Geir Werner Nilsen

TOPICS IN OPEN AND CLOSED LOOP SUBSPACE SYSTEM IDENTIFICATION: FINITE DATA-BASED METHODS.

Doctoral thesis
for the degree of doktor ingeniør

Trondheim, May 2006

Norwegian University of
Science and Technology

Department of Electrical Engineering, Information Technology,
and Cybernetics

TOPICS IN OPEN AND CLOSED LOOP SUBSPACE SYSTEM IDENTIFICATION: FINITE DATA-BASED METHODS.

Geir Werner Nilsen
Telemark University College
N-3914 Porsgrunn, Norway
E-mail: gw-nils@online.no

Porsgrunn May 26, 2006

Abstract

When subspace identification methods for finite closed loop data sets are presented they are often compared to poor subspace identification methods. The methods presented in this thesis are always compared to the results from the prediction error method implemented in Matlab 6.5.

The advantage of the subspace identification methods is the computational efficiency and the ability to estimate the system order, or alternatively help the user to choose the correct system order. One of the assumptions in subspace identification is that the input is uncorrelated to the noise at the output. In closed loop systems this assumption is not necessarily fulfilled. This can lead to a bias problem. Only the performance with finite data sets is considered in this thesis.

The thesis also presents the classic subspace identification algorithms DSR and N4SID together with an error-in-variable based subspace identification algorithm using the notation used in the outline of the DSR algorithm. The projections used in DSR to estimate the extended observability matrix, and the eigenvalues, are compared to the projections used in the other algorithms.

The effect of the parameter choice in DSR when used on finite closed loop data sets is presented. It is especially the estimation of the zeros which is hard using the classic subspace identification algorithms for direct closed loop identification. Solutions to help reducing the bias on the zeros are presented.

The closed loop can be modified to either reduce the noise in the feedback or make the noise through the feedback uncorrelated to the noise on the output. The effect of using different types of filters in the feedback loop is considered. The optimal filter used in the feedback is not the noise free output or a 1^{st} order low-pass filter rather the Kalman filter.

This leads to a new three-step closed loop subspace identifications algorithm based on the DSR algorithm and the Kalman filter properties. In an initial step DSR is used for identification of the process model, including the Kalman filter gain. The next step is to implement the Kalman filter in the feedback in such

a way that the controller uses the filtered output from the filter, not the actual process measurement. The final step is to use DSR to identify the process model when the feedback is filtered through the Kalman filter.

The goal for closed loop subspace identification algorithms is to be as easy to use for direct identification on finite closed loop data sets as the original subspace identification algorithms are on finite open loop data sets. In addition the closed loop subspace identification algorithms are to provide results comparable to the results from PEM in closed loop. The DSR_e algorithm is a modification of the existing DSR algorithm fulfilling these requirements. The algorithm is based on the fact that the noise innovation process can be identified directly from the data in a first step.

The system identification process is considered in two different ways. The first way is when all information regarding the process is considered as known and a benchmark is performed to see how good the performance can be. In this case DSR_e is comparable to PEM. The other is when the system order has to be estimated from the process data.

Methods to estimate the system order of systems operating in a closed loop by subspace identification are presented. The methods are meant to help users without experience in using subspace identification algorithms.

In addition a procedure is suggested combining the visual inspection of singular values from DSR_e and the search for the minimum prediction error. Visual inspection of the singular values gave the correct estimate of system order every time, independent of the choice of past and future horizons in DSR_e. The parameter settings for DSR_e found by searching for the minimum prediction error resulted in estimates comparable to the estimates from PEM. This indicates that this is a good practical approach for the use of DSR_e for direct closed loop system identification.

Preface

The work on this thesis has resulted in five publications.

Nilsen, G. W. and Di Ruscio, D. (2003). On the Total Least Squares and Ridge Regression Problems. *Nordic Process Control Workshop 11, Trondheim, Norway 2003.*

Nilsen, G. W. and Di Ruscio, D. (2004a). Using a dithering signal in the reference to improve the estimates from subspace identification methods on closed loop data. *The 7^{th} International Conference on Dynamics and Control of Process Systems, DYCOPS 7 - 2004, Boston, USA 2004.*

Nilsen, G. W. and Di Ruscio, D. (2004b). Using a dithering signal in the reference to improve the estimates from subspace identification methods on closed loop data. *The 8th World Multi-Conference on Systemics, Cybernetics and Informatics, SCI 2004, Orlando, Florida, USA 2004.*

Nilsen, G. W. and Di Ruscio, D. (2004c). A comparison of the estimates from subspace identification methods used on closed loop data. *Nordic Process Control Workshop 12, Gothenburg, Sweden 2004.*

Nilsen, G. W. and Di Ruscio, D. (2006). Closed Loop Subspace Identification. Modeling, Identification and Control, Vol. 26, No. 3.

The conference paper Nilsen and Di Ruscio (2003) was written to gain knowledge on regularization to be able to work further on the idea of regularization in DSR introduced in Di Ruscio (2003b). There has been no further work on the idea, therefore the results in the paper are not utilized in this thesis.

