Subspace system identification of the Kalman filter: open and closed loop systems

Dr. David Di Ruscio
Telemark university college
Department of process automation
N-3914 Porsgrunn, Norway
Fax: +47 35 57 52 50
Tel: +47 35 57 51 68

February 22, 2007

Abstract

Some proofs concerning a subspace identification algorithm are presented. It is proved that the Kalman filter gain and the noise innovations process can be identified directly from known input and output data without explicitly solving the Riccati equation. Furthermore, it is in general and for colored inputs, proved that the subspace identification of the states only is possible if the deterministic part of the system is known or identified beforehand. However, if the inputs are white, then, it is proved that the states can be identified directly. Some alternative projection matrices which can be used to compute the extended observability matrix directly from the data are presented. Furthermore, an efficient method for computing the deterministic part of the system is presented. The closed loop subspace identification problem is also addressed and it is shown that this problem is solved and unbiased estimates are obtained by simply including a filter in the feedback. Furthermore, an algorithm for consistent closed loop subspace estimation is presented.

Keywords: Identification methods; Subspace methods; Stochastic systems; Sampled data systems; Linear systems.

1 Introduction

A complete subspace identification (SID) algorithm are discussed and derived in this paper. The derivation presented is different from the other published papers on subspace identification, Van Overschee and De Moor (1994), Larimore (1990), Viberg (1995) and Van Overschee (1995) and the references therein, because we are using general input and output matrix equations which describes the relationship between the past and the future input and output data matrices.

One of the contributions in this paper is that it is shown that the Kalman filter model matrices, including the Kalman gain and the noise innovations process, of a combined deterministic and stochastic system can be identified directly from certain projection matrices which are computed from the known input and output data, without solving any Riccati or Lyapunov matrix equations. This subspace method and results was presented without proof in Di Ruscio (1995) and Di Ruscio (1997). One contribution in this paper is a complete derivation with proof. A new method for computing the matrices in the deterministic part of the system is presented. This method has been used in the DSR Toolbox for Matlab, Di Ruscio (1996), but has not been published earlier.

Furthermore, it is pointed out that the states, in general (i.e. for colored input signals), only can be computed if the complete deterministic part of the model is known or identified first. This is probably the reason for which the state based subspace algorithms which are presented in the literature does not work properly for colored input signals. The SID algorithm in Verhagen (1994) works for colored input signals. The stochastic part of the model is not computed by this algorithm. The N4SID algorithm in Van Overschee and De Moor (1994) works well and only for white input signals. The stochastic part of the model is computed by solving a Riccati equation. However, the robust modification in Van Overschee and De Moor (1995) works well also for colored input signals.

The rest of this paper is organized as follows. Some basic matrix definitions and notations are presented in Section 2. The problem of subspace identification of the states for both colored and white input signals is discussed in Section 3.1. The subspace identification of the extended observability matrix, which possibly is the most important step in any SID algorithm, are discussed in Section 3.2. It is proved that the Kalman filter gain matrix and the noise innovations process can be identified directly from the data in Section 3.3. A least squares optimal method for computing the deterministic part of the combined deterministic and stochastic system is presented in Section 3.4.

The problem of using subspace methods for closed loop systems are pointed out and some solutions to the problem are pointed out in section 4.

The main contribution in this paper is a new method for subspace system identification that works for closed loop as well as open loop systems. The method are based on the theory in Section 3 and is presented in Section 5. This method is probably one of the best for closed loop subspace system identification.

Some topics and remarks related to the algorithm are presented in Section 6. Numerical examples are provided in Section 7 in order to illustrate the behaviour of the algorithm both in open and closed loop. Some concluding remarks follows in Section 8.

2 Notation and definitions

2.1 System and matrix definitions

Consider the following state space model on innovations form

$$\bar{x}_{k+1} = A\bar{x}_k + Bu_k + Ce_k, \tag{1}$$

$$y_k = D\bar{x}_k + Eu_k + Fe_k, \tag{2}$$

where e_k is white noise with covariance matrix $\mathrm{E}(e_k e_k^T) = I_m$. One of the problems addressed and discussed in this paper is to directly identify (subspace identification) the system order, n, the state vector $\bar{x}_k \in \mathbb{R}^n$, and the matrices (A, B, C, D, E, F) from a sequence of known input and output data vectors, u_k, $\in \mathbb{R}^r$ and y_k, $\in \mathbb{R}^m$, respectively. A structure parameter, g, is introduced so that $g = 1$ when E is to be identified and $g = 0$ when E is a-priori known to be zero. This should be extended to a structure matrix G with ones and zeroes, the ones pointing to the elements in E which are to be estimated. This is not considered further here. Based on (1) and (2) we make the following definitions for further use:

Definition 2.1 (Basic matrix definitions)
The extended observability matrix, O_i, for the pair (D, A) is defined as

$$O_i \stackrel{\text{def}}{=} \begin{bmatrix} D \\ DA \\ \vdots \\ DA^{i-1} \end{bmatrix} \in \mathbb{R}^{im \times n}, \tag{3}$$

where the subscript i denotes the number of block rows.
The reversed extended controllability matrix, C_i^d, for the pair (A, B) is defined as

$$C_i^d \stackrel{\text{def}}{=} \begin{bmatrix} A^{i-1}B & A^{i-2}B & \cdots & B \end{bmatrix} \in \mathbb{R}^{n \times ir}, \tag{4}$$

where the subscript i denotes the number of block columns. A reversed extended controllability matrix, C_i^s, for the pair (A, C) is defined similar to (4), i.e.,

$$C_i^s \stackrel{\text{def}}{=} \begin{bmatrix} A^{i-1}C & A^{i-2}C & \cdots & C \end{bmatrix} \in \mathbb{R}^{n \times im}, \tag{5}$$

i.e., with B substituted with C in (4). The lower block triangular Toeplitz matrix, H_i^d, for the quadruple matrices (D, A, B, E)

$$H_i^d \stackrel{\text{def}}{=} \begin{bmatrix} E & 0_{m\times r} & 0_{m\times r} & \cdots & 0_{m\times r} \\ DB & E & 0_{m\times r} & \cdots & 0_{m\times r} \\ DAB & DB & E & \cdots & 0_{m\times r} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ DA^{i-2}B & DA^{i-3}B & DA^{i-4}B & \cdots & E \end{bmatrix} \in \mathbb{R}^{im \times (i+g-1)r}, \tag{6}$$

where the subscript i denotes the number of block rows and $i + g - 1$ is the number of block columns. Where $0_{m\times r}$ denotes the $m \times r$ matrix with zeroes. A lower block triangular Toeplitz matrix H_i^s for the quadruple (D, A, C, F) is defined as

$$H_i^s \stackrel{\text{def}}{=} \begin{bmatrix} F & 0_{m\times m} & 0_{m\times m} & \cdots & 0_{m\times m} \\ DC & F & 0_{m\times m} & \cdots & 0_{m\times m} \\ DAC & DC & F & \cdots & 0_{m\times m} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ DA^{i-2}C & DA^{i-3}C & DA^{i-4}C & \cdots & F \end{bmatrix} \in \mathbb{R}^{im \times im}. \tag{7}$$

2.2 Hankel matrix notation

Hankel matrices are frequently used in realization theory and subspace system identification. The special structure of a Hankel matrix as well as some matching notations, which are frequently used througout, are defined in the following.

Definition 2.2 (Hankel matrix) *Given a (vector or matrix) sequence of data*

$$s_t \;\in\; \mathbb{R}^{nr\times ns} \;\forall\; t = 0,1,2,\dots,t_0,t_0+1,\dots, \tag{8}$$

where nr is the number of rows in s_t and nc is the number of columns in s_t. Define integer numbers t_0, L and K and define the matrix S_t as follows

$$S_{t_0|L} \overset{def}{=} \begin{bmatrix} s_{t_0} & s_{t_0+1} & s_{t_0+2} & \cdots & s_{t_0+K-1} \\ s_{t_0+1} & s_{t_0+2} & s_{t_0+3} & \cdots & s_{t_0+K} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ s_{t_0+L-1} & s_{t_0+L} & s_{t_0+L+1} & \cdots & s_{t_0+L+K-2} \end{bmatrix} \in \mathbb{R}^{Lnr\times Knc}. \tag{9}$$

which is defined as a Hankel matrix because of the special structure. The integer numbers t_0, L and K are defined as follows:

- *t_0 start index or initial time in the sequence, s_{t_0}, which is the upper left block in the Hankel matrix.*
- *L is the number of nr-block rows in $S_{t_0|L}$.*
- *K is the number of nc-block columns in $S_{t_0|L}$.*

A Hankel matrix is symmetric and the elements are constant across the anti-diagonals. We are usually working with vector sequences in subspace system identification, i.e., s_t is a vector in this case and hence, $nc = 1$. Examples of such vector processes, to be used in the above Hankel-matrix definition, are the measured process outputs, $y_t \in \mathbb{R}^m$, and possibly known inputs, $u_t \in \mathbb{R}^r$. Also define

$$y_{j|i} \overset{\text{def}}{=} \begin{bmatrix} y_j^T & y_{j+1}^T & \cdots & y_{j+i-1}^T \end{bmatrix}^T \;\in\; \mathbb{R}^{im}, \tag{10}$$

which is refereed to as an extended (output) vector, for later use.

2.3 Projections

Given two matrices $A \in \mathbb{R}^{i\times k}$ and $B \in \mathbb{R}^{j\times k}$. The orthogonal projection of the row space of A onto the row space of B is defined as

$$A/B = AB^T(BB^T)^\dagger B. \tag{11}$$

The orthogonal projection of the row space of A onto the orthogonal complement of the row space of B is defined as

$$AB^\perp = A - A/B = A - AB^T(BB^T)^\dagger B. \tag{12}$$

The following properties are frequently used

$$A / \begin{bmatrix} A \\ B \end{bmatrix} = A, \tag{13}$$

$$A / \begin{bmatrix} A \\ B \end{bmatrix}^{\perp} = 0. \tag{14}$$

Prof of (13) and (14) can be found in e.g., Di Ruscio (1997b). The Moore-Penrose pseudo-inverse of a matrix $A \in \mathbb{R}^{i \times k}$ where $k > i$ is defined as $A^{\dagger} = A^T(AA^T)^{-1}$. Furthermore, consistent with (12) we will use the definition

$$B^{\perp} = I_k - B^T(BB^T)^{\dagger}B, \tag{15}$$

throughout the paper. Note also the properties that $(B^{\perp})^T = B^{\perp}$ and $B^{\perp}B^{\perp} = B^{\perp}$.

3 Subspace system identification

3.1 Subspace identification of the states

Consider a discrete time Kalman filter on innovations form, i.e.,

$$\bar{x}_{k+1} = A\bar{x}_k + Bu_k + K\varepsilon_k, \tag{16}$$

$$y_k = D\bar{x}_k + Eu_k + \varepsilon_k, \tag{17}$$

where $\bar{x}_k \in \mathbb{R}^n$ is the predicted state in a minimum variance sense, $\varepsilon_k \in \mathbb{R}^m$ is the innovations at discrete time k, i.e., the part of $y_k \in \mathbb{R}^m$ that cannot be predicted from past data (i.e. known past inputs and outputs) and the present input. Furthermore, $\bar{y}_k = D\bar{x}_k + Eu_k$ is the prediction of y_k, and ε_k is white noise with covariance matrix $\Delta = \mathrm{E}(\varepsilon_k\varepsilon_k^T)$. Here $\varepsilon_k = Fe_k$ is the innovations and the model (1) and (2) is therefore equivalent with the Kalman filter (16) and (17). Furthermore, we have that $K = CF^{-1}$ and $\Delta = \mathrm{E}(\varepsilon_k\varepsilon_k^T) = FF^T$, when F is non-singular, i.e., when the system is not deterministic and when the Kalman filter exists.

