0 \ , \ \begin{pmatrix} Q & S \\ S^t & R \end{pmatrix} \geq 0 \quad (7)$$

With these definitions we find the following expression for the output covariance matrices :

$$\Lambda_i \stackrel{\text{def}}{=} \mathbf{E}(y_{k+i}y_k^t) = CA^{i-1}G|_{t_i} + \Lambda_0\delta_{i0} + G^t(A^t)^{-i-1}C^t|_{t_{-i}} \ , \ k = -\infty, \ldots, \infty \quad (8)$$

with $t_i = 1$ when $i > 0$ and $t_i = 0$ otherwise. We call this sequence Λ_i the *covariance sequence*.

2.2 Forward and Backward Innovation Models

There are two very important special forms of the general stochastic model : the forward and backward innovation model. To obtain the forward innovation model, we apply Kalman filtering to the system of (1-3), resulting in [9] :

$$\hat{x}_{k+1} = A\hat{x}_k + Be_{f_k} \quad (9)$$
$$y_k = C\hat{x}_k + De_{f_k} \quad (10)$$

With $B \in \Re^{r\times l}, D \in \Re^{l\times l}$. Also : e_{f_k} is a zero mean white noise sequence with $\mathbf{E}(e_{f_i}e_{f_k}^t) = I_l\delta_{ik}$, and $\mathbf{E}(\hat{x}_k\hat{x}_k^t) = P$. The last two formulas are independent of k because of stationarity. P is the solution of the forward Riccati equation :

$$P = APA^t + (G - APC^t)(\Lambda_0 - CPC^t)^{-1}(G - APC^t)^t \quad (11)$$

The solution to the forward Riccati equation (11) can be found by solving the generalized eigenvalue problem :

$$\begin{pmatrix} A^t - C^t\Lambda_0^{-1}G^t & 0 \\ -G\Lambda_0^{-1}G^t & I_n \end{pmatrix}\begin{pmatrix} W_1 \\ W_2 \end{pmatrix} = \begin{pmatrix} I_n & -C^t\Lambda_0^{-1}C \\ 0 & A - G\Lambda_0^{-1}C \end{pmatrix}\begin{pmatrix} W_1 \\ W_2 \end{pmatrix}\Lambda$$

and putting $P = W_2W_1^{-1}$. Λ contains the n stable eigenvalues of the generalized eigenvalue pencil. In [4], a numerical more robust algorithm based on the Schur decomposition is presented. The backward innovation model was derived in [16] as :

$$\hat{z}_{k-1} = A^t\hat{z}_k + Ee_{b_k} \quad (12)$$
$$y_k = G^t\hat{z}_k + Fe_{b_k} \quad (13)$$

With $E \in \Re^{r\times l}, F \in \Re^{l\times l}$. Also : e_{b_k} is a zero mean white noise sequence with $\mathbf{E}(e_{b_i}e_{b_k}^t) = I_l\delta_{ik}$, and $\mathbf{E}(\hat{z}_k\hat{z}_k^t) = N$. Once again, these formulas are independent of k. N is the solution of the backwards Riccati equation :

$$N = A^tNA + (C^t - A^tNG)(\Lambda_0 - G^tNG)^{-1}(C^t - A^tNG)^t \quad (14)$$

This equation can be solved in a similar way as the forward Riccati equation. The matrices B, D, E and F from formulas (9-13) can be found from A, G, C, Λ_0, P, N through the application of following formulas :

$$D = (\Lambda_0 - CPC^t)^{\frac{1}{2}} \qquad F = (\Lambda_0 - G^tN^{-1}G)^{\frac{1}{2}}$$
$$B = (G - APC^t)D^{-1} \qquad E = (C^t - A^tN^{-1}G)F^{-1}$$

2.3 The General Stochastic Identification Problem

The general stochastic identification problem can be formulated as follows :
Given the output measurements of a stochastic system ($\mathbf{y_k}$), solve :

- **The covariance identification problem :** *determine A, G, C, Λ_0 from the given data. This can be done in two ways : via the classical stochastic realization theory (section 4), or through the new stochastic subspace theory (section 5).*
- **The Markovian representation problem :** *Given A, G, C, Λ_0, find all the matrices $\Sigma > 0, Q, R, S$, so that equations (4-7) are satisfied. Or in other words : find <u>all</u> state space representations that will have the same statistics up to second order as the given sequence $\mathbf{y_k}$. This will be the topic of section 3.*

3 Markovian Representation Problem

In this section, we'll describe how to find <u>all</u> the stochastic models that have the same second order statistics as the given data $\mathbf{y_k}$, starting from the matrices A, G, C, Λ_0.
We presume that A, G, C, Λ_0 are given (section 4 and 5 deal with the problem of identifying these matrices). The Markovian representation problem was solved by Faurre, and we summarize the major results :

Theorem 1 Faurre's theorem *[12]*
The set of stochastic models that generate a given output covariance matrix sequence Λ_i as in (8) is characterized as follows :

1. *The matrices A, C and G are unique up to within a similarity transformation.*
2. *The set of all state covariance matrices Σ is closed and bounded :*

   $$P \leq \Sigma \leq N^{-1}$$

   where P and N are the solutions of the forward (11) respectively backward (14) Riccati equation.
3. *For every state covariance matrix Σ satisfying these bounds, the associated matrices R, S and Q follow from equations (4-6).*

The matrices B, D, E and F can then be found from the formulas of section 2.2.

4 A Realization Approach to the Covariance Identification Problem

In this section, we'll present a alternative way to derive the Principal Component algorithm [3], which is the simpelest of the stochastic realization algorithms.
Let us first assume that the exact covariances Λ_i are available. The block Hankel matrix H_i, formed with elements of this sequence is defined as follows :

$$H_i = \begin{pmatrix} \Lambda_1 & \Lambda_2 & \dots & \Lambda_i \\ \Lambda_2 & \Lambda_3 & \dots & \Lambda_{i+1} \\ \dots & \dots & \dots & \dots \\ \Lambda_i & \Lambda_{i+1} & \dots & \Lambda_{2i-1} \end{pmatrix}$$

If we substitute the Λ_i's by expression (8) this matrix H_i can be split up in an extended observability and controllability matrix as follows :

$$H_i = \mathcal{O}_i \mathcal{C}_i = \begin{pmatrix} C \\ CA \\ \dots \\ CA^{i-1} \end{pmatrix} \begin{pmatrix} G & AG & \dots & A^{i-1}G \end{pmatrix}$$

Clearly H_i is rank deficient and has a rank equal to the order of the stochastic system n. The parameters A, G, C can be found through application of any of the deterministic realization techniques [14], [15].