The rest of the publications are referred in the same manner as the other publications listed in the list of references.

Several persons have contributed to this thesis. My supervisor Dr. ing. David Di Ruscio, at the Department of Electrical Engineering, Information Technology and Cybernetics at Telemark University College has been a constant source of inspiration throughout this work. I would also like to thank Kjetil Fjalestad at

Norsk Hydro Research for interesting discussions. I acknowledge the assistance from Stewart Clark at Norwegian University of Science and Technology (NTNU) in editing the thesis.

Geir Werner Nilsen

Porsgrunn, May 2006

Contents

Chapter 1

Introduction

The reason for the problems that can occur when applying subspace identification (SID) algorithms for direct identification of closed loop data is the projection of the future outputs onto the future inputs. Future inputs and the noise on future outputs are assumed to be uncorrelated. In closed loop operation this assumption is not necessarily fulfilled. This is the cause of the bias. Solutions to overcome the problems that can occur when subspace methods are used on closed loop data have been suggested. Van Overschee and De Moor (1996), (1997) are using the Markov parameters of the controller in the algorithm to avoid the problem. The controller is assumed to be linear. Chou and Verhaegen (1997) have developed a SID algorithm for error-in-variable (EIV) problems. They have shown that this algorithm will give consistent estimates on closed loop data, if a persistence of excitation requirement is satisfied. This leads to the use of signals with relatively high order of persistent excitation in the reference. Gustafsson (2001) has proposed an Instrument Variable (IV) approach SID algorithm as an improvement of the already existing EIV algorithm of Chou and Verhaegen (1997). The modified algorithm is named SIV (Subspace-based Identification using instrumental Variables). Katayama et al. (2005b), Katayama (2005a) have developed a SID algorithm utilizing orthogonal projections.

There is no reason to state that these algorithms have solved the problems concerning using SID algorithms on closed loop data. The need for having knowledge about the controller and assuming it to be linear, like in Van Overschee and De Moor (1996), (1997), limits the applicability. Forssell and Ljung (1997) have commented that in industrial practice very few controllers are linear. Functions like the anti-windup function will introduce nonlinearities in the controller. Gustafsson (2001) argues that future inputs should not be used as one of the instruments in SIV. The argument is that if future inputs are included in the EIV case, the influence of noise will not vanish when the number of data points turns to infinity. In order to avoid the input being correlated with the output it is assumed that there is at least a delay in the controller. This will make the algorithm consistent

with the closed loop data. The SID algorithms for EIV problems need a high order of persistent excitation to give estimates with acceptable variance. One of the main causes of this need for signals with high order of persistent excitation is that the future inputs are not used in the instrument variables. The method developed by Katayama et al. (2005b) utilizes the references explicit in the algorithm in addition to the input and output data that the other methods utilizes. The references are assumed known and are limited to white noise only. This assumption limits the applicability.

This thesis is divided into six main chapters. Throughout the thesis only the performance on finite data sets will be considered. Chapter 2 contains the basic theory to understand the properties of the SID algorithms presented in Chapter 4. The examples used throughout the thesis are presented in Chapter 3.

Some of the classic SID algorithms are presented in Chapter 4 using the notation used in the outline of the SID algorithm DSR, Aoki and Havenner (1997), Di Ruscio (1995)-(2004). The projections used in DSR to estimate the extended observability matrix, and the eigenvalues, are compared to the projections used in other SID algorithms in Section 4.4. A special case is presented in Section 4.5 where a special choice of parameters eliminates the bias when DSR is used on closed loop data sets.

It is especially the estimation of the zeros which is hard using the classic SID algorithms for direct closed loop identification. The parameter choice in DSR when used on finite closed loop data sets is treated in Section 4.6. Dithering signals with different order of persistent excitation are used. Both dithering signals in the reference and on the input are considered.

An other approach to reduce the bias problem is to modify the closed loop, Chapter 5, to either reduce the noise in the feedback or make the noise through the feedback uncorrelated to the noise on the output. Di Ruscio (2003a) has already shown that using a filter in the feedback loop is a solution to reduce the bias on the estimated eigenvalues from SID algorithms caused by the error term that occur when the future inputs are correlated with the future noise on the output. The effect of using different types of filters in the feedback loop is investigated. A new three-step algorithm based on the DSR algorithm and the Kalman filter properties is introduced in Section 5.4.

A closed loop SID algorithm named DSR_e, Di Ruscio (2004), and simulation studies of it are treated in Chapter 6. The DSR_e algorithm is a modification of the existing DSR algorithm which gives consistent estimates on finite closed loop data sets. Jansson (2003) has introduced a SID algorithm which first estimates a higher order ARX model to get estimates of the impulse response coefficients.

Then projections are performed and the matrices in the state space model are estimated. A system identification method based on model reduction of a higher order ARX model is presented in Section 6.1. The method shows that when a higher order ARX model is identified there is no need for additional projections to estimate the system matrices. It is sufficient to perform a model reduction step. This means that the method presented in Section 6.1 is not a SID algorithm and therefore the method by Jansson (2003) can hardly be called a SID algorithm either.