A well known belief is that the states is a function of the past. Let us have a lock at this statement. The predicted state at time $k := t_0 + J$, i.e. $\bar{x}_{t_0+J}$ of a Kalman filter with the initial predicted state at $k := t_0$, i.e. $\bar{x}_{t_0}$ given, can be expressed as

$$\bar{x}_{t_0+J} = \tilde{C}_J^s y_{t_0|J} + \tilde{C}_J^d u_{t_0|J} + (A - KD)^J \bar{x}_{t_0}, \tag{18}$$

where $\tilde{C}_J^s = C_J(A - KD, K)$ is the reversed extended controllability matrix of the pair $(A - KD, K)$, $\tilde{C}_J^d = C_J(A - KD, B - KE)$ is the reversed extended controllability matrix of the pair $(A - KD, B - KE)$ and $\bar{x}_{t_0}$ is the initial predicted state (estimate) at the initial discrete time t_0. See (5) for the definition of the reversed controllability matrix. J is the *past horizon*, i.e., the number of

past outputs and inputs used to define the predicted state (estimate) $\bar{x}_{t_0+J}$ at the discrete time $t_0 + J$.
Using (18) for different t_0, i.e. for t_0, $t_0 + 1$, $t_0 + 2$, …, $t_0 + K - 1$, gives the matrix equation

$$X_{t_0+J} = \tilde{C}_J^s Y_{t_0|J} + \tilde{C}_J^d U_{t_0|J} + (A - KD)^J X_{t_0}, \tag{19}$$

where

$$X_{t_0+J} = \begin{bmatrix} \bar{x}_{t_0+J} & \bar{x}_{t_0+J+1} & \cdots & \bar{x}_{t_0+J+K-1} \end{bmatrix} \in \mathbb{R}^{n\times K}, \tag{20}$$
$$X_{t_0} = \begin{bmatrix} \bar{x}_{t_0} & \bar{x}_{t_0+1} & \cdots & \bar{x}_{t_0+K-1} \end{bmatrix} \in \mathbb{R}^{n\times K}. \tag{21}$$

where K is the number of columns in the data matrices. Note that K also is equal to the number of vector equations of the form (18) which is used to form the matrix version (19). Note also that the state matrix X_{t_0} can be eliminated from (19) by using the relationship

$$Y_{t_0|J} = O_J X_{t_0} + H_J^d U_{t_0|J+g-1} + H_J^s E_{t_0|J}, \tag{22}$$

which we have deduced from the innovations form, state space model (1) and (2). Putting $t_0 =: t_0 + J$ in (22) gives

$$Y_{t_0+J|L} = O_L X_{t_0+J} + H_L^d U_{t_0+J|L+g-1} + H_L^s E_{t_0+J|L}. \tag{23}$$

Using (19) to eliminate X_{t_0+J} in (23) gives a matrix equation which relates the future data matrices $Y_{t_0+J|L}$, $U_{t_0+J|L+g-1}$, $E_{t_0+J|L}$ and the past data matrices $Y_{t_0|J}$, $U_{t_0|J}$, $E_{t_0|J}$.
The data is usually defined at time instant (or number of observations) $k = 1, 2, \ldots, N$. Hence, $t_0 = 1$ in this case. However, we are often defining $t_0 = 0$ which corresponds to data defined at $k = 0, 1, \ldots, N - 1$. The bar used to indicate predicted state is often omitted. Hence, for simplicity of notation, we define the following equations from (19), (22) and (23),

$$Y_{0|J} = O_J X_0 + H_J^d U_{0|J+g-1} + H_J^s E_{0|J}, \tag{24}$$

$$X_J = \tilde{C}_J^s Y_{0|J} + \tilde{C}_J^d U_{0|J} + (A - KD)^J X_0, \tag{25}$$

$$Y_{J|L} = O_L X_J + H_L^d U_{J|L+g-1} + H_L^s E_{J|L}, \tag{26}$$

for lather use. Furthermore, (26) and (25) gives

$$Y_{J|L} = \begin{bmatrix} H_L^d & O_L\tilde{C}_J^d & O_L\tilde{C}_J^s \end{bmatrix} \begin{bmatrix} U_{J|L+g-1} \\ U_{0|J} \\ Y_{0|J} \end{bmatrix} + O_L(A - KD)^J X_0 + H_L^s E_{J|L}. \tag{27}$$

Equation (27) is important for understanding a SID algorithm, because, it gives the relationship between the past and the future. Note also the terms in (27) which are "proportional" with the extended observability matrix O_L. From (27) we see that the effect from the future inputs, $U_{J|L+g-1}$, and the future noise, $E_{J|L}$, have to be removed from the future outputs, $Y_{J|L}$, in order to recover the subspace spanned by the extended observability matrix, O_L. A variation of this equation, in which the term X_0 is eliminated by using (22) or (24) is presented in Di Ruscio (1997b). Note also that (25) and (24) gives

$$X_J = \begin{bmatrix} P_J^u & P_J^y \end{bmatrix} \begin{bmatrix} U_{0|J} \\ Y_{0|J} \end{bmatrix} - P_J^e E_{0|J}, \tag{28}$$

$$\begin{aligned} P_J^u &= \tilde{C}_J^d - (A-KD)^J O_J^\dagger H_J^d, & (29)\\ P_J^y &= \tilde{C}_J^s + (A-KD)^J O_J^\dagger, & (30)\\ P_J^e &= (A-KD)^J O_J^\dagger H_J^s, & (31)\end{aligned}$$

where we for the sake of simplicity and without loss of generality have put $g=1$. Equation (28) is useful because it shows that the future states X_J is in the range of a matrix consisting of past inputs, $U_{0|J}$, and past outputs, $Y_{0|J}$ (in the deterministic case or when $J \to \infty$). Note that we have introduced the notation, P_J^u, in order to represent the influence from the *past* inputs upon the future. Combining (28) and (26) gives an alternative to (27), i.e. the "past-future" matrix equation,

$$Y_{J|L} = \begin{bmatrix} H_L^d & O_L P_J^u & O_L P_J^y \end{bmatrix} \begin{bmatrix} U_{J|L+g-1} \\ U_{0|J} \\ Y_{0|J} \end{bmatrix} - O_L P_J^e E_{0|J} + H_L^s E_{J|L}. \tag{32}$$

The two last terms in (32) cannot be predicted from data, i.e., because $E_{0|J}$ and $E_{J|L}$ are built from the innovations process e_k.
It is important to note that a consistent estimate of the system dynamics can be obtained by choosing L and N properly. Choosing $L_{\text{min}} \leq L$ where $L_{\text{min}} = n + \text{rank}(D) - 1$ and letting $N \to \infty$, is in general, necessary conditions for a consistent estimate of the dynamics. See Section 3.2 for further details.
On the other side, it is in general, also necessary to let $J \to \infty$ in order to obtain a consistent estimate of the states. The reason for this is that the term $(A-KD)^J = 0$ in this case. Hence, the effect of the initial state matrix X_0 on the future states X_J has died out. We have the following Lemma

Lemma 3.1 (Subspace identification of the states)
Let $K \to \infty$ in the data matrices. The projected state matrix is defined as

$$\begin{aligned} X_J / \begin{bmatrix} U_{J|L+g-1} \\ U_{0|J} \\ Y_{0|J} \end{bmatrix} &= O_L^\dagger (\overbrace{Y_{J|L} / \begin{bmatrix} U_{J|L+g-1} \\ U_{0|J} \\ Y_{0|J} \end{bmatrix}}^{Z_{J|L}^d} - H_L^d U_{J|L+g-1}) \\ &= \tilde{C}_J^s Y_{0|J} + \tilde{C}_J^d U_{0|J} + (A-KD)^J X_0 / \begin{bmatrix} U_{J|L+g-1} \\ U_{0|J} \\ Y_{0|J} \end{bmatrix}. \end{aligned} \tag{33}$$

Consider the case when

$$(A-KD)^J X_0 / \begin{bmatrix} U_{J|L+g-1} \\ U_{0|J} \\ Y_{0|J} \end{bmatrix} = 0, \tag{34}$$

which is satisfied when $J \to \infty$ and $(A-KD)$ is stable. This gives

$$X_J / \begin{bmatrix} U_{J|L+g-1} \\ U_{0|J} \\ Y_{0|J} \end{bmatrix} = X_J, \tag{35}$$

and hence we have, in general, the following expression for the future states

$$X_J = O_L^{\dagger}(\overbrace{Y_{J|L} / \begin{bmatrix} U_{J|L+g-1} \\ U_{0|J} \\ Y_{0|J} \end{bmatrix}}^{Z^d_{J|L}} - H^d_L U_{J|L+g-1}). \tag{36}$$

△

Proof 3.1 *The proof is divided into two parts.*

Part 1

The relationship between the future data matrices is given by

$$Y_{J|L} = O_L X_J + H^d_L U_{J|L+g-1} + H^s_L E_{J|L}. \tag{37}$$

Projecting the row space of each term in (37) onto the row space of $\begin{bmatrix} U_{J|L+g-1} \\ U_{0|J} \\ Y_{0|J} \end{bmatrix}$ *gives*

$$Y_{J|L} / \begin{bmatrix} U_{J|L+g-1} \\ U_{0|J} \\ Y_{0|J} \end{bmatrix} = O_L X_J / \begin{bmatrix} U_{J|L+g-1} \\ U_{0|J} \\ Y_{0|J} \end{bmatrix} + H^d_L U_{J|L+g-1} + dE_1 \tag{38}$$

where the error term is given by

$$dE_1 = H^s_L E_{J|L} / \begin{bmatrix} U_{J|L+g-1} \\ U_{0|J} \\ Y_{0|J} \end{bmatrix}. \tag{39}$$

It make sense to assume that future noise matrix $E_{J|L}$ is uncorrelated with past data and the future inputs, hence, we have that (w.p.1)

$$\lim_{K\to\infty} dE_1 = 0. \tag{40}$$

Part 2

Equation (25) gives the relationship between the future state matrix X_J and the past data matrices. Projecting the row space of each term in this equation onto the row space of $\begin{bmatrix} U_{J|L+g-1} \\ U_{0|J} \\ Y_{0|J} \end{bmatrix}$ *gives*

$$X_J / \begin{bmatrix} U_{J|L+g-1} \\ U_{0|J} \\ Y_{0|J} \end{bmatrix} = \tilde{C}_J^s Y_{0|J} + \tilde{C}_J^d U_{0|J} + (A - KD)^J X_0 / \begin{bmatrix} U_{J|L+g-1} \\ U_{0|J} \\ Y_{0|J} \end{bmatrix}. \quad (41)$$

Letting $J \to \infty$ (or assuming the last term to be zero) gives

$$X_J / \begin{bmatrix} U_{J|L+g-1} \\ U_{0|J} \\ Y_{0|J} \end{bmatrix} = \tilde{C}_J^s Y_{0|J} + \tilde{C}_J^d U_{0|J}. \quad (42)$$

Letting $J \to \infty$ and assuming the system matrix $(A - KD)$ for the predicted outputs to be stable in (25) shows that

$$X_J = \tilde{C}_J^s Y_{0|J} + \tilde{C}_J^d U_{0|J}. \quad (43)$$

Comparing (42) and (43) gives

$$X_J = X_J / \begin{bmatrix} U_{J|L+g-1} \\ U_{0|J} \\ Y_{0|J} \end{bmatrix}. \quad (44)$$

Using (44) in (38) and solving for X_J gives (36). □

The condition in (35) is usually satisfied for large J, i.e., we have that $\lim_{J\to\infty}(A - KD)^J = 0$ when $A - KD$ is stable. Note also that the eigenvalues of $A - KD$ usually are close to zero for "large" process noise (or "small" measurements noise). Then, $(A - KD)^J$ is approximately zero even for relatively small numbers J. We will now discuss some special cases

Lemma 3.2 (SID of states: white input)
Consider a combined deterministic and stochastic system excited with a white input signal. Then

$$X_J = O_L^{\dagger} Y_{J|L} / \begin{bmatrix} U_{J|L+g-1} \\ U_{0|J} \\ Y_{0|J} \end{bmatrix} U_{J|L+g-1}^{\perp} \quad (45)$$

when $J \to \infty$.

Proof 3.2 *This result follows from the proof of Lemma 3.1 and (36) and using that*

$$X_J U_{J|L+g-1}^{\perp} = X_J \quad (46)$$

when u_k is white and, hence, $X_0 / U_{J|L+g-1} = 0$. □

Lemma 3.3 (SID of states: pure stochastic system)
Consider a stochastic system. Then we simply have that

$$X_J = O_L^{\dagger} Y_{J|L}/Y_{0|J} \tag{47}$$

when $J \rightarrow \infty$ or when $(A - KD)^J X_0/Y_{0|J} = 0$ is satisfied.

Proof 3.3 *This result follows from the proof of Lemma 3.1 by putting the measured input variables equal to zero.* □

Lemma 3.1 shows that it is in general (i.e. for colored input signals) necessary to know the deterministic part of the system, i.e., the Toepliz matrix H_L^d in (36), in order to properly identify the states. This means that the matrices B and E in addition to D and A has to be identified prior to computing the states. I.e. we need to know the deterministic part of the model. However, a special case is given by Lemma 3.2 and Equation (45) which shows that the states can be identified directly when the input signals is white. Note also that the extended observability matrix O_L is needed in (36) and (45). O_L can be identified directly from the data. This is proved in the next Section 3.2, and this is indeed the natural step in a SID algorithm.
In the case of a white input signal or when $J \rightarrow \infty$ then, H_L^d, and the state matrix, X_J, can be computed as by the N4SID algorithm, Van Overschee and De Moor (1996). From (32) and (28) we have the following lemma

Lemma 3.4 (States, X_J, and Toepliz matrix H_L^d: N4SID)
The following LS solution

$$\left[\begin{array}{ccc} H_L^d & O_L P_J^u & O_L P_J^y \end{array} \right] = Y_{J|L} \left[\begin{array}{c} U_{J|L+g-1} \\ U_{0|J} \\ Y_{0|J} \end{array} \right]^{\dagger} + dE. \tag{48}$$

holds in:

i) The deterministic case, provided the input is PE of order $J+L+g-1$. The error term, $dE = 0$, in this case.

ii) When $J \rightarrow \infty$, and the input is PE of infinite order. The error term, $dE = 0$, in this case.

iii) A white u_k gives a consistent estimate of H_L^d irrespective of $J > 0$. However, $O_L P_J^u$ and $O_L P_J^y$ are not consistent estimates in this case. The first $mL \times (L+g)r$ part of the error term, dE, is zero in this case.