In reality however, only estimates of the sequence Λ_i are available. From now on, we denote all estimated and identified matrices with a tilde.
Define the block Toeplitz matrix Y_{past} and the block Hankel matrix Y_{future} formed with the outputs of the stochastic system y_k as :

$$\begin{aligned} Y_{past} &= \begin{pmatrix} y_{k-1} & y_k & \dots & y_{k+j-2} \\ y_{k-2} & y_{k-1} & \dots & y_{k+j-3} \\ \dots & \dots & \dots & \dots \\ y_{k-i} & y_{k-i+1} & \dots & y_{k+j-i-1} \end{pmatrix} \\ Y_{future} &= \begin{pmatrix} y_k & y_{k+1} & \dots & y_{k+j-1} \\ y_{k+1} & y_{k+2} & \dots & y_{k+j-2} \\ \dots & \dots & \dots & \dots \\ y_{k+i-1} & y_{k+i} & \dots & y_{k+j+i-2} \end{pmatrix} \end{aligned}$$

If we multiply these two matrices Y_{past} and Y_{future}, we obtain :

$$\begin{aligned} \tilde{H}_i &= \frac{1}{j} Y_{future} Y_{past}^t \\ &= \frac{1}{j} \begin{pmatrix} \sum_{p=0}^{j-1} y_{k+p} y_{k-1+p}^t & \dots & \sum_{p=0}^{j-1} y_{k+p} y_{k-i+p}^t \\ \dots & \dots & \dots \\ \sum_{p=0}^{j-1} y_{k+i-1+p} y_{k-1+p}^t & \dots & \sum_{p=0}^{j-1} y_{k+i-1+p} y_{k-i+p}^t \end{pmatrix} \end{aligned} \tag{15}$$

Because of ergodicity, we have the following equality in the limit :

$$\lim_{j \to \infty} (\frac{1}{j} \sum_{p=0}^{j-1} y_{r+p} y_{s+p}^t) = \mathbf{E}(y_r y_s^t) = \Lambda_{r-s}$$

If we apply this on formula (15), we get the following equality :

$$\lim_{j \to \infty} \tilde{H}_i = \mathbf{E}(\frac{Y_{future} Y_{past}^t}{j}) = H_i$$

So $\tilde{H}_i$ is easy to construct out of the given data y_k and forms an estimate of H_i.
The stochastic realization approach for the identification of the covariance sequence would then go as follows :

1. Form the block Toeplitz Y_{past} and block Hankel Y_{future} matrix out of the given data y_k. Form also the matrix $\tilde{\Lambda}_0 = \sum_{l=0}^{j-1} y_{k+l} y_{k+l}^t / j$, which is an estimate for Λ_0.

2. Multiply both matrices to find $\tilde{H}_i$.
3. The rank of $\tilde{H}_i$ is the order of the system n.
4. Apply the realization theory of section 4 on $\tilde{H}_i$ to find the matrices $\tilde{A}, \tilde{G}, \tilde{C}$.

5 A Subspace Algorithm for the Covariance Identification Problem

This section treats a new way to tackle the covariance identification problem. It does not only make use of the principal angles between Y_{past} and Y_{future}, but also of the principal directions. The subspace approach is inspired by the fact that identification from Markovian parameters can be a bad idea : fast poles die out rapidly and consequently, the Markov parameters are drown in the noise. This problem is avoided when using subspace algorithms [7], [8], because they make use of the input-output data immediately (data driven), instead of going through the whole process of estimating the Markov-parameters (covariance driven). The advantage is clearly that the data are not transformed into covariances that die out rapidly (8), but instead, the data is used in its raw form. Consider the principal angles and directions between Y_{past} and Y_{future}. For a calculation of these directions, see for instance [9], [11]. In [17] it is shown that the principal directions α in Y_{past}, form a basis for the state sequence x_k of the forward innovation model (9-10), in balanced realization. A realization A, G, C, Λ_0 is called balanced when both P and N are diagonal and $P = N$.

$$\alpha = \begin{pmatrix} \hat{x}_k & \hat{x}_{k+1} & \dots & \hat{x}_{k+j-1} \end{pmatrix}$$

The principal directions β in Y_{future}, form a basis for the state sequence z_k of the backward innovation model (12-13), in balanced realization :

$$\beta = \begin{pmatrix} \hat{z}_{k-1} & \hat{z}_k & \dots & \hat{z}_{k+j-2} \end{pmatrix}$$

Denote with :

- $Y = \begin{pmatrix} y_k & y_{k+1} & \dots & y_{k+j-2} \end{pmatrix}$
- $\underline{\alpha}$ all the columns of α, except the last one
- $\overline{\alpha}$ all the columns of α except the first one
- define $\underline{\beta}$ and $\overline{\beta}$ in the same way
- W_f, V_f, W_b, V_b matrices with columns resepectively equal to $Be_{f_s}, De_{f_s}, Ee_{b_s}$ and Fe_{b_s} for $s = k, \dots, k+j-2$

With these definitions and with (9-10) , we find the matrix equations :

$$\overline{\alpha} = \tilde{A}\underline{\alpha} + W_f \quad (16)$$
$$Y = \tilde{C}\underline{\alpha} + V_f \quad (17)$$

And with and (12-13), we find :

$$\underline{\beta} = \tilde{A}^t\overline{\beta} + W_b \quad (18)$$
$$Y = \tilde{G}^t\overline{\beta} + V_b \quad (19)$$

Since V_f, V_b contain white noise sequences, we can solve (17) and (19) using least squares as :

$$\tilde{C} = Y\underline{\alpha}^t(\underline{\alpha}\,\underline{\alpha}^t)^{-1} \quad (20)$$

$$\tilde{G} = (\overline{\beta}\,\overline{\beta}^t)^{-1}\overline{\beta}Y^t \quad (21)$$

Equations (16) and (18) are a little bit harder to solve simultaneously. A least squares problem can be defined as

$$\min_A \|\overline{\alpha} - A\underline{\alpha}\|_F^2 + \|\underline{\beta} - A^t\overline{\beta}\|_F^2 \quad (22)$$

Theorem 2
The solution to the least squares problem (22) is given by the solution to the following Sylvester equation :

$$A(\underline{\alpha}\,\underline{\alpha}^t) + (\overline{\beta}\,\overline{\beta}^t)A = \overline{\alpha}\underline{\alpha}^t + \overline{\beta}\underline{\beta}^t \quad (23)$$