Throughout this thesis there is no focus on how well the presented SID algorithms perform compared to the poorest SID algorithms. Instead it is focused on how well the algorithms perform compared to PEM (Prediction Error Method) in the system identification toolbox in Matlab. A description of prediction error methods in general and their properties can be found in Ljung (1999). The reason for using this benchmark is that the goal for developers of closed loop SID algorithms must be to introduce algorithms for direct identification which are as easy to use on finite closed loop data sets as the original SID algorithms are to use on finite open loop data sets, with estimates comparable to the estimates from PEM.

Software utilizing DSR_e for automatic identification of system order is presented in Section 6.4. A practical approach to subspace identification of systems operating in closed loop is presented in Section 6.5.

The main contributions in the thesis are as follows. The development of rules for the parameter choice in DSR when used on finite closed loop data sets in Section 4.6. The three-step algorithm based on the DSR algorithm and the Kalman filter properties presented in Section 5.4. The simulation studies in Section 6.3.1 - Section 6.3.5 comparing DSR_e with PEM. The development of software utilizing DSR_e for automatic identification of system order presented in Section 6.4. And the suggested practical approach to subspace identification of systems operating in closed loop presented in Section 6.5.

Chapter 2

Basic theory

2.1 Basic projections

Subspace-based identification methods are based on projections. The basic projections used are orthogonal projection and oblique projection. These projections are used in different ways in the different methods. By using the definition of the Moore-Penrose pseudo inverse of a matrix this makes the definition of the projections easier. The Moore-Penrose pseudo inverse of a matrix $A \in \mathbb{R}^{i \times k}$ where $k > i$ is defined as $A^{\dagger} = A^T(AA^T)^{-1}$.

2.1.1 Orthogonal projections

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

$$A/B \stackrel{def}{=} AB^T(BB^T)^{\dagger}B. \tag{2.1}$$

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

$$AB^{\perp} \stackrel{def}{=} A - A/B = A - AB^T(BB^T)^{\dagger}B. \tag{2.2}$$

Consistent with Equation (2.2) the definition

$$B^{\perp} \stackrel{def}{=} I_k - B^T(BB^T)^{\dagger}B \tag{2.3}$$

is introduced where I_k is the k x k identity matrix.

Frequently used properties are

$$A/\begin{bmatrix} A \\ B \end{bmatrix} = A \tag{2.4}$$

$$A/\begin{bmatrix} A \\ B \end{bmatrix}^{\perp} = 0 \tag{2.5}$$

$$(B^{\perp})^T = B^{\perp} \tag{2.6}$$

$$B^{\perp}B^{\perp} = B^{\perp}. \tag{2.7}$$

Proof of Equations (2.4) and (2.5) can be found in Di Ruscio (1997b).

2.1.2 Oblique projections

The oblique projection of the row space of $A \in \mathbb{R}^{i\times k}$ along the row space of $B \in \mathbb{R}^{j\times k}$ on the row space of $C \in \mathbb{R}^{l\times k}$ is defined as

$$A/_B C \stackrel{def}{=} (A/B^{\perp})(C/B^{\perp})^{\dagger}C. \tag{2.8}$$

Some properties of the oblique projection, van Overschee (1996), are

$$B/_B C = 0 \tag{2.9}$$

$$C/_B C = C. \tag{2.10}$$

2.2 State space model

A time invariant linear dynamic system can be described by a linear discrete time invariant state space model given by

$$\begin{aligned} x_{k+1} &= Ax_k + Bu_k + v_k, & (2.11)\\ y_k &= Dx_k + Eu_k + w_k, & (2.12) \end{aligned}$$

where, $x_k \in \mathbb{R}^n$ is the state vector, $u_k \in \mathbb{R}^r$ is the input vector, $y_k \in \mathbb{R}^m$ is the output vector, v_k is the process noise and w_k is the measurement noise at discrete time k. Introducing $v_k = \tilde{C}\tilde{v_k}$ makes it easier to reconstruct the simulation

examples presented throughout the thesis.

Subspace-based identification methods identify and realize a dynamic state space model direct from measured data, including the system order. Therefore the methods are computationally efficient. The model identified by the subspace-based identification methods is the discrete time Kalman filter in innovation form

$$\begin{aligned} \bar{x}_{k+1} &= A\bar{x}_k + Bu_k + Ce_k, && (2.13)\\ y_k &= D\bar{x}_k + Eu_k + Fe_k, && (2.14) \end{aligned}$$

where, $e_k \in \mathbb{R}^m$ is the innovation with covariance matrix $E(e_k e_k^T) = I$ and $\bar{x}_k \in \mathbb{R}^n$ is the predicted state vector. A is the the state transition matrix, B is the the input matrix, C is the external input matrix, D is the output matrix, E is the direct input to output matrix and F is the direct external input to output matrix. The prediction of y_k is given by $\bar{y}_k = D\bar{x}_k + Eu_k$. An alternate expression is

$$\begin{aligned} \bar{x}_{k+1} &= A\bar{x}_k + Bu_k + K\varepsilon_k, && (2.15)\\ y_k &= D\bar{x}_k + Eu_k + \varepsilon_k, && (2.16) \end{aligned}$$

where, $\varepsilon_k \in \mathbb{R}^m$ is the innovation, $E(\varepsilon_k \varepsilon_k^T) = FF^T$ is the covariance matrix of the innovation and $K = CF^{-1}$ is the Kalman filter gain.