Hence, under conditions i) and ii), $O_L P_J^u$ and $O_L P_J^y$ can be computed as in (48). Then the states can be consistently estimated as

$$X_J = O_L^{\dagger} \left[\begin{array}{cc} O_L P_J^u & O_L P_J^y \end{array} \right] \left[\begin{array}{c} U_{0|J} \\ Y_{0|J} \end{array} \right], \tag{49}$$

provided conditions i) and ii) are satisfied, and $O_L^{\dagger}$ is known.

Proof 3.4 *The PE conditions in the lemma are due to the existence of the LS solution, i.e., the concatenated matrix* $\begin{bmatrix} U_{J|L+g-1} \\ U_{0|J} \end{bmatrix}$ *has to be of full row rank. From (32) we have that the error term in the LS problem is*

$$dE = (-O_L P_J^e E_{0|J} + H_L^s E_{J|L}) \begin{bmatrix} U_{J|L+g-1} \\ U_{0|J} \\ Y_{0|J} \end{bmatrix}^{\dagger} = -O_L P_J^e E_{0|J} \begin{bmatrix} U_{J|L+g-1} \\ U_{0|J} \\ Y_{0|J} \end{bmatrix}^{\dagger} . \quad (50)$$

It is clear from (31) that the error term $dE = 0$ *when* $J \to \infty$. *This proves condition i) in the lemma. Furthermore, the error term,* $dE = 0$, *in the deterministic case because* $E_{0|J} = 0$ *in this case. This proves condition ii). Analyzing the error term,* dE, *for a white input shows that the error term is of the form*

$$dE = \begin{bmatrix} 0_{mL \times (L+g)r} & dE_2 & dE_3 \end{bmatrix}^{\dagger}, \quad (51)$$

where the dE_2 *and* dE_3 *are submatrices in* dE *different from zero. Note that* $dE_2 = 0$ *for strictly proper systems,* $g = 0$, *when* u_k *is white. This proves condition iii).*
The states can then be computed by using (28) or (43), provided conditions i) or ii) are satisfied. □

One should note that in the N4SID algorithm the past horizon is put equal to the future horizon (N4SID parameter i). In order for the above lemma to give the same results as in the N4SID algorithm we have to put $i = L+1$, $J = L+1$ and $g = 1$, i.e so that $J + L = 2L + 1 = 2i$. Note that this last result does not hold in general. It holds in the deterministic case or when $J \to \infty$. The extended observability matrix O_L can be computed as presented in the next section.

3.2 The extended observability matrix

An important first step in the SID algorithm is the identification of the system order, n, and the extended observability matrix O_{L+1}. The reason for searching for O_{L+1} is that we have to define A from the shift invariance property, Kung (1978), or a similar method, e.g. as in Di Ruscio (1995). The key is to compute a special projection matrix from the known data. This is done without using the states. We will in this section show how this can be done for colored input signals.

Lemma 3.5 (SID of the extended observability matrix)
The following projections are equivalent

$$Z_{J|L+1} = (Y_{J|L+1} / \begin{bmatrix} U_{J|L+g} \\ U_{0|J} \\ Y_{0|J} \end{bmatrix}) U_{J|L+g}^{\perp} \quad (52)$$

$$Z_{J|L+1} = (Y_{J|L+1} U_{J|L+g}^{\perp}) / (\begin{bmatrix} U_{0|J} \\ Y_{0|J} \end{bmatrix} U_{J|L+g}^{\perp}) \quad (53)$$

$$Z_{J|L+1} = Y_{J|L+1}/(\begin{bmatrix} U_{0|J} \\ Y_{0|J} \end{bmatrix} U^{\perp}_{J|L+g}) \tag{54}$$

Furthermore, $Z_{J|L+1}$ is related to the extended observability matrix O_{L+1} as

$$Z_{J|L+1} = O_{L+1} X^a_J, \tag{55}$$

where the "projected states" X^a_J can be expressed as

$$\begin{aligned} X^a_J &= (X_J / \begin{bmatrix} U_{J|L+g} \\ U_{0|J} \\ Y_{0|J} \end{bmatrix}) U^{\perp}_{J|L+g} & (56) \\ &= (\tilde{C}^d_J U_{0|J} + \tilde{C}^s_J Y_{0|J} - (A-KD)^J X_0 / \begin{bmatrix} U_{J|L+g} \\ U_{0|J} \\ Y_{0|J} \end{bmatrix}) U^{\perp}_{J|L+g} & (57) \\ &= (X_J - (A-KD)^J X_0 \begin{bmatrix} U_{J|L+g} \\ U_{0|J} \\ Y_{0|J} \end{bmatrix}^{\perp}) U^{\perp}_{J|L+g} & (58) \\ &= (X_J + (A-KD)^J O^{\dagger}_J H^s_J E_{0|J} \begin{bmatrix} U_{J|L+g} \\ U_{0|J} \\ Y_{0|J} \end{bmatrix}^{\perp}) U^{\perp}_{J|L+g} & (59) \end{aligned}$$

Furthermore, the column space of $Z_{J|L+1}$ coincide with the column space of O_{L+1} and $n = rank(Z_{J|L+1})$ if $rank(X^a_J) = n$.

Proof 3.5 *The proof is divided into two parts. In the first part (52) and (55) with the alternative expressions in (56) to (58) are proved. In the second part the equivalence with (52), (53) and (54) are proved.*

Part 1 *Projecting the row space of each term in (26) with $L := L+1$ onto the row space of $\begin{bmatrix} U_{J|L+g-1} \\ U_{0|J} \\ Y_{0|J} \end{bmatrix}$ gives*

$$Y_{J|L+1} / \begin{bmatrix} U_{J|L+g-1} \\ U_{0|J} \\ Y_{0|J} \end{bmatrix} = O_{L+1} X_J / \begin{bmatrix} U_{J|L+g-1} \\ U_{0|J} \\ Y_{0|J} \end{bmatrix} + H^d_{L+1} U_{J|L+g-1} + dE_1, \tag{60}$$

where we have used (13). Then, w.p.1

$$\lim_{K \to \infty} dE_1 = 0, \tag{61}$$

where the error term, dE_1, is given by (39) with $L := L+1$. Removing the effect of the future input matrix, $U_{J|L+g-1}$, on (60) gives (52) and (55) with X^a_J as in (56).

Furthermore, projecting the row space of each term in (25) onto the row space of $\begin{bmatrix} U_{J|L+g-1} \\ U_{0|J} \\ Y_{0|J} \end{bmatrix}$ gives

$$X_J / \begin{bmatrix} U_{J|L+g-1} \\ U_{0|J} \\ Y_{0|J} \end{bmatrix} = \tilde{C}^s_J Y_{0|J} + \tilde{C}^d_J U_{0|J} + (A-KD)^J X_0 / \begin{bmatrix} U_{J|L+g-1} \\ U_{0|J} \\ Y_{0|J} \end{bmatrix} \tag{62}$$

From (25) we have that

$$\tilde{C}_J^s Y_{0|J} + \tilde{C}_J^d U_{0|J} = X_J - (A - KD)^J X_0. \tag{63}$$

Combining (60), (62) and (63) gives (52) and (57)-(58).
Part 2 *It is proved in Di Ruscio (1997) that*

$$\begin{aligned} Z_{J|L+1} &= Y_{J|L+1} / \begin{bmatrix} U_{J|L+g} \\ W \end{bmatrix} U_{J|L+g}^{\perp} \\ &= Y_{J|L+1} U_{J|L+g}^{\perp} W^T (W U_{J|L+g}^{\perp} W^T)^{-1} W U_{J|L+g}^{\perp}, \end{aligned} \tag{64}$$

where

$$W = \begin{bmatrix} U_{0|J} \\ Y_{0|J} \end{bmatrix}. \tag{65}$$

Using that $U_{J|L+g}^{\perp} U_{J|L+g}^{\perp} = U_{J|L+g}^{\perp}$ in (64) proves the equivalence between (53), (54) and (52). □

Lemma 3.6 (Consistency: Stochastic and deterministic systems)
Let $J \to \infty$, then

$$Z_{J|L+1} = O_{L+1} X_J U_{J|L+g}^{\perp}, \tag{66}$$

where $Z_{J|L+1}$ is defined as in Lemma 3.5. A sufficient condition for consistency, and that O_{L+1} is contained in the column space of $Z_{J|L+1}$, is that there are no pure state feedback.

Proof 3.6 *Letting $J \to \infty$ in (58) gives (66). This can also be proved by using (44) in (56). Furthermore, if there are pure state feedback then $X_J U_{J|L+g}^{\perp}$ will lose rank below the normal rank which is n.* □

Lemma 3.7 (Deterministic systems)
For pure deterministic systems we have that (66) can be changed to

$$Z_{J|L+1} =: Y_{J|L+1} U_{J|L+g}^{\perp} = O_{L+1} X_J U_{J|L+g}^{\perp}. \tag{67}$$

The extended observability matrix O_{L+1} can be computed from the column space of $Y_{J|L+1} U_{J|L+g}^{\perp}$. Furthermore, one can let $J = 0$ in the deterministic case.

Proof 3.7 *This follows from (66) and Lemma 3.5 by excluding the projection which removes the noise.* □

Lemma 3.8 (Stochastic systems)
For pure stochastic systems we have that (66) can be changed to

$$Z_{J|L+1} =: Y_{J|L+1} / Y_{0|J} = O_{L+1} X_J. \tag{68}$$

The extended observability matrix O_{L+1} can be computed from the column space of $Y_{J|L+1} / Y_{0|J}$.

Proof 3.8 *This follows from (66) and Lemma 3.5 by excluding the input matrices from the equations and definitions.* □

3.3 Identification of the stochastic subsystem

We will in this section prove that, when the extended observability matrix is known (from Section 3.2), the kalman filter gain matrix can be identified directly from the data. Furthermore, it is proved that the noise innovations process can be identified directly in a first step in the DSR subspace algorithm. This result was first presented in Di Ruscio (1995) without proof. Some results concerning this is also presented in Di Ruscio (2001) and (2003).

Lemma 3.9 (The innovations)
Define the following projection from the data

$$Z^s_{J|L+1} = Y_{J|L+1} - Y_{J|L+1}/\begin{bmatrix} U_{J|L+g} \\ U_{0|J} \\ Y_{0|J} \end{bmatrix} = Y_{J|L+1}\begin{bmatrix} U_{J|L+g} \\ U_{0|J} \\ Y_{0|J} \end{bmatrix}^{\perp}. \tag{69}$$

Then w.p.1 as $J \to \infty$

$$Z^s_{J|L+1} = H^s_{L+1}E_{J|L+1}. \tag{70}$$

Hence, the Toeplitz matrix H^s_{L+1} (with Markov matrices F, DC, …, $DA^{L-1}C$) for the stochastic subsystem is in the column space of $\frac{1}{\sqrt{K}}Z^s_{J|L+1}$ since $\frac{1}{K}E_{J|L+1}E^T_{J|L+1} = I_{L+1\times L+1}$.

Proof 3.9 *The relationship between the future data matrices is given by*

$$Y_{J|L} = O_L X_J + H^d_L U_{J|L+g-1} + H^s_L E_{J|L}. \tag{71}$$

Projecting the row space of each term in (71) onto the row space of $\begin{bmatrix} U_{J|L+g-1} \\ U_{0|J} \\ Y_{0|J} \end{bmatrix}$ *gives*

$$Y_{J|L}/\begin{bmatrix} U_{J|L+g-1} \\ U_{0|J} \\ Y_{0|J} \end{bmatrix} = O_L X_J/\begin{bmatrix} U_{J|L+g-1} \\ U_{0|J} \\ Y_{0|J} \end{bmatrix} + H^d_L U_{J|L+g-1} + dE_1, \tag{72}$$

then, w.p.1

$$\lim_{K\to\infty} dE_1 = 0, \tag{73}$$

where dE_1 is given in (39). Furthermore,

$$\lim_{J\to\infty} X_J/\begin{bmatrix} U_{J|L+g-1} \\ U_{0|J} \\ Y_{0|J} \end{bmatrix} = X_J, \tag{74}$$

where we have used Equations (44) and (38). From (71), (72) and (74) we have that

$$Y_{J|L} - Y_{J|L}/\begin{bmatrix} U_{J|L+g-1} \\ U_{0|J} \\ Y_{0|J} \end{bmatrix} = H^s_L E_{J|L}. \tag{75}$$

Putting $L := L+1$ in (75) gives (69). □

Note the following from Lemma 3.9. The innovations can be identified directly as for $g = 1$

$$Z^s_{J|1} = FE_{J|1} = Y_{J|1} - Y_{J|1}/\begin{bmatrix} U_{J|1} \\ U_{0|J} \\ Y_{0|J} \end{bmatrix} \tag{76}$$

or for $g = 0$ when $E = 0$

$$Z^s_{J|1} = FE_{J|1} = Y_{J|1} - Y_{J|1}/\begin{bmatrix} U_{0|J} \\ Y_{0|J} \end{bmatrix} \tag{77}$$

One should note that (77) holds for both open and closed loop systems. For closed loop systems it make sense to only consider systems in which the direct feed-through matrix, E, from the input u_k to the output y_k is zero. This result will be used in order to construct a subspace algorithm whic gives consisten results for close loop systems, se Section 5.