Proof : Equation (22) can be rewritten as :

$$\min_A(tr((\overline{\alpha} - A\underline{\alpha})(\overline{\alpha} - A\underline{\alpha})^t) + tr((\underline{\beta} - A^t\overline{\beta})(\underline{\beta} - A^t\overline{\beta})^t))$$

$$= \min_A \sum_{k=1}^n \sum_{p=1}^{j-1} \begin{array}{l}[\overline{\alpha}_{kp}^2 - 2\sum_{m_1=1}^n \overline{\alpha}_{kp}A_{km_1}\underline{\alpha}_{m_1p} + (\sum_{m_1=1}^n A_{km_1}\underline{\alpha}_{m_1p})^2 + \\ \underline{\beta}_{kp}^2 - 2\sum_{m_2=1}^n \underline{\beta}_{kp}A_{m_2k}\overline{\beta}_{m_2p} + (\sum_{m_2=1}^n A_{m_2k}\overline{\beta}_{m_2p})^2]\end{array}$$

When we set the partial derivative with respect to a_{rs} equal to zero, we get :

$$\begin{array}{lll}
-2\sum_{p=1}^{j-1}\overline{\alpha}_{rp}\underline{\alpha}_{sp} + 2\sum_{p=1}^{j-1}\sum_{m_1=1}^n A_{rm_1}\underline{\alpha}_{m_1p}\underline{\alpha}_{sp} - 2\sum_{p=1}^{j-1}\underline{\beta}_{sp}\overline{\beta}_{rp} + 2\sum_{p=1}^{j-1}\sum_{m_2=1}^n A_{m_2s}\overline{\beta}_{m_2p}\overline{\beta}_{rp} & = & 0 \\
\sum_{p=1}^{j-1}[\sum_{m_1=1}^n A_{rm_1}\underline{\alpha}_{m_1p}\underline{\alpha}_{sp} - \overline{\alpha}_{rp}\underline{\alpha}_{sp} + \sum_{m_2=1}^n A_{m_2s}\overline{\beta}_{m_2p}\overline{\beta}_{rp} - \underline{\beta}_{sp}\overline{\beta}_{rp}] & = & 0 \\
\sum_{m_1=1}^n A_{rm_1}(\underline{\alpha}\,\underline{\alpha}^t)_{m_1s} - (\overline{\alpha}\,\underline{\alpha}^t)_{rs} + \sum_{m_2=1}^n A_{m_2s}(\overline{\beta}\,\overline{\beta}^t)_{m_2r} - (\overline{\beta}\,\underline{\beta}^t)_{rs} & = & 0 \\
(A\underline{\alpha}\,\underline{\alpha}^t - \overline{\alpha}\,\underline{\alpha}^t + \overline{\beta}\,\overline{\beta}^t A - \overline{\beta}\,\underline{\beta}^t)_{rs} & = & 0
\end{array}$$

Since this is true for all r and s, we find :

$$A(\underline{\alpha}\,\underline{\alpha}^t) + (\overline{\beta}\,\overline{\beta}^t)A = \overline{\alpha}\,\underline{\alpha}^t + \overline{\beta}\,\underline{\beta}^t$$

This completes the proof. □

A robust algorithm to solve this Sylvester equation is described in [10]. So, with formulas (20), (21) and (23), we can find estimates $\tilde{A}, \tilde{G}$ and $\tilde{C}$, without constructing the matrix $\tilde{H}_i$ of Markov parameters explicitly.

For the determination of the order of the system, we turn to the principal angles. The fact that $\tilde{H}_i$ is rank deficient (in the limit for $j \to \infty$), implies the existence of an $(li-n)$-dimensional subspace of the li-dimensional rowspace of Y_{past}, which is orthogonal to an $(li-n)$-dimensional subspace of the row space of Y_{future}. In both spaces there are n-dimensional subspaces that are not orthogonal to each other, but are "correlated". In other words, there are *n principal angles* (which is a generalization of the angle between two vectors, see e.g. [11] for a definition) between the row spaces of Y_{past} and Y_{future} that are not equal to $\pi/2$. All others are equal to $\pi/2$.
The total subspace algorithm will thus go as follows :

1. Form the block Toeplitz Y_{past} and block Hankel Y_{future} matrix out of the given data y_k. Form also the matrix $\tilde{\Lambda}_0 = \sum_{l=0}^{j-1} y_{k+l}y_{k+l}^t/j$.

2. Determine the principal angles between Y_{past} and Y_{future}. The number of angles different from $\pi/2$ determines the order n.

3. Determine the canonical directions between Y_{past} and Y_{future}. Use formulas (20),(21) and (23) to find matrices $\tilde{A}, \tilde{G}, \tilde{C}$.

The example in section 8 will clearly show the superior performance of this subspace algorithm compared to that of the classical realization algorithms.

6 About Positive Sequences and Positive Real Functions

In this section, we will show that a covariance sequence is only physically meaningful iff *it is positive real. So, it is highly necessary that the sequence $\tilde{\Lambda}_i$ formed out of the identified parameters $\tilde{A}, \tilde{G}, \tilde{C}$ and $\tilde{\Lambda}_0$ is positive real.*
Faurre proved the following lemma in [12]

Lemma 1 *Positive real lemma*

1. *The sequence $\tilde{\Lambda}_i \quad i = -\infty \ldots \infty$ is a covariance sequence* iff *it is positive real. A sequence, is positive real* iff *for all real sequences u_i,*

$$\begin{pmatrix} u_0 & u_1 & u_2 & \ldots \end{pmatrix} \begin{pmatrix} \tilde{\Lambda}_0 & \tilde{\Lambda}_1 & \tilde{\Lambda}_2 & \ldots \\ \tilde{\Lambda}_1^t & \tilde{\Lambda}_0 & \tilde{\Lambda}_1 & \ldots \\ \tilde{\Lambda}_2^t & \tilde{\Lambda}_1^t & \tilde{\Lambda}_0 & \ldots \\ \ldots & \ldots & \ldots & \ldots \end{pmatrix} \begin{pmatrix} u_0 \\ u_1 \\ u_2 \\ \ldots \end{pmatrix} > 0 \tag{24}$$

   In other words, the matrices $R_{past_i} = Y_{past} Y_{past}^t / j$ and $R_{future_i} = Y_{future} Y_{future}^t / j$, which can be expressed in function of $\tilde{A}, \tilde{G}, \tilde{C}, \tilde{\Lambda}_0$ in the following way :

$$R_{future_i} = \begin{pmatrix} \tilde{\Lambda}_0 & \tilde{C}\tilde{G} & \tilde{C}\tilde{A}\tilde{G} & \ldots & \tilde{C}\tilde{A}^{i-2}\tilde{G} \\ \tilde{G}^t\tilde{C}^t & \tilde{\Lambda}_0 & \tilde{C}\tilde{G} & \ldots & \tilde{C}\tilde{A}^{i-3}\tilde{G} \\ \ldots & \ldots & \ldots & \ldots & \ldots \\ \tilde{G}^t(\tilde{A}^t)^{i-2}\tilde{C}^t & \tilde{G}^t(\tilde{A}^t)^{i-3}\tilde{C}^t & \ldots & \tilde{G}^t\tilde{C}^t & \tilde{\Lambda}_0 \end{pmatrix} \tag{25}$$

   have to be positive definite for all i (because of the eigenvalue interlacing theorem).