As an alternative to using state space presentation, the model can also be expressed by transfer functions. Introducing the forward shift operator, z, given by $zu_k = u_{k+1}$, the model can be presented by

$$y_k = H^d(z)u_k + H^s(z)\varepsilon_k, \tag{2.17}$$

where, $H^d(z)$ is the transfer function from the input, u_k, to the output, y_k, given by

$$H^d(z) = D(zI - A)^{-1}B + E, \tag{2.18}$$

and $H^s(z)$ is the transfer function from the innovation, ε_k, to the output, y_k, given by

$$H^s(z) = D(zI - A)^{-1}K + I. \tag{2.19}$$

In case of multiple input multiple output systems $H^d(z)$ and $H^s(z)$ are matrices of transfer functions.

Throughout the thesis the following notation will be used: the eigenvalues of the system matrix A, $\lambda(A)$, the eigenvalues of the Kalman filter $\lambda(A-KD)$, the deterministic transition zeros of the system $\rho(H^d(z))$ or $\rho(A,B,D,E)$, the deterministic steady state gain matrix $H^d(1)$ and the stochastic steady state gain matrix $H^s(1)$.

2.3 Quality criteria

A set of quality criteria has to be defined to evaluate the quality of the estimated model. Some basic properties have to be defined before the criteria can be defined. The squared Frobenius norm of a matrix $A \in \mathbb{R}^{m\times n}$ is equal to the trace of the product A^TA, and defined as follows

$$\|A\|_F^2 \stackrel{def}{=} \mathrm{tr}(A^TA) = \sum_{i=1}^{m}\sum_{j=1}^{n} a_{ij}^2. \tag{2.20}$$

For a complex number $c = a + \mathrm{j}\cdot b$ the following definitions are used

$$\mathrm{Re}(c) \stackrel{def}{=} a \text{ and } \mathrm{Im}(c) \stackrel{def}{=} b, \tag{2.21}$$

where j is the imaginary unit $j = \sqrt{-1}$.

A quadratic criterion on the eigenvalues of the estimated model is defined. The criterion, V^1 Equation (2.22), sums up the square of the deviation between the eigenvalues of the estimated model and the actual eigenvalues, the Squared Eigenvalue Error (SEE).

$$\begin{aligned} V^1 \stackrel{def}{=} & \tfrac{1}{M}\textstyle\sum_{i=1}^{M} \|\mathrm{Re}(\lambda(\overline{A}_i)) - \mathrm{Re}(\lambda(A))\|_F^2 + \\ & \tfrac{1}{M}\textstyle\sum_{i=1}^{M} \|\mathrm{Im}(\lambda(\overline{A}_i)) - \mathrm{Im}(\lambda(A))\|_F^2. \end{aligned} \tag{2.22}$$

The quality criterion (2.22) is presented using Matlab notation where $\lambda(A) \in \mathbb{R}^n$ is the vector of the eigenvalues of the true state transition matrix and $\lambda(\overline{A}_i) \in \mathbb{R}^n$ is the vector of the estimated eigenvalues of the state transition matrix at Monte Carlo run number i. M is the total number of Monte Carlo runs.

A quadratic criterion on the zeros of the estimated model is defined. The criterion, V^2 Equation (2.23), sums up the square of the deviation between the zeros of the estimated model and the actual zeros, the Squared Zero Error (SZE).

$$\begin{aligned} V^2 \stackrel{def}{=} & \tfrac{1}{M}\textstyle\sum_{i=1}^{M} \|\mathrm{Re}(\rho(\overline{A}_i,\overline{B}_i,\overline{D}_i,\overline{E}_i) - \mathrm{Re}(\rho(A,B,D,E))\|_F^2 + \\ & \tfrac{1}{M}\textstyle\sum_{i=1}^{M} |\mathrm{Im}(\rho(\overline{A}_i,\overline{B}_i,\overline{D}_i,\overline{E}_i)) - \mathrm{Im}(\rho(A,B,D,E))\|_F^2. \end{aligned} \tag{2.23}$$

The quality criterion (2.23) is presented using Matlab notation where $\rho(A, B, D, E)$ is the vector of the zeros of the true system and $\rho(\overline{A}_i, \overline{B}_i, \overline{D}_i, \overline{E}_i)$ is the vector of the estimated zeros of the estimated state space model at Monte Carlo run number i. M is the total number of Monte Carlo runs.