It is now possible to directly identify the matrices C and F in the innovations model (1) and (2) and K and Δ in the Kalman filter (16) and (17). Two methods are presented in the following. The first one is a direct covariance based method for computing K and Δ and the second one is a more numerically reliable "square root" based method for computing C and F.

Lemma 3.10 (correlation method for K and Δ) *Define the projection matrix $Z^s_{J|L+1}$ as in (69) and define the correlation matrix*

$$\Delta_{L+1} = \frac{1}{K} Z^s_{J|L+1}(Z^s_{J|L+1})^T = H^s_{L+1}(H^s_{L+1})^T. \tag{78}$$

where the Toepliz matrix H^s_{L+1} can be partitioned as

$$H^s_{L+1} = \begin{bmatrix} F & 0_{m\times Lm} \\ O_L C & H^s_L \end{bmatrix}, \tag{79}$$

where $C = KF$. Hence, (78) can be written as

$$\Delta_{L+1} = \begin{bmatrix} \Delta_{11} & \Delta_{12} \\ \Delta_{21} & \Delta_{22} \end{bmatrix} = \begin{bmatrix} FF^T & F(O_L C)^T \\ O_L C F^T & O_L C(O_L C)^T + H^s_L(H^s_L)^T \end{bmatrix}. \tag{80}$$

From this we have

$$E(\varepsilon_k \varepsilon_k^T) = FF^T = \Delta_{11} \tag{81}$$

and

$$K = CF^{-1} = O_L^\dagger \Delta_{21}\Delta_{11}^{-1}. \tag{82}$$

Lemma 3.11 (square-root method for C and F) *The LQ decomposition of $\frac{1}{\sqrt{K}} Z^s_{J|L+1}$ gives*

$$\frac{1}{\sqrt{K}} Z^s_{J|L+1} = R_{33}Q_3. \tag{83}$$

Then, the Toeplitz matrix H^s_{L+1}, and the Markov matrices F, DC, ..., $DA^{L-1}C$, are given directly by

$$H^s_{L+1} = R_{33}. \tag{84}$$

F can be taken as one of the diagonal $m \times m$ block matrices in R_{33}, e.g. the lower left sub-matrix, i.e.

$$F = R_{33}(Lm+1 : (L+1)m, Lm+1 : (L+1)m), \tag{85}$$

or as the mean of all the diagonals. Furthermore,

$$O_L C = R_{33}(m+1 : (L+1)m, 1 : m). \tag{86}$$

The system matrix C is given by

$$C = O_L^\dagger O_L C = O_L^\dagger R_{33}(m+1 : (L+1)m, 1 : m). \tag{87}$$

The Kalman filter gain matrix and the innovations covariance matrix are given by

$$\begin{aligned} K &= CF^{-1} && (88)\\ &= O_L^\dagger R_{33}(m+1 : (L+1)m, 1 : m) R_{33}^{-1}(Lm+1 : (L+1)m, Lm+1 : (L+1)m) \\ \Delta &= FF^T. && (89) \end{aligned}$$

3.4 SID of the deterministic subsystem

The parameters in the B and E matrices can be computed from an optimal least squares problem. A solution to this is given in the following

Lemma 3.12 (Extended state space model)
The states can be eliminated from the state space model (1) and (2) to yield the so called Extended State Space Model (ESSM)

$$Y_{J+1|L} = \tilde{A}_L Y_{J|L} + \tilde{B}_L U_{J|L+g} + \tilde{C}_L E_{J|L+1}, \tag{90}$$

where

$$\begin{aligned} \tilde{A}_L &\overset{\text{def}}{=} O_L A (O_L^T O_L)^{-1} O_L^T && \in \mathbb{R}^{Lm \times Lm}, && (91)\\ \tilde{B}_L &\overset{\text{def}}{=} \begin{bmatrix} O_L B & H_L^d \end{bmatrix} - \tilde{A}_L \begin{bmatrix} H_L^d & 0_{Lm \times r} \end{bmatrix} && \in \mathbb{R}^{Lm \times (L+g)r}, && (92)\\ \tilde{C}_L &\overset{\text{def}}{=} \begin{bmatrix} O_L C & H_L^s \end{bmatrix} - \tilde{A}_L \begin{bmatrix} H_L^s & 0_{Lm \times m} \end{bmatrix} && \in \mathbb{R}^{Lm \times (L+1)m}. && (93) \end{aligned}$$

Proof 3.10 *Putting $J =: J+1$ in (37) and substituting $X_{J+1} = AX_J + BU_J + CE_J$ into this equation gives*

$$Y_{J+1|L} = O_L A X_J + \begin{bmatrix} O_L B & H_L^d \end{bmatrix} U_{J|L+g} + \begin{bmatrix} O_L C & H_L^s \end{bmatrix} E_{J|L+1}. \tag{94}$$

Equation (37) can be solved for X_J when (A, D) is observable, i.e.,

$$X_J = O_L^\dagger (Y_{J|L} - H_L^d U_{J|L+g-1} - H_L^s E_{J|L}), \tag{95}$$

where $O_L^\dagger = (O_L^T O_L)^{-1} O_L^T$ is the More-Penrose pseudo inverse of O_L. Substituting (95) into (94) gives (90)-(93). □

Lemma 3.13 (Projection matrix for the deterministic subsystem) *Define the projection matrix*

$$Z^d_{J|L+1} = Y_{J|L+1} / \begin{bmatrix} U_{J|L+g} \\ U_{0|J} \\ Y_{0|J} \end{bmatrix}. \tag{96}$$

This matrix can be partitioned into matrices $Z^d_{J+1|L}$ and $Z^d_{J|L}$ which satisfy the deterministic model

$$Z^d_{J+1|L} = \tilde{A}_L Z^d_{J|L} + \tilde{B}_L U_{J|L+g}. \tag{97}$$

Proof 3.11 *This follows from (90) and (13) and that*

$$\lim_{K\to\infty} \tilde{C}_L E_{J|L+1} / \begin{bmatrix} U_{J|L+g} \\ U_{0|J} \\ Y_{0|J} \end{bmatrix} = 0, \tag{98}$$

when the future inputs, $U_{J|L+g}$, the past data, $U_{0|J}$ and $Y_{J|J}$, are all independent of the future noise term, $\tilde{C}_L E_{J|L+1}$. □

Lemma 3.14 (SID of the system matrices B and E) *Define from (97) the linear equation*

$$\mathcal{Y} = \tilde{B}_L \mathcal{U}, \tag{99}$$

where

$$\mathcal{Y} = Z^d_{J+1|L} - \tilde{A}_L Z^d_{J|L}, \tag{100}$$
$$\mathcal{U} = U_{J|L+g}. \tag{101}$$

From (99) a least squares problem

$$\min_{B,E} \parallel \mathcal{Y} - \tilde{B}_L(B,E)\mathcal{U} \parallel_F^2 \tag{102}$$

for the unkown elements in B and E are defined as

$$vec(\mathcal{Y}) = \mathcal{X} vec(\begin{bmatrix} B \\ E \end{bmatrix}), \tag{103}$$

which can be solved for the unknown parameters as

$$vec(\begin{bmatrix} B \\ E \end{bmatrix}) = \mathcal{X}^\dagger vec(\mathcal{Y}), \tag{104}$$

where $\mathcal{X}^\dagger = (\mathcal{X}^T\mathcal{X})^{-1}\mathcal{X}^T$. The matrix $\mathcal{X}$ is defined as follows

$$\mathcal{X} \overset{\text{def}}{=} \sum_{i=1}^{L+g} R_i^T \otimes (E_{i-1} - \tilde{A}_L E_i) \ \in \ \mathbb{R}^{LmK\times(n+gm)r}, \tag{105}$$

where $\otimes$ denotes the Kronecker tensor product. The matrices R_i and E_i are defined in the following. The matrices

$$R_i \in \mathbb{R}^{r\times K} \;\forall\; i = 1, \cdots, L+g, \tag{106}$$

are r-block rows in the $\mathcal{U} \in \mathbb{R}^{(L+g)r\times K}$ matrix. I.e. extracted from

$$\mathcal{U} = \begin{bmatrix} R_1 \\ \vdots \\ R_{L+g} \end{bmatrix}. \tag{107}$$

The matrices

$$E_{i-1} \in \mathbb{R}^{Lm\times(n+mg)} \;\forall\; i = 1, \cdots, L+g, \tag{108}$$

are defined as follows:

$$E_0 = \begin{bmatrix} O_L & 0_{Lm\times m}, \end{bmatrix}, \quad E_{L+1} = 0_{Lm\times(n+m)}, \tag{109}$$

$$E_1 = \begin{bmatrix} 0_{m\times n} & I_{m\times m} \\ D & 0_{m\times m} \\ DA & 0_{m\times m} \\ \vdots & \vdots \\ DA^{L-2} & 0_{m\times m} \end{bmatrix}, E_2 = \begin{bmatrix} 0_{m\times n} & 0_{m\times m} \\ 0_{m\times n} & I_{m\times m} \\ D & 0_{m\times m} \\ \vdots & \vdots \\ DA^{L-3} & 0_{m\times m} \end{bmatrix}, E_L = \begin{bmatrix} 0_{m\times n} & 0_{m\times m} \\ 0_{m\times n} & 0_{m\times m} \\ 0_{m\times n} & 0_{m\times m} \\ \vdots & \vdots \\ 0_{m\times n} & I_{m\times m} \end{bmatrix} \tag{110}$$

The matrix $I_{m\times m}$ denotes the $m \times m$ identity matrix.

Proof 3.12 *From (92) we have that $\tilde{B}_L$ is a linear function of B and E when A and D are given. The matrix $\mathcal{X}$ in the LS problem (103) is defined from $\tilde{B}_L\mathcal{U}$ by using the identity*

$$vec(AXB) = (B^T \otimes A)vec(X). \tag{111}$$

□

Note that the number of columns in $\mathcal{Y}$ and $\mathcal{U}$, which is defined in (100) and (101), can be reduced to $K =: (L+g)r$ by post-multiplying both (100) and (101) with $U^T_{J|L+g}$. However, this does not affect the estimates of B and E but will in general reduce the computation. Another variant, which should be preferred for numerical reasons, is to define $\mathcal{Y}$ and $\mathcal{U}$ from the R matrix provided by the RQ/LQ decomposition. This will also reduce the number of columns to $K =: (L+g)r$ in Lemma 3.14.

Note that only a matrix of size $(n+gm)r \times (n+gm)r$ has to be inverted (i.e., the matrix $\mathcal{X}^T\mathcal{X}$ in the LS solution (104)) in order to solve for the unknown parameters in B and E. This method combined with the LQ decomposition is found to be very efficient. The method in Lemma 3.14 require only that the input is exciting of order $n+gm$, and hence, independent of the user specified parameters L and J. This is consistent with the lower bound on the order

of persistence of excitation for consistent estimation of an nth order possibly proper ($g = 1$) linear system.
Note that the alternative strategy of first solving for $\tilde{B}_L$ in (97) and then extracting B and E would require the inversion of an $(L+g)r \times (L+g)r$ matrix $U_{J|L+g}U_{J|L+g}^T$. This matrix may be singular for colored input signals, and hence is not preferred.

4 Closed loop subspace identification

We have in Section 3.2 shown that the extended observability matrix O_{L+1} can be estimated from the column space of the projection matrix $Z_{J|L+1}$ as defined in (52). Let us look at the error term in this projection. We have

$$Z_{J|L+1} = O_{L+1}X_J^a + dZ, \tag{112}$$

The error term dZ is given by

$$\begin{aligned} dZ &= H_{L+1}^s(E_{J|L+1}/\begin{bmatrix} U_{J|L+g} \\ W \end{bmatrix})U_{J|L+g}^\perp \\ &= H_{L+1}^s E_{J|L+1}U_{J|L+g}^\perp W^T(WU_{J|L+g}^\perp W^T)^{-1}WU_{J|L+g}^\perp, \\ &\approx -H_{L+1}^s E_{J|L+1}/U_{J|L+g}W^T(WU_{J|L+g}^\perp W^T)^{-1}WU_{J|L+g}^\perp, \end{aligned} \tag{113}$$

where W is defined in (65). We have in the last expression in (113) used that $E_{J|L+1}W^T/K \approx 0$ when the number of columns K tends to infinity. The remaining projection in the error term is then $E_{J|L+1}/U_{J|L+g}$. This term will also be approximately zero for open loop and many closed loop problems, which will be pointed out in the following. However, the term $E_{J|L+1}/U_{J|L+g}$ may be non-zero and cause biased estimates for feedback systems in which the control is directly proportional to the innovations noise. We will in the next sections discuss how to overcome this problem. We will also stress that biased estimates may be more reliable than estimates from an unbiased algorithm because the variance may be small. This is illustrated in the section of examples.