2. *The sequence $\tilde{\Lambda}_i$ is a covariance sequence* iff *the set of Markovian realizations of the covariance sequence is not empty. Since this set is bounded by P and N^{-1}, this can be reformulated as follows :*

3. *The sequence $\tilde{\Lambda}_i$ is a covariance sequence* iff *the forward (11) and backward (14) Riccati equations have a positive definite solution for the identified matrices $\tilde{A}, \tilde{G}, \tilde{C}, \tilde{\Lambda}_0$.*

So, it is clear that, if the Riccati equations have no positive definite solution, the set of possible Markovian realizations is empty, and we are not able to identify a forward or backward innovations model, let alone any other model ! In other words, it is highly necessary to identify a covariance sequence (determined by $\tilde{A}, \tilde{G}, \tilde{C}, \tilde{\Lambda}_0$), that leads to a *positive real* sequence !

Now, if one starts from real raw data, or even from data that was generated by simulation of a linear stochastic system, it turns out that this condition is hardly ever satisfied. The

identified $\tilde{A}, \tilde{G}, \tilde{C}, \tilde{\Lambda}_0$ do not result in a positive real sequence $\tilde{\Lambda}_i$. This observation is fairly independent of the covariance identification algorithm used.

This practical problem is rarely addressed in the literature. Aoki ([2],p124) mentions it once for a rank 1 SISO system, and Vaccaro [20] addresses the problem for a general SISO system, where he scales a Riccati equation until it has a positive definite solution. The basic problem is thus to perturb the estimated covariance sequence, until it becomes positive real or in other words, until the set of Markovian representations is not empty any more.

7 Approximation Algorithm

In this section, we'll present an algorithm that enables us to adjust the identified covariance sequence until it is positive real. This is achieved by perturbing the lag zero covariance $\tilde{\Lambda}_0$. The heuristically inspired algorithm is guaranteed to return a positive real sequence in all cases and is based on an algorithm that calculates the L_∞ norm of a transfer matrix presented by Boyd [5].

7.1 Adaption of the Estimated $\tilde{\Lambda}_0$

Another way than equation (24-25) to guarantee the positive realness of the covariance sequence $\tilde{\Lambda}_i$ is the following :

Theorem 3 *[1], [6]*
The covariance sequence $\tilde{\Lambda}_i$ is a positive real sequence iff *the function*

$$T(z) = \tilde{\Lambda}_0 + \tilde{C}(zI - \tilde{A})^{-1}\tilde{G} + \tilde{G}^t(z^*I - \tilde{A}^t)^{-1}\tilde{C}^t \tag{26}$$

is a positive real function. This means that it has to be stable (no poles inside the unit circle) and that the real part has to be positive definite on the unit circle. $T(z)$ is the powerspectrum, or in other words : the z-transform of the covariance sequence $\tilde{\Lambda}_i$.

So far, we have seen two different practical ways to check the positive realness of the sequence $\tilde{\Lambda}_i$:

- Solve the Riccati equations (11-14). If there is no positive definite solution, then the sequence $\tilde{\Lambda}_i$ is not possitive real.
- Determine z on the unit circle, for which the real part of $T(z)$ has the smallest eigenvalue. Call this point z_{min} and the minimum ϵ_{min}. If ϵ_{min} is positive, then $T(z)$ is a positive real function.

We can use theorem 3 in the following way : a covariance identification procedure (realization or subspace) will return the estimated matrices : $\tilde{A}, \tilde{G}, \tilde{C}, \tilde{\Lambda}_0$. If these matrices form a positive sequence $\tilde{\Lambda}_i$, then there is no problem (and this will be the case for $j \to \infty$, see further). But, if the sequence is not positive, then $\tilde{\Lambda}_0$ can be corrected in the following way :

- Determine ϵ_{min}. If ϵ_{min} is positive, then nothing needs to be done because this means that $T(z)$ already is a positive real function. On the other hand, when ϵ_{min} is negative, calculate $T_{min} = \Re(T(z_{min}))$.
- Calculate the positive definite matrix $\tilde{T}_{min}$ that is closest to T_{min} in Frobenius norm in the following way [13] :

– obtain the polar decomposition of T_{min} via its SVD :

$$T_{min} = USV^t = (UV^t)(VSV^t) = U_\theta R$$

– The closest positive definite matrix $\hat{T}_{min}$ is then given by

$$\hat{T}_{min} = (T_{min} + R)/2$$

- substitute :

$$\tilde{\Lambda}_0 \leftarrow \tilde{\Lambda}_0 + (\hat{T}_{min} - T_{min})$$

It is possible that this procedure has to be repeated, if there is more than one cluster of negative eigenvalues. Every step, the direction of Λ_0 corresponding to the eigenvector of T_{min} associated with ϵ_{min} is adapted. This method :

1. ensures that the final sequence is positive real.
2. is unbiased for $j \to \infty$ since at $j = \infty$, ϵ_{min} will be positive. This is because the estimates of the covariances obey the following formulas :

$$\text{mean :} \quad \mathbf{E}(\tfrac{1}{p}\textstyle\sum_{k=1}^{p} y_{k+i}y_k^t) \quad = \quad \Lambda_i$$

$$\text{covariance :} \quad \mathbf{E}((\tfrac{1}{p}\textstyle\sum_{k=1}^{p} y_{k+i}y_k^t - \Lambda_i)(\tfrac{1}{n}\textstyle\sum_{k=1}^{p} y_{k+i}y_k^t - \Lambda_i)^t) \quad \sim \quad \tfrac{1}{p}$$

This last equation shows that the variance on the estimates of Λ_i will go to zero, so to estimate of Λ_i goes to Λ_i with probability 1. So, the estimated covariance sequence $\tilde{\Lambda}_i$ will converge to its exact value, and so will the powerspectrum $T(z)$.

Why do we adjust Λ_0 ? First of all, because it is the easiest way to find a positive real covariance sequence that is close to the identified one. But also because the influences of the estimation errors on the identified sequence $\tilde{\Lambda}_i : i = 1, \ldots,$ have already been reduced. These matrices are the result of fitting of linear system through their estimates. On the other hand, there has been no correction whatsoever on the error on the estimate of $\tilde{\Lambda}_0$ since it is not used in the covariance identification procedure.