Chapter 3

Examples

All the examples that will be used throughout the thesis are presented in this chapter in order to make the thesis easier to read. If physical systems are presented the frequency is converted to unit-less frequency.

3.1 Example 1

A single input single output system introduced by Di Ruscio (2003b) given by

$$A = \begin{bmatrix} 0 & 1 \\ -0.7 & 1.5 \end{bmatrix}, B = \begin{bmatrix} 0.25 \\ 0.625 \end{bmatrix}, \tilde{C} = \begin{bmatrix} 0.5 \\ 0.5 \end{bmatrix}, D = \begin{bmatrix} 1 & 0 \end{bmatrix}, E = \begin{bmatrix} 0 \\ 0 \end{bmatrix} \quad (3.1)$$

and controlled by a PI-controller with $K_p = 2$ and $T_i = 5$ is used as an example. The PI-controller in discrete form is given by

$$u_k = K_p \epsilon_k + z_k, \quad (3.2)$$

$$z_{k+1} = z_k + \frac{K_p}{T_i} \epsilon_k, \quad (3.3)$$

where ϵ_k is given by

$$\epsilon_k = r_k - y_k. \quad (3.4)$$

The process noise variance used is $E(\tilde{v}_k \tilde{v}_k^T) = 0.01$ and the measuring noise variance used is $E(w_k^2) = 0.01$. Time series of N=1000 discrete data points, $k = 0, 1, \ldots, N-1$, are generated.

3.2 Example 2

In order to generate closed loop data, a model of a chemical reactor operating in closed loop is seleced. The reaction mechanism of the reactor is given by

$$A \xrightarrow{k_1} B \xrightarrow{k_2} C, \tag{3.5}$$

$$2A \xrightarrow{k_3} D. \tag{3.6}$$

The reaction from body A to body D is of order two, while the other reactions are of order 1. The body B is controlled by a PI-controller with $K_p = 50$ and $T_i = \frac{1}{75}$. The manipulated variable is the feed flow (flow rate) u $[\frac{1}{hours}]$. The concentration of body A in the feed flow, u, is θ. The concentration of body A and body B in the tank is respectively x_1 and x_2. The connection from u to $y = x_2$ is then given by the model

$$\dot{x}_1 = -k_1 x_1 - k_3 x_1^2 + (\theta - x_1)u, \tag{3.7}$$

$$\dot{x}_2 = k_1 x_1 - k_2 x_2^2 - x_2 u, \tag{3.8}$$

$$y = x_2, \tag{3.9}$$

where the reaction rate constants are given by $k_1 = 50$, $k_2 = 100$ and $k_3 = 10$. The stationary values of the states, disturbances, parameters and manipulated variable are given: $x_1^s = 2.5$, $x_2^s = 1$, $\theta^s = 10$ and $u^s = 25$. We assume that θ is constant and known.

Defining

$$\delta u_k \stackrel{def}{=} u_k - u^s, \tag{3.10}$$

$$\delta x_k \stackrel{def}{=} x_k - x^s, \tag{3.11}$$

$$\delta y_k \stackrel{def}{=} y_k - y^s, \tag{3.12}$$

$$\delta \theta_k \stackrel{def}{=} \theta_k - \theta^s, \tag{3.13}$$

a linearized discrete model can be expressed in the form

$$\delta x_{k+1} = A\delta x_k + B\delta u_k + C\delta\theta_k + v_k, \tag{3.14}$$

$$\delta y_k = D\delta x_k + E\delta u_k + w_k, \tag{3.15}$$

where in addition the additive process and measurement noise, v_k and w_k, are added respectively.

After linearization and discretization the system is expressed by

$$A = \begin{bmatrix} 0.8750 & 0 \\ 0.0500 & 0.8750 \end{bmatrix}, B = \begin{bmatrix} 0.0075 \\ -0.0010 \end{bmatrix}$$

$$C = \begin{bmatrix} 0.0250 \\ 0 \end{bmatrix}, D = \begin{bmatrix} 0 & 1 \end{bmatrix} \tag{3.16}$$

and the matrix E is the zero matrix. The linearization is done at the stationary point. The discretization is done by explicit Euler with uniform sampling interval $\delta t = 0.001$. A time series from 0 to 1, with N=1000 discrete data points, $k = 0, 1, \ldots, N-1$, is considered.

The linearized reactor model, (3.14) and (3.15), and the PI-controller, (3.2) and (3.3), are used in order to generate closed loop data. The reference will be constant $r_k = 1$ at each time instant k, superposed with a dithering signal. The process noise variance used is $E(v_k v_k^T) = 0.0001 \cdot I$ and the measuring noise variance used is $E(w_k^2) = 0.00001$. The reason for using a linearized discrete model of the chemical reactor is to have the ability to compare the estimated eigenvalues of the system, zeros of the system and the zeros of the Kalman filter to the respective values of the linearized discrete model.

The stationary values in both inputs and outputs will be removed prior to identification.