4.1 Closed loop subspace identification: Using a filter in the feeback loop!

Since we are allowing the inputs to be colored the question whether it is possible with feedback in the inputs have to be pointed out. An external (dither) signal, i.e., a reference signal, should be used to excite the system when collecting data for subspace identification. The natural excitations from process disturbances are often insufficient. The SID algorithm, e.g. DSR, works perfect for closed loop deterministic systems. Our simulation results also shows that the results may be good even for combined deterministic and stochastic systems, however, the results depends on the dither signal or the signal to noise ratio. If the signal to noise ratio is low then there may be a bias in the estimates. However, the variance may be small. There may also exist an "optimal" dither signal which gives very accurate subspace estimates (small bias and small variance) even for systems with a large signal to noise ratio. This will be illustrated in Example 7.4. A white noise or random binary signal in the reference usually gives very good closed loop identification results. Furthermore, a minimum of measurements noise is, as always, to be preferred in order to obtain good closed loop estimates.

It is believed that SID of systems with state feedback or feedback from Kalman filter states would work well, provided an external dither signal is introduced in the loop. The reason for this is that the states are ”noise-free” and not correlated with the innovations noise. There are no problems by using subspace identification methods in these cases.

The key is to make the term $E_{J|L+1}/U_{J|L+g}$ small, which is equivalent to making the error term (39) small.

The (open loop) subspace identification methods may give biased estimates for closed loop systems as in Figures 1 and 3 when the signal-to-noise ratio is low. The reason for this is that the error term in (39) is not zero when the future inputs, $U_{J|L+g-1}$, are correlated with the future noise, $E_{J|L}$. Most of our simulations shows that the bias in the DSR algorithm, due to noisy closed loop data, is less than the bias in the other algorithms as N4SID, SUBID (Van Overschee and De Moor (1996)), MOESP (Verhagen (1994)).

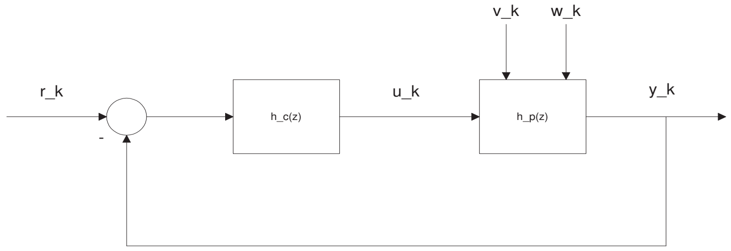

Figure 1: Standard feedback system with reference signal, r_k, controller represented with $u_k = h_c(z)(r_k - y_k)$ and the system represented with $h_p(z)$. Possibly process and measurements noise are represented with v_k and w_k, respectively.

One of our solutions to the bias problem is to include a filter, e.g. a first order low-pass filter, in the feedback path of the control system as illustrated in Figure 2. This filter will reduce or eliminate the feedback problem when using subspace identification algorithms. Hence, the input and output data, u_k and y_k, can be used directly to identify a state space model of the system. The reason for this is that the input is no longer directly proportional to the measurements noise w_k in the output $y_k = Dx_k + w_k$ (or the innovations e_k in the output $y_k = Dx_k + e_k$). This solution to the feedback problem will be illustrated in Example 7.3.

Note that the control system in Figure 2 is a special case of the more general control configuration in Figure 3.

Note that SID algorithms are very useful for model predictive control in which the control input signal often is the reference signal to some local controller for the process. Hence, the SID algorithm are used to identify the closed loop model from the reference signal to the output. In this case we do not have

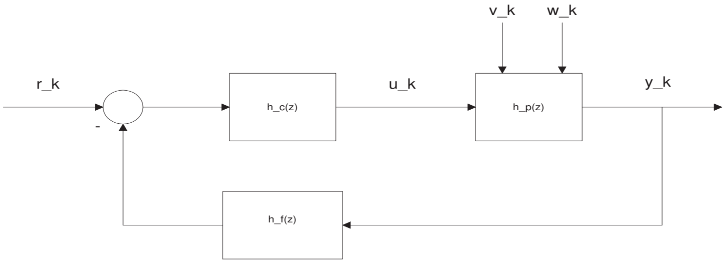

Figure 2: Feedback system with a filter in the feeback loop in order to eliminate problems with feeback in the data when using subspace identification algorithms. Here, u_k is the control/input signal, y_k the output signal and r_k, is the reference signal. The controller is represented with $u_k = h_c(z)(r_k - \bar{y}_k)$ where $\bar{y}_k = h_f(z)y_k$ is the filtered output. The system is represented with the transfer function $h_p(z)$. The controller is represented with the transfer function $h_c(z)$ and the filter is represented with the transfer function $h_f(z)$. Possibly process and measurements noise are represented with v_k and w_k, respectively.

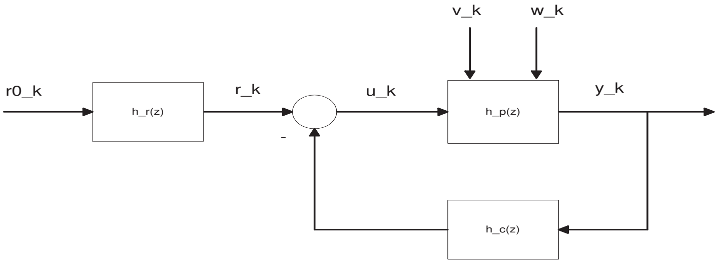

Figure 3: Alternative feedback system with reference signal, r_k, controller represented with $u_k = r_k - h_c(z)y_k$ and the system represented with $h_p(z)$. Possibly process and measurements noise are represented with v_k and w_k, respectively.

problems with feedback in the data.

4.2 Closed loop subspace identification: Using the controller

The knowledge of the controller or the reference signal can be used to obtain consistent subspace identification algorithms for closed loop systems. Consider a linear state space model of the controller in Figure 1, i.e.

$$\begin{aligned} x_{k+1}^c &= A_c x_k^c + B_c(r_k - y_k), & (114) \\ u_k &= D_c x_k^c + E_c(r_k - y_k), & (115) \end{aligned}$$

where $A_c \in \mathbb{R}^{n_c \times n_c}$, $B_c \in \mathbb{R}^{n_c \times m}$, $D_c \in \mathbb{R}^{r \times n_c}$ and $E_c \in \mathbb{R}^{r \times m}$ is the state space model matrices of the controller and $x_k^c \in \mathbb{R}^n$ is the controller state vector. Note also that the results which follows also holds for the control strategy in Figure 3.

We will in the following assume that the controller is linear and that the quadruple (A_c, B_c, D_c, E_c) matrices in addition to the input and output data, u_k and y_k, are known. One should also note that the linear controller matrices can be exactly identified if r_k, u_k and y_k are known. The problem of identifying the controller is deterministic and one can in this case perfectly identify (A_c, B_c, D_c, E_c) by using $r_k - y_k$ as inputs and u_k as outputs by using e.g. the DSR subspace algorithm (provided that noise-free r_k's and u_k's are given).

Consider the following matrix equation obtained from the state space model (114) and (115) of the controller in Figure 1, i.e.

$$U_{J|L} = O_L^c X_J^c + H_L^c(R_{J|L} - Y_{J|L}). \tag{116}$$

We will now adopt the idea in Van Overschee and De Moor (1997) and define a signal/matrix $M_{J|L}$ from (116) which is orthogonal to the future noise matrix $E_{J|L}$ as

$$M_{J|L} \overset{\text{def}}{=} U_{J|L} + H_L^c Y_{J|L} = O_L X_J^c + H_L^c R_{J|L} \tag{117}$$

The signal/matrix 117 was introduced by Van Overschee and De Moor (1997) in order to solve the bias problem in the subspace identification algorithms.

Note also that a similar signal can be defined from the control system in Figure 3. The only difference is that the right hand side of (117) becomes $R_{J|L} - O_L X_J^c$ in this case. This matrix is also orthogonal to the future noise and the closed loop subspace identification algorithm which will be presented in the following thus holds for both control systems as presented in Figures 1 and 3. The main point of introducing $M_{J|L}$ is that

$$E_{J|L}/M_{J|L} = 0, \tag{118}$$

which holds if $E_{J|L}$ is orthogonal to both X_J^c and $R_{J|L}$.

4.3 Closed loop subspace identification: Indirect method

We will in this section derive a consistent version of the closed loop subspace identification algorithm which is presented in Van Overschee and De Moor (1997). We have the following consistent projection lemma for closed loop subspace system identification

Lemma 4.1 (Closed loop SID)
Given the following closed loop projection

$$Z_{J|L+1} = (Y_{J|L+1} / \begin{bmatrix} M_{J|L+g} \\ U_{0|J} \\ Y_{0|J} \end{bmatrix}) M^{\perp}_{J|L+g} \tag{119}$$

Then, $Z_{J|L+1}$ is related to the extended observability matrix O_{L+1} as

$$Z_{J|L+1} = T^{-1} O_{L+1} X^a_J, \tag{120}$$

where $T \in \mathbb{R}^{(L+1)m \times (L+1)m}$ is a lower block Toepliz matrix given by

$$\begin{aligned} T &= I_{(L+1)m} + H^d_{L+1} H^c_{L+g} && \textit{when } g = 1 \quad (121) \\ T &= I_{(L+1)m} + \begin{bmatrix} H^d_{L+1} H^c_{L+g} & 0_{(L+1)m \times m} \end{bmatrix} && \textit{when } g = 0 \quad (122) \end{aligned}$$

Furthermore, the system order is given by

$$n = \textit{rank}(Z_{J|L+1}). \tag{123}$$

From the SVD

$$Z_{J|L+1} = U_1 S_1 V_1^T + U_2 S_2 V_2^T, \tag{124}$$

where the n large/dominant singular values are contained on the diagonal in S_1 and the other zero/smaller singular values on the diagonal of S_2. Furthermore, $U_1 \in \mathbb{R}^{(L+1)m \times n}$ and $U_2 \in \mathbb{R}^{(L+1)m \times ((L+1)m-n)}$.
From this we have the estimate

$$T^{-1} O_{L+1} = U_1. \tag{125}$$

Furthermore, the "autonomous" states are determined as $X^a_J = S_1 V_1^T$.

Proof 4.1 *A simple proof is given as follows. From (26) we have*

$$Y_{J|L+1} = O_{L+1} X_J + H^d_{L+1} U_{J|L+g} + H^s_{L+1} E_{J|L+1}. \tag{126}$$

Adding $H^d_{L+1} H^c_{L+g} Y_{J|L+g}$ on both sides of (126) and using the definition in (117) gives

$$T Y_{J|L+1} = O_{L+1} X_J + H^d_{L+1} M_{J|L+g} + H^s_{L+1} E_{J|L+1}. \tag{127}$$

where T is as in (121) and (122). Since the matrix $\begin{bmatrix} M_{J|L+g} \\ U_{0|J} \\ Y_{0|J} \end{bmatrix}$ *is uncorrelated (orthogonal) to the future noise matrix* $E_{J|L+1}$ *we have that*

$$TZ^d_{J|L+1} = O_{L+1}X^d_J + H^d_{L+1}M_{J|L+g} \tag{128}$$

where

$$Z^d_{J|L+1} =: Y_{J|L+1}/\begin{bmatrix} M_{J|L+g} \\ U_{0|J} \\ Y_{0|J} \end{bmatrix}, \tag{129}$$

and

$$X^d_{J|L+1} =: X_J/\begin{bmatrix} M_{J|L+g} \\ U_{0|J} \\ Y_{0|J} \end{bmatrix}, \tag{130}$$

The lower triangular matrix T is non-singular if $I_m + EE_c$ is non-singular. Hence,

$$Z^d_{J|L+1} = T^{-1}O_{L+1}X^d_J + T^{-1}H^d_{L+1}M_{J|L+g} \tag{131}$$

Post-multiplication of (131) with $M^\perp_{J|L+g}$ proves (119) -(121).