7.2 How to Calculate the Minimum Eigenvalue of $\Re(T(z))$

In this section, we'll present an efficient algorithm to calculate the minimum eigenvalue ϵ_{min}. The algorithm is inspired on [5] where a similar idea is applied for an efficient algorithm to compute the L_∞ norm of a transfer matrix. We'll illustrate the algorithm and all potential problems with an example.

7.2.1 The Level Set

The level set S_γ corresponding to the real value γ is defined as the set of points z on the unit circle, for which at least one of the eigenvalues of the real part of $T(z)$ is equal to γ. So, the problem is to find all z on the unit circle, for which :

$$eig(\Re(\tilde{\Lambda}_0 + \tilde{C}(zI - \tilde{A})^{-1}\tilde{G} + \tilde{G}^t(z^*I - \tilde{A}^t)^{-1}\tilde{C}^t)) = \gamma \tag{27}$$

7.2.2 The Level Set Solved as a Generalized Eigenvalue Problem

We can determine S_γ as follows : first we'll try to get rid of the $\Re$ operator in equation (27). With $z = x + iy$ and $x^2 + y^2 = 1$ (because z is a point on the unit circle), we can rewrite $T(z)$ as :

$$\tilde{\Lambda}_0 + \tilde{C}(I - 2x\tilde{A} + \tilde{A}^2)^{-1}(xI - \tilde{A} - iy)\tilde{G} + \tilde{G}^t(I - 2x\tilde{A}^t + (\tilde{A}^t)^2)^{-1}(xI - \tilde{A}^t + iy)\tilde{C}^t \quad (28)$$

If we take the real part of formula (28) and combine it with formula (27), we get the following (p is an eigenvector associated with the eigenvalue γ) :

$$[\tilde{\Lambda}_0 + \tilde{C}(I - 2x\tilde{A} + \tilde{A}^2)^{-1}(xI - \tilde{A})\tilde{G} + \tilde{G}^t(I - 2x\tilde{A}^t + (\tilde{A}^t)^2)^{-1}(xI - \tilde{A}^t)\tilde{C}^t]p = p\gamma \quad (29)$$

Observe there is no y in this formula any more. Now, we define the vectors u and v as follows :

$$\begin{aligned} u &= (I - 2x\tilde{A} + \tilde{A}^2)^{-1}(xI - \tilde{A})\tilde{G}p & (30)\\ v &= (I - 2x\tilde{A}^t + (\tilde{A}^t)^2)^{-1}(xI - \tilde{A}^t)\tilde{C}^t p & (31) \end{aligned}$$

This can be rewritten as :

$$\begin{aligned} (xI - \tilde{A})\tilde{G}p &= (I - 2x\tilde{A} + \tilde{A}^2)u & (32)\\ (xI - \tilde{A}^t)\tilde{C}^t p &= (I - 2x\tilde{A}^t + (\tilde{A}^t)^2)v & (33) \end{aligned}$$

If we combine formulas (29-31), we get the following :

$$\begin{aligned} \tilde{\Lambda}_0 p + \tilde{C}u + \tilde{G}^t v &= p\gamma \\ (\gamma I - \tilde{\Lambda}_0)^{-1}\tilde{C}u + (\gamma I - \tilde{\Lambda}_0)^{-1}\tilde{G}^t v &= p & (34) \end{aligned}$$

Now, we eliminate p through a combination of formulas (32), (33) and (34). This leads to the following generalized eigenvalue problem (with $L_\gamma = (\gamma I - \tilde{\Lambda}_0)^{-1}$) :

$$\begin{pmatrix} \tilde{A}\tilde{G}L_\gamma\tilde{C} + I + \tilde{A}^2 & \tilde{A}\tilde{G}L_\gamma\tilde{G}^t \\ \tilde{A}^t\tilde{C}^t L_\gamma\tilde{C} & \tilde{A}^t\tilde{C}^t L_\gamma\tilde{G}^t + I + (\tilde{A}^t)^2 \end{pmatrix} \begin{pmatrix} u \\ v \end{pmatrix} = \begin{pmatrix} \tilde{G}L_\gamma\tilde{C} + 2\tilde{A} & \tilde{G}L_\gamma\tilde{G}^t \\ \tilde{C}^t L_\gamma\tilde{C} & \tilde{C}^t L_\gamma\tilde{G}^t + 2\tilde{A}^t \end{pmatrix} \begin{pmatrix} u \\ v \end{pmatrix} x \quad (35)$$

If we retain only the solutions x for which x is real and $|x| < 1$, we find all the points $z = x + iy$ (with $y = \sqrt{1 - x^2}$) for which at least one of the eigenvalues of the real part of $T(z)$ is equal to γ. We'll presume there are m_γ of these points z (m_γ depends on γ). This set forms the level set S_γ.

7.2.3 Bisection on the Level Set and Practical Implications

Call R_γ the ordened set of solutions x (the real part of the elements in S_γ), extended with -1 on the left on $+1$ on the right :

$$R_\gamma = \begin{pmatrix} -1 & x_1 & \dots & x_{m_\gamma} & 1 \end{pmatrix}$$

The remaining part of the algorithm bisects the intervals formed by two succesive points in R_γ : take the middle of all intervals $[R_{\gamma_i}, R_{\gamma_{i+1}}]$, $i = 1, \dots, m_\gamma + 1$, (with R_{γ_i} denoting the

i^{th} element in R_γ and form the corresponding point z_i on the unit circle. Then, for every z_i, find the minimum eigenvalue of the real part of $T(z_i)$. Retain the minimum of these $m_\gamma + 1$ minima, and use this as γ in formula (35) to determine the new intersections.

What to do when there is no intersection (the set S_γ is empty). There are two cases where this could happen :

1. During the first iteration. If we take γ_0 equal to zero, and if $T(z)$ is already positive real, then there will be no intersection. R_γ will then be equal to the interval $(-1\ ,\ 1)$, and the bisection algorithm will return a positive γ. So, in this case, the error is corrected automatically by the algorithm.

2. When the minimum is pretty flat, it is possible that, even though there should be an intersection (since γ is equal to the minimum eigenvalue of $\Re(T(z_{prev}))$ of the previous step), due to numerical errors, there is no intersection any more. This hapens when γ is already close to the minimum. In this case the algorithm terminates and ϵ_{min} is set equal to the minimum of $\min(eig(\Re(T(z_{prev}))))$ and $\min(eig(\Re(T(z_{prev+1}))))$ with (z_{prev}, z_{prev+1}) the interval in which the last γ was found in.

In summary, the algorithm will proceed as follows :

1. Put $\gamma_{new} = 0$.

2. Solve the generalized eigenvalue problem of formula (35) with $\gamma = \gamma_{new}$. Retain of the solutions x only those that are real and have $|x| < 1$. If there is no solution :

   - If it is the first step : just go on.
   - if it is not the first step : $\epsilon_{min} = \gamma_{new}$. Stop.