3.3 Example 3

A single input single output system introduced by Quin and Ljung (2003) is given by

$$y_k + a y_{k-1} = b u_{k-1} + e_k + c e_{k-1} \tag{3.17}$$

with a feedback controller

$$u_k = -K y_k + r_k \tag{3.18}$$

where, $a = -0.9$, $b = 1$, $c = 0.9$ and $K = 0.6$, is used as an example. The standard deviation for e_k is one and the standard deviation for r_k is two. Both of the signals are Gaussian white noise.

The discrete process model can be transformed to a state space model Equations (2.11) - (2.12) where $A = 0.9$, $B = 1$, $C = 1.8$, $D = 1$, $E = 0$ and $F = 1$. In the simulation time series of N=2000 discrete data points are collected and 20 Monte Carlo simulations are performed.

3.4 Example 4

A single input single output system introduced by Chiuso and Picci (2004) is described in Figure 3.1

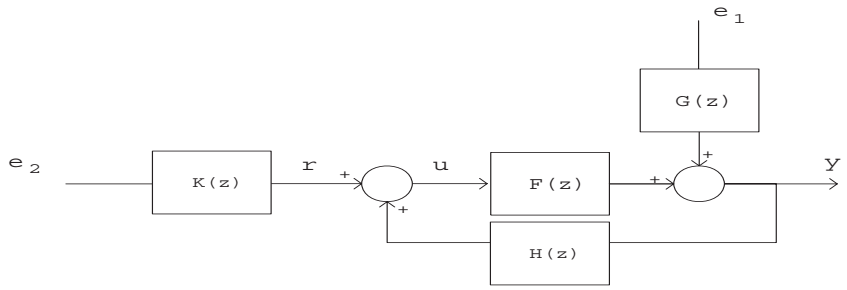

Figure 3.1: System operating in closed loop in Example 4.

where

$$F(z) = \frac{b}{z-a},\ G(z) = \frac{z+c}{z},\ H(z) = -1, \tag{3.19}$$

and $a = 3$, $b = 2.5$, $c = 0.999$ which is an unstable system.

Using $F(z)$ and $G(z)$ to describe the system, the noise model may lead the reader to consider this as a first order system, which is false. Expressing $F(z)$ and $G(z)$ as an augmented state space model, Equations (2.11) - (2.12), where the matrices are

$$A = \begin{bmatrix} a & 1 \\ 0 & 0 \end{bmatrix}, B = \begin{bmatrix} b \\ 0 \end{bmatrix}, C = \begin{bmatrix} c \\ -ac \end{bmatrix}, D = \begin{bmatrix} 1 & 0 \end{bmatrix}, \tag{3.20}$$

where in addition E is the zero matrix and F is the identity matrix. This is a second order system.

The reference r used is generated from the transfer function $K(z) = \frac{0.2(z+0.999)}{z-0.99}$ driven by a white noise input, unless something else is specified. When transforming to state space model, Equations (2.11) and (2.12), the transfer function can be expressed by

$$A_r = 0.99,\ B_r = 1,\ C_r = 0,\ D_r = 0.3978,\ E_r = 0.2,\ F_r = 0, \tag{3.21}$$

where subscript r indicates the system matrices used to generate the reference signal from a white noise input.

3.5 Example 5

A multiple input multiple output system is given by

$$A = \begin{bmatrix} 1.5 & 1.0 & 0.1 \\ -0.7 & 0.0 & 0.1 \\ 0 & 0 & 0.85 \end{bmatrix}, B = \begin{bmatrix} 0 & 0 \\ 0 & 1 \\ 1 & 0 \end{bmatrix}, \tilde{C} = \begin{bmatrix} 0.0 & 0.1 \\ 0.1 & 0 \\ 0 & 0.2 \end{bmatrix},$$

$$D = \begin{bmatrix} 3 & 0 & -0.6 \\ 0 & 1 & 1 \end{bmatrix}, E = \begin{bmatrix} 0 & 0 \\ 0 & 0 \end{bmatrix}. \tag{3.22}$$

Input 1 is used to control output 1 using a PI-controller with $K_p = 0.02$ and $T_i = 2$. Input 2 is used to control output 2 using a PI-controller with $K_p = -0.02$ and $T_i = 2$. The PI-controllers used to control the outputs are given by Equations (3.2) and (3.3). The process noise variance used is $E(\tilde{v}_k \tilde{v}_k^T) = \begin{bmatrix} 0.001 & 0 \\ 0 & 0.001 \end{bmatrix}$ and the measuring noise variance used is $E(w_k w_k^T) = \begin{bmatrix} 0.001 & 0 \\ 0 & 0.0005 \end{bmatrix}$. Time series are generated of N=1000 discrete data points, $k = 0, 1, \ldots, N-1$.

Chapter 4

Classic Subspace Identification

Before the specific projections used in subspace identifications methods are presented some frequently used notation and definitions have to be presented.