Furthermore we have that

$$U_2^T Z^d_{J|L+1} = U_2^T T^{-1}H^d_{L+1}M_{J|L+g} \tag{132}$$

where U_2 is the left singular vectors from the SVD of $Z_{J|L+1}$ which is related to the "zero/small" singular values. Equation (132) is obtained by pre-multiplying (131) with U_2^T and using that $T^{-1}O_{L+1} = U_1$ and $U_2^T U_1 = 0$.
Equation (132) is a linear equation of the elements in the lower block triangular Toepliz matrix $T^{-1}H^d_{L+1}$. The solution to this problem is an important step in the closed loop subspace algorithm and therefore needs further discussion. In order to do this we write (132) as

$$\mathcal{Y} = \mathcal{U}\mathcal{K}M_{J|L+g}, \tag{133}$$

where

$$\mathcal{Y} \;=:\; U_2^T Z^d_{J|L+1} \tag{134}$$

$$\mathcal{U} \;=:\; U_2^T \tag{135}$$

$$\mathcal{K} \overset{\text{def}}{=} T^{-1}H^d_{L+1} \;\in\; \mathbb{R}^{(L+1)m\times(L+g)r} \tag{136}$$

The matrix $\mathcal{K}$ is a lower block triangular Toepliz matrix with $m \times r$ blocks K_i $\forall\ i = 1,\ldots,L+1$. Hence, we can solve (132) (or equivalent (133) in a least

squares optimal sense for the parameters in $\mathcal{K}$. An algorithm for doing this is presented in Van Overschee and De Moor (1996b) Appendix C.
The problem of solving (132) is very similar to the least squares problem in Section 3.4 for determining B and E for the open loop subspace problem.
Since H^c_{L+g} is known the matrix H^d_{L+1} is simply obtained from $\mathcal{K}$ as (when $g = 1$)

$$H^d_{L+1} = \mathcal{K}(I_{(L+1)r} - H^c_{L+g}\mathcal{K})^{-1}. \tag{137}$$

Finally, the extended observability matrix of the system can be obtained as (when $g = 1$)

$$O_{L+1} = (I_{(L+1)m} + H^d_{L+1}H^c_{L+g})U_1. \tag{138}$$

The system matrices A and D are then computed from O_{L+1}. The system matrices B and E are computed from H^d_{L+1} and O_L.
The stochastic part of the system is determined very similar to the theory in Section 3.3. The difference is that the projection matrix now is given by

$$Z^s_{J|L+1} = Y_{J|L+1} - Y_{J|L+1}/\begin{bmatrix} M_{J|L+g} \\ U_{0|J} \\ Y_{0|J} \end{bmatrix} = T^{-1}H^s_{L+1}E_{J|L+1} \tag{139}$$

The projection matrices $Z_{J|L+1}$, $Z^d_{J|L+1}$ and $Z^s_{J|L+1}$ can effectively be computed by the LQ/QR decomposition.
The above ideas is among other details used to construct a MATLAB function, **dsr_cl.m**, for consistent closed loop subspace identification.

4.4 Closed loop subspace identification: Direct method

A drawback with the above algorithm is that T and H^d_{L+1} have to be identified before the extended observability matrix O_{L+1} and the system matrices A and D could be identified. We will in this section present a solution to the closed loop subspace identification problem which is more consistent with the DSR subspace algorithm in which O_{L+1} and A and D is identified directly in a first step. We have the following lemma

Lemma 4.2 (Direct closed loop SID)
The extended observability matrix O_{L+1} is obtained from the following projection

$$\begin{aligned} Z_{J|L+1} &= Z^d_{J|L+1}(M_{J|L+g} - H^c_{L+1}Z^d_{J|L+1})^{\perp} \\ &= Z^d_{J|L+1}(U_{J|L+g} + H^c_{L+1}Z^s_{J|L+1})^{\perp} = O_{L+1}X^a_J \end{aligned} \tag{140}$$

where $Z^d_{J|L+1}$ and $Z^s_{J|L+1}$ are defined in (129) and (139), respectively. Furthermore, B and E (or also H^d_{L+1}) can be obtained from

$$Z^d_{J|L+1} = O_{L+1}X^d_J + H^d_{L+1}(M_{J|L+g} - H^c_{L+1}Z^d_{J|L+1}) \tag{141}$$

or from (as in the standard DSR algorithm) the equation

$$Z^d_{J+1|L+1} - \tilde{A}_L Z_{J|L} = \tilde{B}_L(M_{J|L+g} - H^c_{L+1}Z^d_{J|L+1}) \tag{142}$$

Proof 4.2 *From (128) we have that*

$$(I + H^d_{L+1}H^c_{L+g})Z^d_{J|L+1} = O_{L+1}X^d_J + H^d_{L+1}M_{J|L+g}. \tag{143}$$

Rearranging (143) gives

$$Z^d_{J|L+1} = O_{L+1}X^d_J + H^d_{L+1}(M_{J|L+g} - H^c_{L+g}Z^d_{J|L+1}) \tag{144}$$

The closed loop subspace algorithm which is presented in this section is very similar to the open loop subspace identification algorithm which is presented in Section 3. The only difference is that the projection matrices are modified to incorporate the Markov parameters of the controller (the Toepliz matrix H^c_{L+g}.

5 A new subspace identification method for closed and open loop systems

It was presented in Di Ruscio (1995) and proved in Di Ruscio (2001) that the noise innovation process could be identified directly from the data in a first step. This approach is valid for both open and closed loop systems.

We will in the following consider a closed loop system in which it make sense to assume that $E = 0$. Putting $g = 0$ and letting $J \to \infty$ in Equation (27) gives

$$Y_{J|1} = D \begin{bmatrix} \tilde{C}^d_J & \tilde{C}^s_J \end{bmatrix} \begin{bmatrix} U_{0|J} \\ Y_{0|J} \end{bmatrix} + FE_{J|1}. \tag{145}$$

Hence, the innovations can simply be identified as (for $g = 0$)

$$Z^s_{J|1} = FE_{J|1} = Y_{J|1} - Y_{J|1}/ \begin{bmatrix} U_{0|J} \\ Y_{0|J} \end{bmatrix} \tag{146}$$

This result is also obtained directly from Lemma 3.9, and Di Ruscio (2001). It is clear that the above approach is valid for both open and closed loop systems since the past data, $U_{0|J}$ and $Y_{0|J}$, is uncorrelated with the future noise $E_{J|1}$.

The innovations is known and the DSR algorithm is used directly with the innovations as extra inputs in order to identify the "deterministic" model

$$\begin{aligned} x_{k+1} &= Ax_k + \begin{bmatrix} B & K \end{bmatrix} \begin{bmatrix} u_k \\ \varepsilon_k \end{bmatrix}, & (147) \\ y_k - \varepsilon_k &= Dx_k. & (148) \end{aligned}$$

We now simply solve a deterministic subspace identification problem in order to obtain the system matrices A, B, K and D. Any efficient subspace method, e.g. the DSR method as presented earlier in this paper, can be used.

The DSR algorithm for closed loop systems is sketched simply as follows:

1. From the known input and output data u_k and y_k ∀ $k = 0, 1, \ldots, N-1$ and a large past horizon, J, the future innovations matrix $Z^s_{J|1} = FE_{J|1}$ is computed from (146). The projection involved can be computed directly from the definition or preferably from a QR decomposition.

2. The innovations sequence $\varepsilon_k \ \forall\ k = J, J+1, \dots, N-1$ in the Kalman filter is given directly from the corresponding data in $Z^s_{J|1}$. From the known innovations sequence we form the Hankel matrix $E_{J|L+1}$.

3. Define the known input and output data sequence u_k and y_k $\forall\ k = J, J+1, \dots, N-1$ and form the matrices $U_{J|L}$ and $Y_{J|L+1}$.

4. The data from step 2 and 3 and a specified future horizon, L, are used to identify the state space model by an efficient subspace identification method. As a rule of thumb, chose L small such that $1 \leq n \leq Lm$. The theory in Sections 3.2 and 3.4 can be used directly by zeroing out the matrices $U_{0|J}$ and $Y_{0|J}$ from the projections.

Hence, we simply obtain the model matrices in the Kalman filter from the projection equations

$$Z_{J|L+1} = Y_{J|L+1} \begin{bmatrix} U_{J|L} \\ E_{J|L+1} \end{bmatrix}^{\perp} \approx O_{L+1} X^a_J \tag{149}$$

which gives O_{L+1}, A and D and

$$Y_{J+1|L} = \tilde{A}_L Y_{J|L} + \begin{bmatrix} \tilde{B}_L & \tilde{C}_L \end{bmatrix} \begin{bmatrix} U_{J|L} \\ E_{J|L+1} \end{bmatrix} \tag{150}$$

gives B and K according to the theory in Section 3.4. This algorithm is implemented in the **DSR_e** MATLAB function in the D-SR Toolbox for MATLAB.

6 Further remarks

6.1 Choice of algorithm parameters

There are two parameters in the algorithm, i.e., L and J. L is interpreted as the identification-horizon used to predict the number of states. J is interpreted as the horizon (into the past) used to define instruments from the data which are used to remove noise. The system order, n, which is specified or identified, is bounded by the user-specified parameter L, i.e. so that, $1 \leq n \leq mL$ where m is the number of outputs. Hence, Lm singular values are computed by the algorithm and the user may chose the system order, n, by inspection of the, n, non-zero singular values.

A rule of thumb is that L should be chosen as small as possible if the inputs are poorly exciting. The minimum identification-horizon, $L_{\min}$, so that the pair (D, A) is observable and $\text{rank}(O_{L_{\min}}) = n$ is bounded by, $\lceil \frac{n}{m} \rceil \leq L_{\min} \leq n - \text{rank}(D) + 1$, where $\lceil \cdot \rceil$ is the ceiling function, i.e., rounding towards plus infinity. If the outputs are independent, then, we suggest to use $L_{\min} = n - m + 1$ when $n \geq m$ and $L_{\min} = 1$ when $n = m$. If the inputs are rich, e.g. white, then this point is not critical. In practice, it is suggested that model validation on independent data is taken into consideration when choosing the "optimal" settings for L.

The past horizon, J, may for combined deterministic and stochastic systems and for pure stochastic systems usually be chosen as $J = L + 1$ or $J = L$. Note that the estimates of C and the Kalman filter gain matrix $K = CF^{-1}$ usually becomes better when J increases. For pure deterministic systems we may chose $J = 1$. The instruments $Y_{0|J}$ and $U_{0|J}$ can also be removed from the projections, i.e., and putting $J = 0$, in this case.

The theory in this paper is the basis for the D-SR Toolbox for MATLAB which are available upon request. The toolbox consists of MATLAB functions for subspace system identification of both open and closed loop systems.

6.2 Choice of input signal

The subspace identification methods tends to be more sensitive to the input signal compared to e.g, the Prediction Error Method (PEM). This means that there may exist colored input signals which gives subspace estimates which are as optimal (efficient and consistent) as the PEM estimates. On the other side there may exist colored input signals where the subspace methods gives poorer results compared to the PEM. An optimal experiment for the subspace methods is in general not a white noise input, but rather a colored input signal where the frequency spectrum is optimized to excite the parameters in the system as well as possible. Our experience is also that an input signal which are minimizing the condition number of the Hankal matrix $U_{0|J}$ or $U_{J|L+g}$, is usually not an optimal input signal.

6.3 N4SID

The N4SID algorithms in Van Overschee and De Moor (1994) are using an oblique-projection

$$\mathcal{O}_i = Y_{i|i} U_{i|i}^{\perp} W_p^T (W_p U_{i|i}^{\perp} W_p^T)^{-1} W_p, \tag{151}$$

$$W_p = \begin{bmatrix} U_{0|i} \\ Y_{0|i} \end{bmatrix}, \tag{152}$$

for the identification of the extended controllability matrix O_i, i.e., O_i is estimated from the column space of $\mathcal{O}_i$ in (151), e.g. using the SVD. Comparing (64) with (151) shows that the extra projection matrix $U_{i|i}^{\perp}$ are missing on the left hand side of (151). Hence, we conclude that in general

$$\mathcal{O}_i \neq O_i X_i^a. \tag{153}$$

The consequence of this is that the subspace identification theorems in Van Overschee and De Moor (1994), (1996), (1997) which are using the oblique projection, to our understanding, are wrong.

The extra projection matrix $U_{i|i}^{\perp}$ on the left hand side of (151) removes the influence of the future inputs on the future outputs, and is necessary in order to obtain accurate/consistent subspace estimates for colored input signals. Hence, a consistent projection is therefore

$$Z_{i|i} = \mathcal{O}_i U_{i|i}^{\perp} = O_i X_i^a U_{i|i}^{\perp}. \tag{154}$$

Hence, the extra projection can not be considered as a weighting matrix but as a projection matrix. One should note that the parameter i used by N4SID is related to the parameter L in DSR as $i = L + 1$. Furthermore, in N4SID the past horizon is put equal to the future horizon. Hence, this corresponds to putting $J = L + 1 = i$ in DSR.

7 Numerical examples

7.1 Example 1

Given the system (1) and (2) with the following matrices and vectors

$$A = \begin{bmatrix} 0 & 1 \\ -0.7 & 1.5 \end{bmatrix}, \; B = \begin{bmatrix} 0.25 \\ 0.625 \end{bmatrix}, \; C = \begin{bmatrix} 0.5 \\ 0.5 \end{bmatrix}, \tag{155}$$

$$D = \begin{bmatrix} 1 & 0 \end{bmatrix}, \; E = 1, \; F = 1. \tag{156}$$

The following colored input signals where used for identification

$$u_k^1 = \sin(k) + \sin(\frac{k}{2}), \tag{157}$$

$$u_k^2 = \sin(k) + \sin(\frac{k}{2}) + \sin(\frac{k}{3}), \tag{158}$$

$$u_k^3 = \text{withe noise with variance } \mathrm{E}(u_k^2) = 1. \tag{159}$$

The number of samples was $N = 1000$. The system was simulated 100 times, each time with the same input but with a different noise realization e_k. However, with the same covariance $\mathrm{E}(e_k^2) = 1$. The DSR parameters where chosen as $L = 2$, $J = 3$. The model structure parameter where $g = 1$. The poles of the 100 identified system matrices are illustrated in Figure 4, 5 and 6. From this we conclude that the method presented in this paper is almost as efficient as the PEM method, for the inputs which are considered. However, the N4SID oblique-method gives an unacceptable bias in the pole estimates for input u_k^1, i.e., because the estimated poles are unstable. From Figure 5 we see that the bias is eliminated but the results from N4SID are highly uncertain. However, the results from N4SID are nearly the same as DSR for the white input signal u_k^3. We also see that the variance of the subspace estimates may be smaller for a colored input signal, Figure 5, than for the white noise input, Figure 6.