   Sort these x's and construct the set R_γ (add -1 and 1).

3. For every two succesive points in R_γ, calculate the middle z_i and evaluate $T(z_i)$. Calculate for every $T(z_i)$ the minimum eigenvalue.

4. Put $\gamma_{old} = \gamma_{new}$. Of all the $m_\gamma + 1$ minimum eigenvalues of step 3, put γ_{new} equal to the minimum one. Call $\delta = |\gamma_{new} - \gamma_{old}|$.

   - If $\delta < \epsilon$, $\epsilon_{min} = \gamma_{new}$. Stop.
   - If $\delta > \epsilon$, goto 2.

   with ϵ the required accuracy on ϵ_{min}.

7.2.4 An Example

Consider the following linear stochastic system ($n = 4, l = 2$) :

$$A = \begin{pmatrix} 0.3253 & -0.3964 & -0.0557 & 0.5482 \\ 4.9394 & -2.8852 & 4.2957 & 2.7416 \\ 3.0678 & -2.3636 & 3.9068 & 2.4470 \\ 0.3289 & 0.6619 & -0.6860 & -0.7470 \end{pmatrix} \quad B = \begin{pmatrix} -0.0335 & 0.0240 \\ 0.0080 & 0.0125 \\ -0.0048 & -0.0224 \\ -0.0615 & -0.0391 \end{pmatrix}$$

$$C = \begin{pmatrix} -1.3820 & 1.2695 & -0.9564 & -1.2538 \\ 1.0763 & 0.4873 & 0.4518 & 0.2564 \end{pmatrix} \quad D = \begin{pmatrix} 0.0421 & 0.1209 \\ 0.1042 & 0.0781 \end{pmatrix}$$

step	γ_{new}	Δ	δ
1	0	0.09722318706026	1.00000000000000
2	-0.09645100937362	0.00077217768664	0.00794232024259
3	-0.09722163275609	0.00000155430417	0.00001598696992
4	-0.09722318705606	0.00000000000420	0.00000000004316
5	-0.09722318706026	0	0

Table 1: For the four steps : γ_{new}, the absolute error Δ and the relative error δ. $\epsilon = 10^{-12}$ in this example.

with eigenvalues : $-0.5 \pm 0.8i$ and $0.8 \pm 0.3i$. Figure 2 shows the minimum eigenvalue of $T(z)$ on the unit circle in function of x. Clearly $T(z)$ is not positive real, since for some x, the minimum eigenvalue is smaller than 0. Figure 2 also shows the results of the algorithm of the previous section applied to this example. The consequitive γ's, the absolute errors $\Delta = \gamma_{new} - \epsilon_{min}$ and relative errors $\delta = \Delta/\epsilon_{min}$ are given in Table 1. The errors go down quadratically, so this suggests that this algorithm (just as the original one for the calculation of the L_∞ norm [5]), is quadratically convergent. This still has to be proven.

8 Example

In this section, we'll illustrate the presented subspace algorithm with an example. We'll also illustrate the approximation algorithm.
We simulated the following stochastic system : $(n = 3, l = 2)$

$$A = \begin{pmatrix} 1.7 & -1.2 & 0.35 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \end{pmatrix} \qquad B = \begin{pmatrix} 1 & 0 \\ 0 & 1 \\ 0 & 0.5 \end{pmatrix}$$

$$C = \begin{pmatrix} 3.7 & -2.2 & -0.15 \\ 2.3 & 0 & -1.5 \end{pmatrix} \qquad D = I_2$$

with eigenvalues : $0.5 \pm 0.5i$ and 0.7

8.1 The Subspace Algorithm

Using the 'rand('normal')' function of MATLAB, we simulated 100 different output sequences of 600 points y_k for this system. To each sequence, we applied 4 different covariance identification algorithms with $i = 10$ and $j = 500$:

1. The Principal Component (P.C.) algorithm [3]
2. The Canonical Correlations (C.C.) algorithm [3]
3. The Scaled Covariance algorithm for finite data sets [19]
4. The Subspace algorithm of this paper

For each algorithm, we used the same data. Figure 3 shows the clusters of estimated poles for the four algorithms. Table 2 lists the sample mean of the estimated poles of the covariance sequence and the sample standard deviation. From this figure and table, it is clear that :