Definition 4.1 (Hankel matrix) *Given a (vector or matrix) sequence of data*

$$s_t \in \mathbb{R}^{nr \times nc} \ \forall \ t = 0, 1, 2, \ldots, t_0, t_0 + 1, \ldots, \tag{4.1}$$

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, i and K and define the matrix S_t as follows

$$S_{t_o|i} \overset{def}{=} \begin{bmatrix} s_{t0} & s_{t0+1} & s_{t0+2} & \cdots & s_{t0+K-1} \\ s_{t0+1} & s_{t0+2} & s_{t0+3} & \cdots & s_{t0+K} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ s_{t0+i-1} & s_{t0+i} & s_{t0+i+1} & \cdots & s_{t0+i+K-2} \end{bmatrix} \in \mathbb{R}^{inr \times Knc} \tag{4.2}$$

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

- *t_0 start index or initial time in the sequence s_{t0} which is the upper-left block in the Hankel matrix.*
- *i is the number of nr-block rows in $S_{t_0|i}$.*
- *K is the number of nc-block columns in $S_{t_0|i}$.*

The elements, are constant across the anti-diagonals in the Hankel matrix and are symmetrical along the diagonal. The Hankel matix is not symmetrical when it is rectangular. Vector sequences are used in subspace system identification. Then s_t is a vector nc=1. Consider the measured process outputs, $y_t \in \mathbb{R}^m$, and possibly known inputs, $u_t \in \mathbb{R}^r$. Given a number i known output vectors and a number i known input vectors the extended output vector Equation (4.3) and the extended output vector Equation (4.4) can be made

$$y_{k|i} \stackrel{def}{=} \begin{bmatrix} y_k \\ y_{k+1} \\ \vdots \\ y_{k+i-1} \end{bmatrix} \in \mathbb{R}^{im} \tag{4.3}$$

$$u_{k|i} \stackrel{def}{=} \begin{bmatrix} u_k \\ u_{k+1} \\ \vdots \\ u_{k+i-1} \end{bmatrix} \in \mathbb{R}^{im}. \tag{4.4}$$

In similar ways extended data matrices can be defined. Define the following output data matrix with i block rows and K columns.

$$Y_{k|i} \stackrel{def}{=} \begin{bmatrix} y_k & y_{k+1} & y_{k+2} & \cdots & y_{k+K-1} \\ y_{k+1} & y_{k+2} & y_{k+3} & \cdots & y_{k+K} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ y_{k+i-1} & y_{k+i} & y_{k+i+1} & \cdots & y_{k+i+K-2} \end{bmatrix} \in \mathbb{R}^{im \times K} \tag{4.5}$$

Define the following input data matrix with i block rows and K columns.

$$U_{k|i} \stackrel{def}{=} \begin{bmatrix} u_k & u_{k+1} & u_{k+2} & \cdots & u_{k+K-1} \\ u_{k+1} & u_{k+2} & u_{k+3} & \cdots & u_{k+K} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ u_{k+i-1} & u_{k+i} & u_{k+i+1} & \cdots & u_{k+i+K-2} \end{bmatrix} \in \mathbb{R}^{ir \times K} \tag{4.6}$$

The extended data matrix $Y_{k|i}$ is a known data matrix of output variables. The extended data matrix $U_{k|i}$ is a known data matrix of input variables.

Associated with the state space model Equations (2.11) and (2.12), the following definitions are introduced

- The extended observability matrix (O_i) for the pair (D, A) is defined as

$$O_i \stackrel{def}{=} \begin{bmatrix} D \\ DA \\ \vdots \\ DA^{i-1} \end{bmatrix} \in \mathbb{R}^{im \times n} \tag{4.7}$$

  where the subscript i denotes the number of block rows.

- The reverse extended controllability matrix C_i^d for the pair (A, B) is defined as

$$C_i^d \stackrel{def}{=} [A^{i-1}B \quad A^{i-2}B \quad \dots \quad B] \in \mathbb{R}^{n \times ir} \tag{4.8}$$

  where the subscript i denotes the number of block columns.

- The reverse extended controllability matrix C_i^s for the pair (A, C) is defined as

$$C_i^s \stackrel{def}{=} [A^{i-1}C \quad A^{i-2}C \quad \dots \quad C] \in \mathbb{R}^{n \times il} \tag{4.9}$$

  where the subscript i denotes the number of block columns.

- The lower block triangular Toeplitz matrix (H_i^d) for the quadruple matrices (D, A, B, E) is defined as

$$H_i^d \stackrel{def}{=} \begin{bmatrix} E & 0 & 0 & \dots & 0 \\ DB & E & 0 & \dots & 0 \\ DAB & DB & E & \dots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ DA^{i-2}B & DA^{i-3}B & DA^{i-4}B & \dots & E \end{bmatrix} \in \mathbb{R}^{im \times (i-1)r} \tag{4.10}$$

  where the subscript i denotes the number of block rows and $i-1$ is the number of block columns.