7.2 Example 2

We consider the following system

$$x_{k+1} = Ax_k + Bu_k + Cv_k, \tag{160}$$

$$y_k = Dx_k + w_k, \tag{161}$$

where the system matrices are the same as in Example 7.1. The process noise, v_k, and the measurements noise, w_k, are both white noise with standard deviation $\sqrt{\mathrm{E}(v_k^2)} = \sqrt{0.02} = 0.1458$ and $\sqrt{\mathrm{E}(w_k^2)} = \sqrt{0.002} = 0.0447$, respectively.

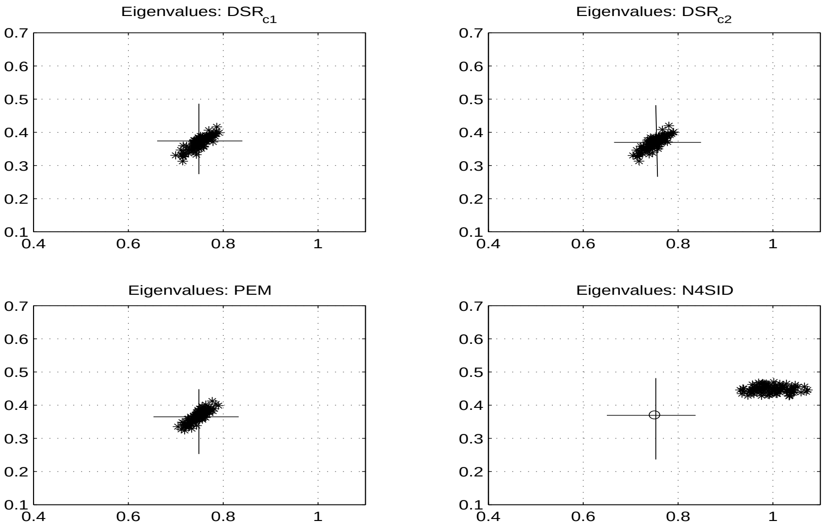

Figure 4: Identified poles for a Monte carlo simulation. The exact pole is marked with a cross. Input signal u_k^1 where used.

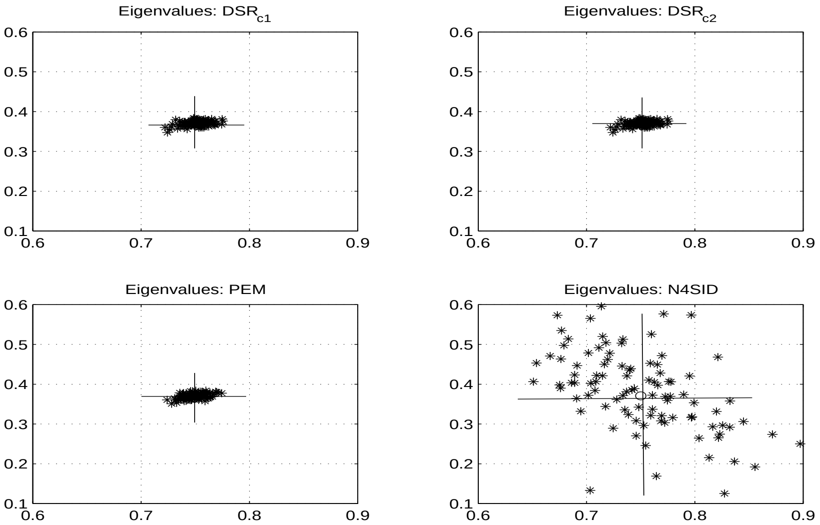

Figure 5: Identified poles for a Monte carlo simulation. The exact pole is marked with a cross.Input signal u_k^2 where used.

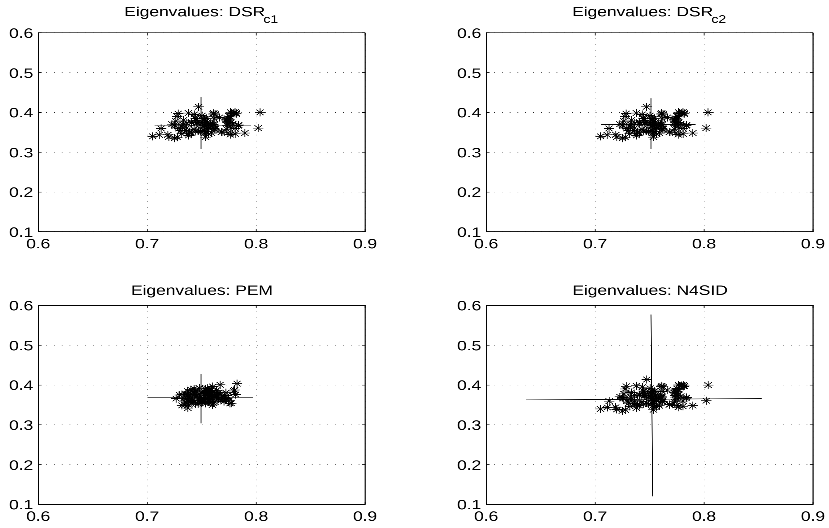

Figure 6: Identified poles for a Monte carlo simulation. The exact pole is marked with a cross.Input signal u_k^3 where used in this case.

The system is operating in closed loop. The input to the system is generated by the following discrete time PI-controller

$$u_k = K_p(r_k - y_k) + z_k, \tag{162}$$

where the controller state, z_k, is defined by

$$z_{k+1} = z_k + \frac{K_p}{T_i}(r_k - y_k). \tag{163}$$

The proportional constant is $K_p = 0.2$, the integral time is $T_i = 5$ and the reference, r_k, is taken as the binary signal in Figure 7.
The number of samples was $N = 1000$. The system was simulated 100 times, each time with the same reference, r_k, but with a different noise realizations v_k and w_k, but with the same variance. The DSR parameters where chosen as $L = J = 3$ and the structure parameter where $g = 0$. The subspace algorithms works perfect in the deterministic case. However, the algorithm gives a small bias in the estimates in the case of noise. The bias is negligible for this example. The pole estimates are presented in Figure 8.

7.3 Example 3

Consider the same closed loop example as in Example 2. We will in this example illustrate the bias-problem when using subspace identification algorithms directly from input and output data collected in closed loop (Figure 1). Furthermore, we will illustrate that the feedback problem can be eliminated by including a low-pass filter in the feedback as in Figure 2.
The process noise, v_k, and the measurements noise, w_k, are both white noise with standard deviation $\sqrt{\mathrm{E}(v_k^2)} = \sqrt{0.05} = 0.2236$ and $\sqrt{\mathrm{E}(w_k^2)} = \sqrt{0.01} =$

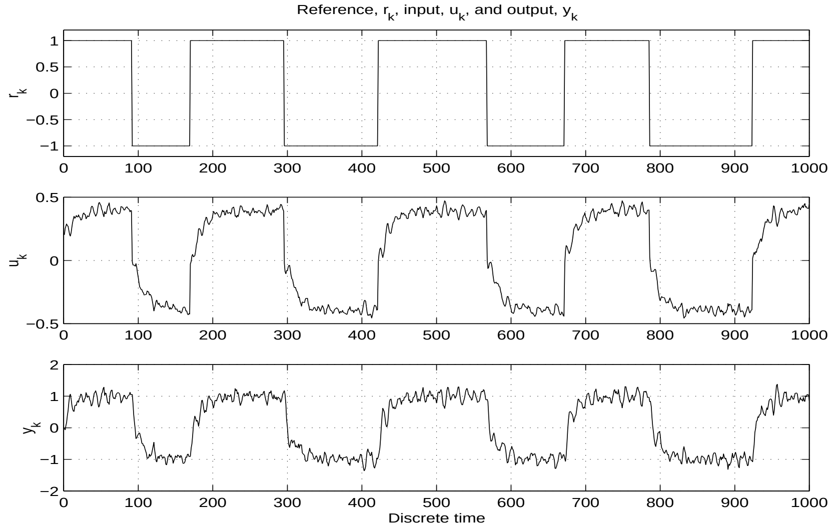

Figure 7: The reference signal, r_k, the input, u_k and the output y_k for two particular noise realizations v_k and w_k, used in Example 7.2.

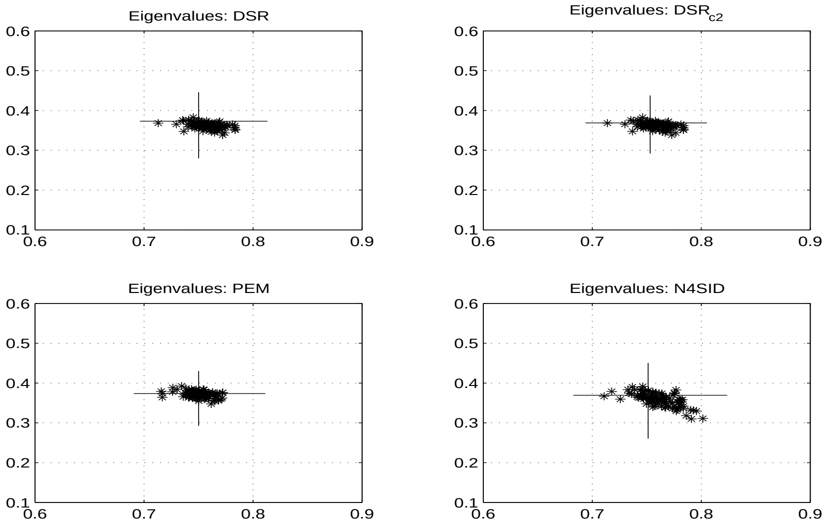

Figure 8: The pole estimates from the closed loop data as described in Example 7.2.

0.1, respectively. The pole estimates after a Monte carlo simulation is presented in Figure 9. We can clearly see a bias in the estimates from the (open loop) subspace identification algorithms. The bias in the DSR estimates is smaller than the bias in the estimates from both SUBID (Van Overschee and De Moor (1996)) and N4SID. This is also the conclusion from many other simulation experiments.

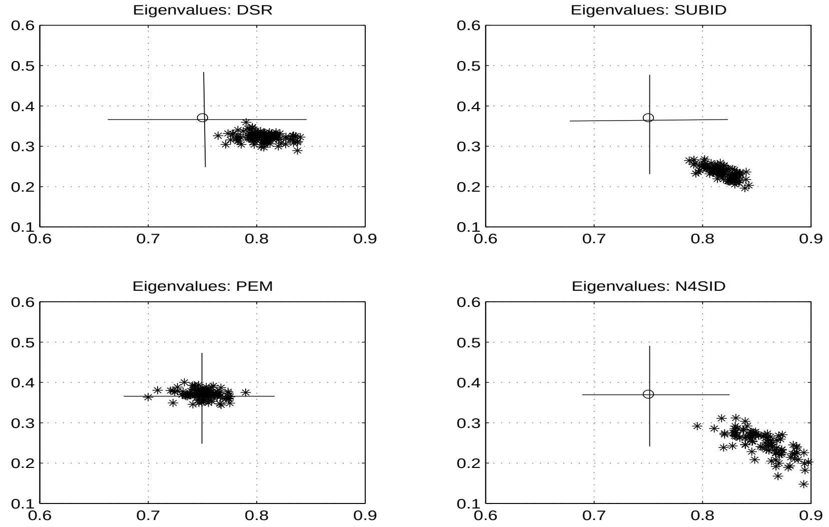

Figure 9: The pole estimates from the closed loop data as described in Example 7.3. The control system is as in Figure 1 with the same reference signal as in Figure 7.

Consider now the feedback system in Figure 2. We are using a PI-controller as in Example 7.2 where the filtered output is used as input to the controller. The controller equations are as follows.

$$u_k = K_p(r_k - \bar{y}_k) + z_k, \tag{164}$$

where the controller state, z_k, is defined by

$$z_{k+1} = z_k + \frac{K_p}{T_i}(r_k - \bar{y}_k). \tag{165}$$

The filter is a 1. order low-pass filter of the form

$$\begin{aligned} \bar{y}_{k+1} &= \bar{y}_k + K_f(y_k - \bar{y}_k) \\ &= (1 - K_f)\bar{y}_k + K_f y_k, \end{aligned} \tag{166}$$

with filter constant $K_f = 0.1$. The initial filter output is taken as $\bar{y}_0 = y_0$. Pole estimates after a Monte Carlo simulation is illustrated in Figure 10. We see that the pole estimates now are consistent.