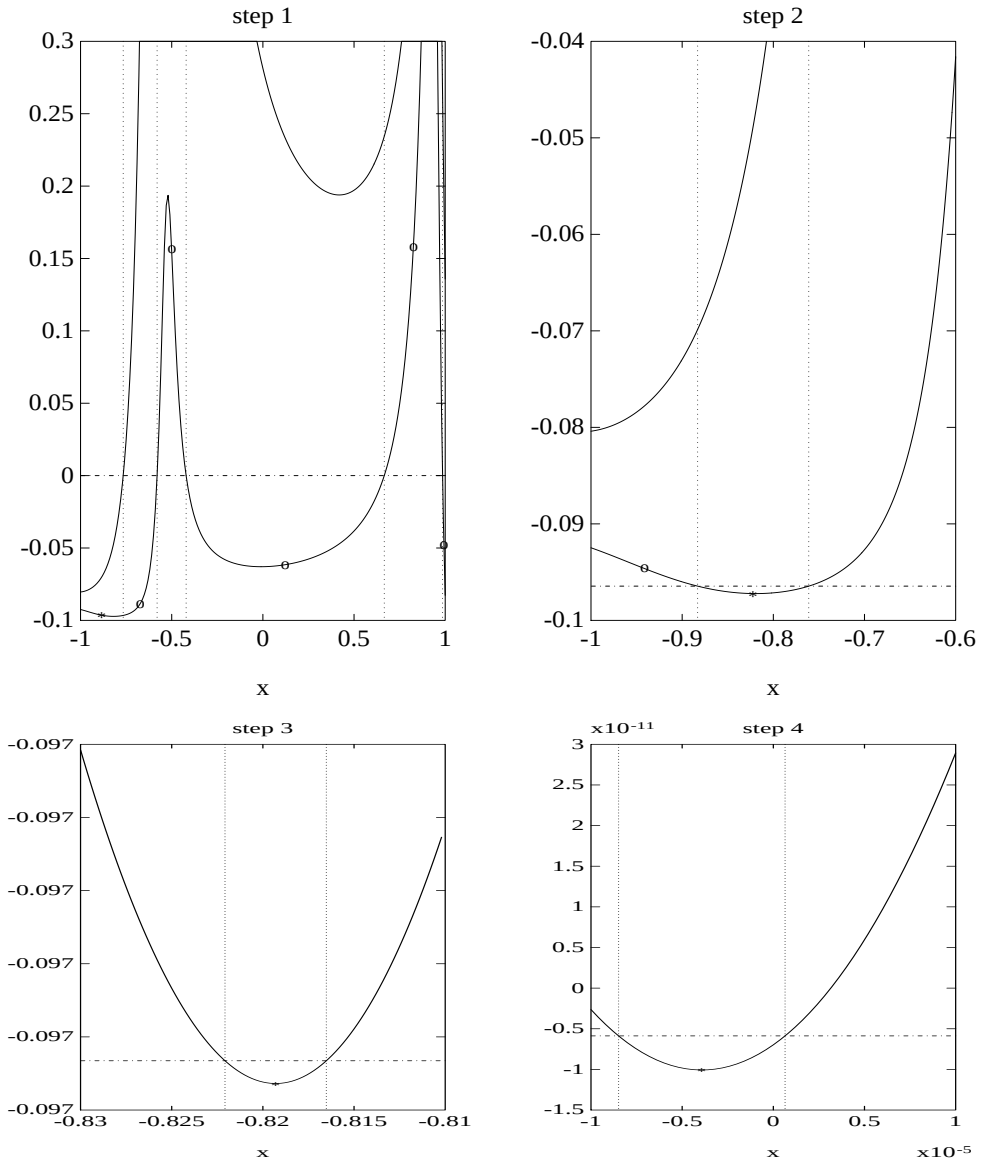


Figure 2: Four steps of the algorithm. The solid lines indicate the minimum (and maximum) eigenvalue of $T(z)$ in function of x. The dashed horizontal line is the value of γ during that step. The dotted vertical lines are the intersections found by the algorithm (R_γ). The dots are midpoints of these intervals and the star is the minimum of these midpoints. The resulting minimum is $\epsilon_{min} = -0.097223187060$. Remark that in step 4, the plot is shifted : on the x-axis : $x + 0.81929$ and on the y-axis : $y + 0.097223187050187$ is plotted (because the graphical program only has a big reolution around 0).

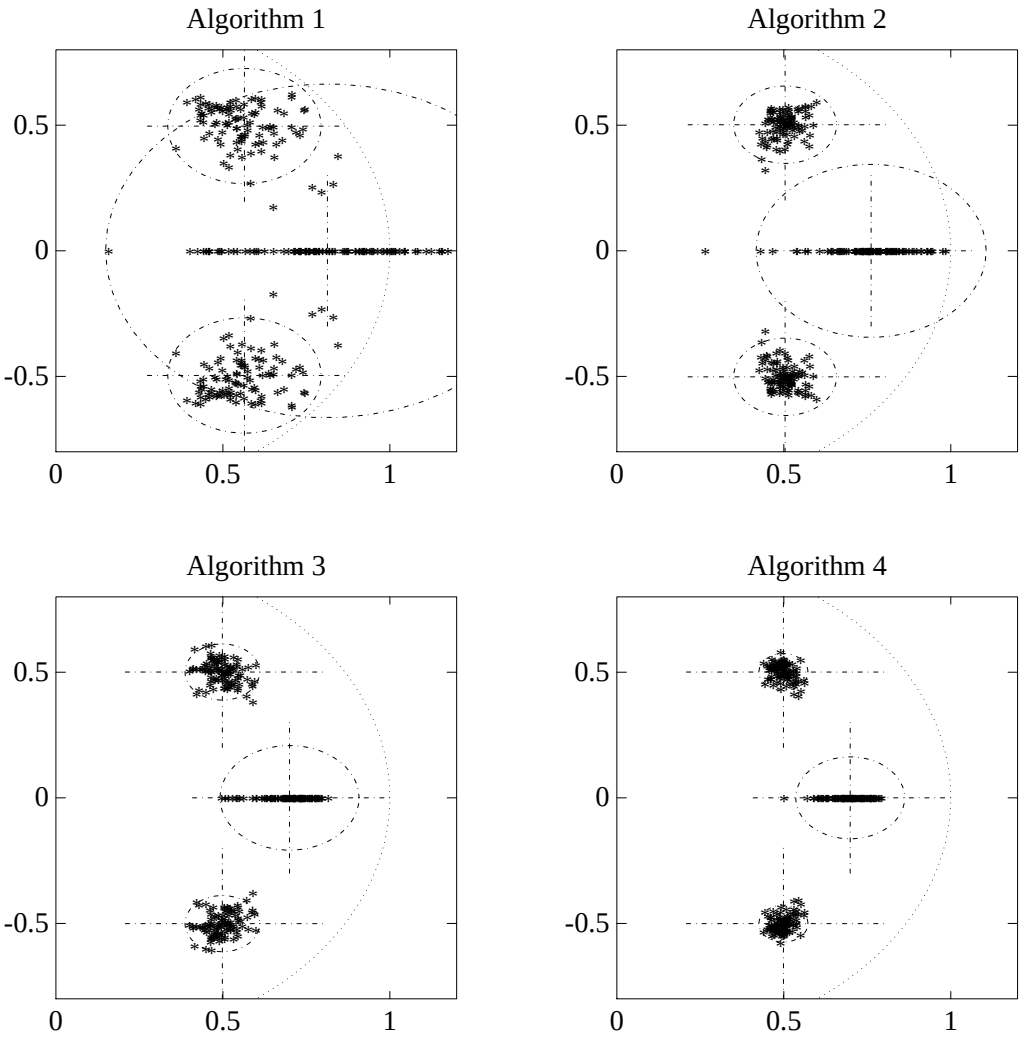


Figure 3: Estimated poles for the four algorithms for 100 different runs : Algorithm 1 = Principal Components, Algorithm 2 = Canonical Correlations, Algorithm 3 = Scaled Covariances, Algorithm 4 = Subspace. The crosses are the sample mean, the circles are drawn at a distance of 3 times the sample standard deviation. The dotted line is the unit circle. Observe how the circles grow smaller with each new algorithm.

alg.	$\|r\|$		$\|c\|$		$\measuredangle c$	
	mean	std	mean	std	mean	std
1	.8130	.2215	.7596	.0766	.7237	.1460
2	.7614	.1148	.7121	.0514	.7820	.0551
3	.6995	.0694	.7083	.0373	.7876	.0718
4	.6988	.0544	.7076	.0245	.7869	.0492

Table 2: Sample mean and standard deviations of the absolute value and angle of the estimated poles (r = real pole : 0.7 and c =complex pole : $0.5 \pm 0.5i = 0.7071 \measuredangle 0.7854$) of the covariance sequence.

j	50	150	250	350	450	550	650	750	850	950	1050
1	2.8	16.0	15.1	18.3	14.8	16.3	27.9	32.1	31.0	18.9	22.2
2	37.2	13.5	24.1	14.2	16.4	11.9	9.7	3.6	4.1	5.1	2.5
3	16.7	0.13	0.17	0.62	0.34	0.40	0.52	0.29	0.09	0.15	0.17
4	8.4	0	3.9	0	0	0.54	0	0	1.7	0.11	0.22

Table 3: Procentual corrections on the estimated $\tilde{\Lambda}_0$: c_p. j indicates the number of measurements used for identification ($i = 10$).

- The bias on the estimate of the poles is the smallest for the subspace algorithm (for both the real and the complex pole).
- The standard deviation of the estimates of the poles is also the smallest for the subspace algorithm. In comparison with the P.C. algorithm the difference is as much as a factor of 4.

So clearly, the estimation of the modes of the stochastic system is a lot more accurate with the subspace technique than with any other technique. The only algorithm that comes close, is the *finite data* algorithm of Vaccaro [19].

8.2 Positive Real Approximations

Identification of the covariance sequence is not enough. The identified sequence also has to be positive real ! With $\tilde{\Lambda}_{0s}$ the corrected $\tilde{\Lambda}_0$ (with the algorithm of section 7.1), we define the procentual correction on $\tilde{\Lambda}_0$ as :

$$c_p = 100 \, \frac{\|\tilde{\Lambda}_{0s} - \tilde{\Lambda}_0\|_F}{\|\tilde{\Lambda}_{0s}\|_F} \%$$

This is one possible measure for how close to being positive real the identified covariance sequences are.

From Table 3, it is clear that for the first two algorithms (P.C. and C.C.), the estimated covariance sequences are pretty far from a positive real sequence. On the other hand, the algorithm of Vaccaro and the subspace algorithm of this paper return results that are within a very small correction from being positive real. Even though the corrections are sometimes a little bit smaller for the algorithm of Vaccaro, a clear advantage of the subspace algorithm is

that quite often the estimated covariance sequence is already positive definite, which means that there is no correction necessary. One can also see that for the last two algorithms, the corrections become smaller as j grows bigger. The covariances $\tilde{\Lambda}_i$ are better estimated when more measurements are used. The only disadvantage seems to be that by modifying $\tilde{\Lambda}_0$, we modify the power of the stochastic process, which is given by $tr(\tilde{\Lambda}_0)$ and also the principal angles, which depend on $\tilde{\Lambda}_0$ through the Riccati equations. An ideal positive real approximation algorithm would try to obtain a stochastic model $\tilde{A}, \tilde{G}, \tilde{C}, \tilde{\Lambda}_0$ such that the observed principal angles are preserved. This is still an open problem.

9 Conclusions

A new subspace algorithm to identify the covariance sequence of a stochastic system has been presented. This approach avoids the construction of the Hankel matrix with Markov parameters. Instead, it uses the principal angles and directions. A Sylvester equation has to be solved to find an unbiased solution. Moreover, the sample standard deviation on the poles is clearly less than for all other stochastic realization algorithms. Also a new algorithm to approximate the identified covariance sequence with a positive real sequence was presented. This algorithm looks for the minimum eigenvalue of the power spectrum on the unit circle. By making use of this approximation algorithm, the procentual correction to the class of positive sequences was defined. This correction once again turned out to be the smallest for the subspace algorithm. The combination of the two presented algorithms leads to a robust stochastic identification algorithm.

References

[1] Akhiezer N.I. *The Classical Moment problem.* Oliver and Boyd, London 1965.

[2] Aoki M. *State Space Modelling of Time Series.* Springer Verlag Berlin Heidelberg, 1987.

[3] Arun K.S., Kung S.Y. *Balanced Aproximations of Stochastic Systems.* SIAM Matrix Analysis and Applications, Vol. 11, No 1, pp 42-68, January 1990.

[4] Barmish B.R. *Robust Solution of Perturbed Dynamical Equations from Within a Convex Restraint Set.* IEEE Transactions on Automatic Control, Vol. AC-24, no.6, December 1979.

[5] Boyd S., Balakrishnan V. *A Regularity Result for the Singular Values of a Transfer Matrix and a Quadratically Convergent Algorithm for Computing its L_∞-norm.* Systems Control Letters, Volume 15, pp 1-7, 1990.

[6] Caines P.E. *Linear Stochastic Systems.* John Wiley and Sons, 1988.

[7] De Moor B. *Mathematical concepts and techniques for modelling of static and dynamic systems.* Doctoral Dissertation, Department of Electrical Engineering, Katholieke Universiteit Leuven, Belgium, June 1988.

[8] De Moor B., Van Overschee P., Suykens J. *Subspace Algorithms for System Identification and Stochastic realization.* Submitted for publication.

[9] Desai U.B., Pal D. *A transformation approach to stochastic model reduction.* IEEE Transactions on Automatic Control, Vol. AC-29, no.12, December 1984.

[10] Golub G., Nash S., Van Loan C. *A Hessenberg-Schur method for the problem* $AX+XB = C$. IEEE Transactions on Automatic Control, Vol. AC-24, pp 909-913, 1979.

[11] Golub G., Van Loan C. *Matrix Computations.* North Oxford Academic, 1983.

[12] Faurre P. *Stochastic realization algorithms.* System identification: Advances and case studies, Eds. Mehra R. and Lainiotis D., Academic Press, 1976.

[13] Higham N.J. *Computing a Nearest Symmetric Positive Semidefinite Matrix.* Linear Algebra and Its Applications, Vol. 103, pp 103-118, May 1988.

[14] Ho B.L., Kalman R.E. *Effective construction of linear state variable models for input output functions.* Regelungstechnik 14, pp 545-548, 1966.

[15] Kung S.Y. *A new Identification and Model Reduction Algorithm via Singular Value Decomposition.* Proceedings of the 12th Asilomar Conference on Circuits, Systems and Computers, Pacific Grove, pp 705-714, 1978.

[16] Ljung L., Kailath T., *Backward Markovian Models for Second-Order Stochastic Processes.* IEEE Transactions on Information Theory, IT-22, pp 483-491, 1976.

[17] Pal D. *Balanced Stochastic Realization and Model Reduction.* Master's thesis, Whasington State University, Electrical Engineering, 1982.

[18] Roy R., Kailath T. *ESPRIT - Estimation of signal parameters via rotational invariance techniques.* IEEE Transactions on Acoustics, Speech and Signal Processing, 37, 7, pp.984-995, July 1989.

[19] Vaccaro R.J. *Finite-Data Algorithms for Approximate Stochastic Realization.* In *Modelling and Applications of Stochastic Processes*, U.B. Desai, ed., Kluwer Academic Publishers, pp. 105-122, 1986.

[20] Vaccaro R.J. *A State Space Approach to Positive Sequences* proceedings of ICASSP, pp 1304-1307, 1987.