7.4 Example 4

We will in this example search for an optimal experiment in the reference. Consider the reference $r_k = \sin(\omega k)$ for varying frequency ω. The following

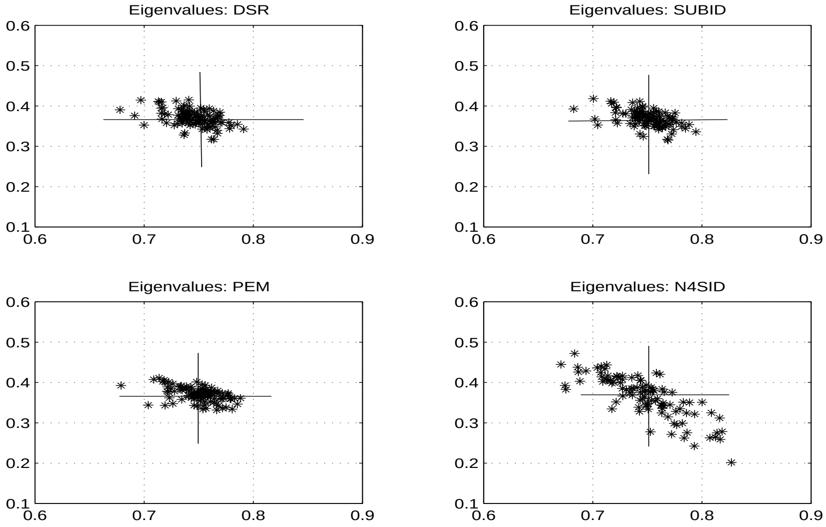

Figure 10: The pole estimates from the closed loop data as described in Example 7.3 with a filter in the feedback. The control system is as in Figure 2.

investigation shows that the bias in the DSR pole estimates is a function of the frequency and that the bias reach a minimum for a particular frequency.
The reference signal which gave the smallest bias in the pole estimates is for this example found to be for $\omega = \frac{1}{1.526}$, i.e.,

$$r_k = \sin(\frac{1}{1.526}k). \tag{167}$$

The pole estimates from an Monte carlo experiment is illustrated in Figure 11. The results are very interesting because, as we see, the pole estimates from the DSR subspace identification method is more reliable than the pole estimates from the prediction error method PEM.
The process noise, v_k, and the measurements noise, w_k, are both white noise with standard deviation $\sqrt{\mathrm{E}(v_k^2)} = \sqrt{0.1} = 0.01$ and $\sqrt{\mathrm{E}(w_k^2)} = \sqrt{0.1} = 0.01$, respectively. The DSR parameters is $L = 5$, $g = 0$ and $J = 6$.

7.5 Example 5

Consider the following system

$$\begin{aligned} x_{k+1} &= 0.9x_k + 0.5u_k + 0.6e_k && (168) \\ y_k &= x_k + e_k && (169) \end{aligned}$$

with e_k white noise with unit variance. The controller is taken as

$$u_k = K_p(y_k - r_k), \tag{170}$$

with $K_p = 0.6$ and a reference signal

$$r_k = \sin(0.5k) + \sin(k). \tag{171}$$

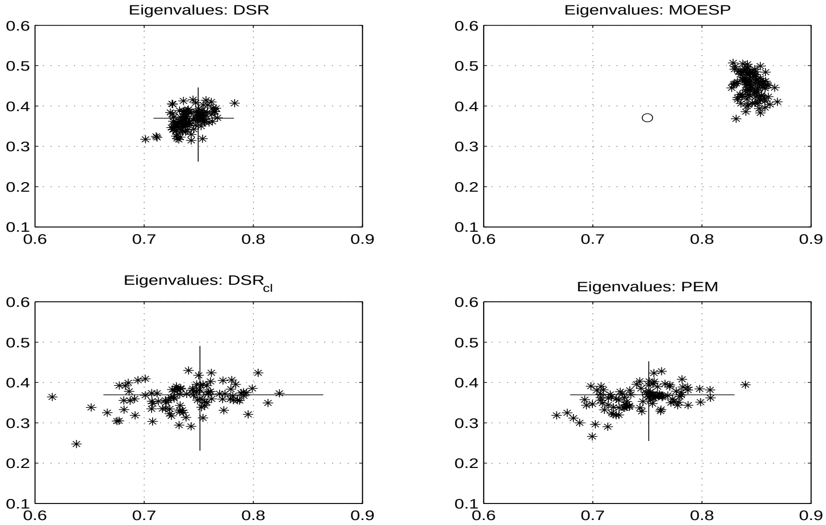

Figure 11: The pole estimates from the closed loop data as described in Example 7.4 with reference signal $r_k = sin(k/1.526)$. The control system is as in Figure 1.

A Monte carlo simulation study is performed with $N = 2000$ data points and $M = 100$ different noise realizations. The results are illustrated in Figures 12 and 13. As we can see, both DSR_e and PEM gives consisten results. It shows also that the subspace method, DSR_e, is as efficient that PEM is. The DSR function results in a smaller bias than MOESP. The parameters $L = 1$ and $J = 6$ was used for the DSR_e function. Parameters $L = 2$ and $J = 6$ for DSR and $i = L + 1 = 3$ for MOESP. The DSR_e function is implemented along the lines in Section 5.

The DSR algorithm gives usually less bias than MOESP for closed loop data, se Figures 12 and 13. It is very interestin that the DSR_e algorithm gives parameter estimates which are as optimal as the corresponding PEM estimates.

8 Conclusion

The extended observability matrix O_{L+1} can be computed directly from the column space of a projection matrix $Z_{J|L+1}$, which is defined in terms of the known data. There are in general two projections involved in order to define $Z_{J|L+1}$. One projection is used to remove the effect of noise and one projection is used to remove the effect of future inputs from the future outputs. A necessary condition for a consistent estimate of O_{L+1} is that the number of columns K in the data matrices tends to infinity.

The states are not needed in order to compute the extended observability matrix and, hence, to identify the system dynamics, i.e., the number of states n and the system matrices A and D.

An additional condition for a consistent state estimate is that the past horizon J has to tend to infinity. Furthermore, for colored input signals both the extended

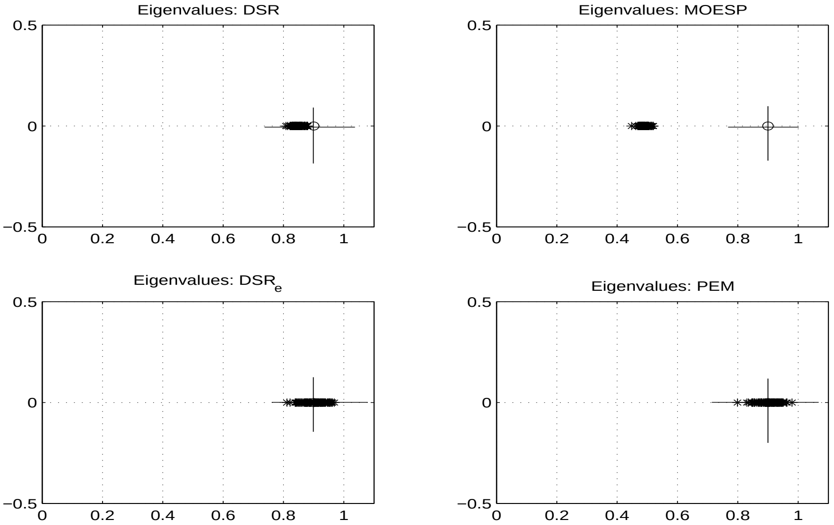

Figure 12: The pole estimates from the closed loop data as described in Example 7.5 with reference signal $r_k = sin(k) + sin(0.5k)$. The control system is as in Figure 1.

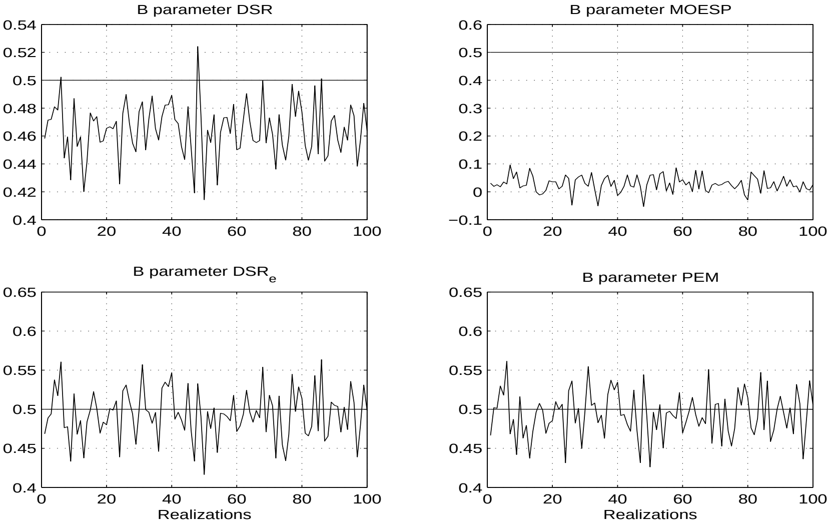

Figure 13: The estimates of the B parameter from the closed loop data as described in Example 7.5 with reference signal $r_k = sin(k) + sin(0.5k)$. The control system is as in Figure 1.

observability matrix O_L and the lower block triangular Toepliz matrix H_L^d has in general to be known in order to properly computing the states.
The stochastic part of the model, i.e., the Kalman filter gain matrix and the innovations covariance matrix can be identified directly from the data, i.e. from the projection matrix $Z_{J|L+1}^s$, without solving any Riccati or Lyapunov matrix equations.
The deterministic part of the model can be identified from an optimal least squares problem defined from the projection matrix $Z_{J|L+1}^d$, the extended observability matrix O_{L+1}, A and D.
Furthermore the necessary projections, $Z_{J|L+1}$, $Z_{J|L+1}^d$ and $Z_{J|L+1}^s$, which are needed in order to compute a complete state space model realization for the sixtuple matrices (A, B, D, E, C, F) (and/or K and $\Delta = \mathrm{E}(\varepsilon_k \varepsilon_k^T)$), can be computed throug a numerically stable LQ decomposition of $\begin{bmatrix} U_{J|L+g} \\ U_{0|J} \\ Y_{0|J} \\ Y_{J|L+1} \end{bmatrix}$. However, it is in general faster to compute $Z_{J|L+1}$, $Z_{J|L+1}^d$ and $Z_{J|L+1}^s$ directly from the definitions. This means that the algorithm both can be implemented as an correlation based method and a square root based method.
Finally, a method for subspace identification of closed loop systems which gives unbiased estimates is presented. Simulation results shows that the estimates are as efficient as those from the prediction error method, however, the estimates are somewhat dependent of the parameters L and J.

References

Di Ruscio, D. (2003). Subspace system identification of the Kalman filter. *Modeling, Identification and Control. Vol24, No.3, pp. 125-157.*

Di Ruscio, D. (2001). Subspace system identification of the Kalman filter gain. *The European Control Conference, ECC2001, Porto, Portogal 2001.*

Di Ruscio, D. (1997). A method for identification of combined deterministic stochastic systems. In: *Applications of Computer Aided Time Series Modeling*, Lecture Notes in Statistics 119, Eds. M. Aoki and A. M. Havenner, Springer Verlag, ISBN 0-387-94751-5.

Di Ruscio, D. (1997b). On Subspace Identification of the Extended Observability Matrix. In the proceedings of *the 36th Conference on Decision and Control 1997*, San Diego, California, December 6-14.

Di Ruscio, D. (1996). Combined Deterministic and Stochastic System Identification and Realization: DSR-a subspace approach based on observations. *Modeling, Identification and Control*, vol. 17, no.3.

Di Ruscio, D. (1995). A method for identification of combined deterministic stochastic systems: robust implementation. The *Third European Control Conference ECC95*, September 5-8, Rome, Italy.

D-SR Toolbox for MATLAB. Copyright D. Di Ruscio. Available upon request, david.di.ruscio@hit.no.

Larimore, W. E. (1990). Canonical Variate Analysis in Identification, Filtering and Adaptive Control. *Proc. of the 29th Conference on Decision and Control*, Honolulu, Hawaii, December 1990, pp. 596-604.

Kung, S. Y. (1978). A new identification and Model Reduction Algorithm via Singular Value Decomposition. *Conf. on Circuits, Systems and Computers*, Pacific Grove, CA, November 1978, pp. 705-714.

Van Overschee, P. and B. De Moor (1994). N4SID: Subspace Algorithms for the Identification of Combined Deterministic Stochastic Systems. *Automatica*, vol. 30, No. 1, pp.75-94.

Van Overschee, P. and B. De Moor (1996). *Subspace Identification for Linear Systems: theory-implementation-applications.* Kluwer Academic Publishers.

Van Overschee, P. and B. De Moor (1996b). Closed loop subspace system identification. Internal report. Katholieke Universiteit Leuven. Department Elektrotechniek. ESAT-SISTA/TR 1996-521.

Van Overschee, P. and B. De Moor (1997). Closed loop subspace system identification. In the proceedings of *the 36th Conference on Decision and Control 1997*, San Diego, California, December 6-14.

Verhagen, M. (1994). Identification of the deterministic part of MIMO state space models given on innovations form from input output data. *Automatica*, vol. 30, No. 1, pp. 61-74.

Viberg, M. (1995). Subspace-Based Methods for the Identification of Linear Time-invariant Systems. *Automatica*, vol. 31, No. 12, pp. 1835-1851.