- The lower block triangular Toeplitz matrix (H_i^s) for the quadruple matrices (D, A, C, F) is defined as

$$H_i^s \stackrel{def}{=} \begin{bmatrix} F & 0 & 0 & \dots & 0 \\ DC & F & 0 & \dots & 0 \\ DAC & DC & F & \dots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ DA^{i-2}C & DA^{i-3}C & DA^{i-4}C & \dots & F \end{bmatrix} \in \mathbb{R}^{im \times im} \tag{4.11}$$

  where the subscript i denotes the number of block rows and $i-1$ is the number of block columns.

Estimation of the extended observability matrix O_{L+1} and the system order n from the column space of a known data matrix is the first step in subspace identification algorithms. The idea is to estimate O_{L+1} by computing a special projection matrix from known data. When O_{L+1} is estimated the system matrix A can be found using the shift invariant property, Di Ruscio (1995).

4.1 DSR

In this section the basic equations in DSR will be presented. The subspace system identification of the Kalman filter performed in DSR will not be treated. The theory can be found in Di Ruscio (2003a) and Di Ruscio (2004).

4.1.1 The parameters to be chosen in DSR

In DSR there are four parameters g, n, L and J that can be chosen by the user. If the structure parameter g is 1, which is the default in DSR, the data matrix E in the state space model is identified. If g is set to zero, the matrix E is forced to be the zero matrix. Parameter n specifies the model order. Parameter L is the number of block rows in the extended observability matrix. L can be interpreted as the identification horizon used to predict the number of states. This again limits the maximal system order which can be identified. The order must be chosen in the interval $1 \leq n \leq L \cdot m$, where m is the number of outputs. Parameter J is the number of time instants in the past horizon which is used for defining the instrument variable matrix which are used to remove noise.

The experience so far in open loop cases is that the parameter L should be chosen to be as small as possible in order to reduce the variance of the estimates. This is especially important in case of poorly excitating input signals. Choosing a large numeric value for the parameter J improves the estimates of the Kalman filter gain. The parameter J is usually chosen as $J = L$ or $J = L + 1$. It is normally not crucial which of these two alternatives are chosen.

4.1.2 The Extended State Space Model, elimination of the states

The state space model Equations (2.11)-(2.12) contains the states. By using the fact that the Extended Output Matrix Equation can be written as follows:

$$Y_{k|L} = O_L X_k + H_L^d U_{k|L+g-1} + H_L^s E_{k|L} \tag{4.12}$$

where

$$X_k \stackrel{def}{=} [x_k \quad x_{k+1} \quad \dots \quad x_{k+K-1}] \quad \in \mathbb{R}^{n \times K} \tag{4.13}$$

it is possible to get an expression where the states are eliminated. By replacing the index k by $k+1$ in Equation (4.12) and using the equations in the state space model Equations (2.13)-(2.14) and finding an expression for X_k from Equation (4.12) we get the Extended State Space Model (ESSM). The ESSM is given by

$$Y_{k+1|L} = \tilde{A}_L Y_{k|L} + \tilde{B}_L U_{k|L+g} + \tilde{C}_L E_{k|L+1} \tag{4.14}$$

where

$$\tilde{A}_L \stackrel{def}{=} O_L A (O_L^T O_L)^{-1} O_L^T = O_L A O_L^\dagger \in \mathbb{R}^{Lm \times Lm} \tag{4.15}$$

$$\tilde{B}_L \stackrel{def}{=} [O_L B \quad H_L^d] - \tilde{A}_L [H_L^d \quad 0_{Lm \times r}] \in \mathbb{R}^{Lm \times (L+g)r} \tag{4.16}$$

$$\tilde{C}_L \stackrel{def}{=} [O_L C \quad H_L^s] - \tilde{A}_L [H_L^s \quad 0_{Lm \times l}] \in \mathbb{R}^{Lm \times (L+g)m}. \tag{4.17}$$

Note that when the states are eliminated the number of unknowns is reduced.

4.1.3 Relationship between past and future data matrices

By using the Extended Output Matrix Equation (4.12) with index k replaced by index J and using the following relation between future states X_J and historic states X_0

$$X_J = A^J X_0 + C_J^d U_{0|J} + C_J^s E_{0|J} \tag{4.18}$$

it is possible to eliminate X_J. X_0 can then be eliminated by rewriting the Extended Output Matrix Equation (4.12) with index k replaced by index 0. By using these properties we get

$$Y_{J|L+1} = [H_{L+1}^d \;\; P_{L+1}^d \;\; \tilde{A}_{L+1}^J] \begin{bmatrix} U_{J|L+1} \\ U_{0|J} \\ Y_{0|J} \end{bmatrix} + [H_{L+1}^s \;\; P_{L+1}^s] \begin{bmatrix} E_{J|L+1} \\ E_{0|J} \end{bmatrix} \tag{4.19}$$

which shows the relationship between past and future data matrices where

$$\tilde{A}_{L+1}^J \stackrel{def}{=} O_{L+1} A^J O_J^\dagger \tag{4.20}$$

$$P_{L+1}^d \stackrel{def}{=} O_{L+1} (C_J^d - A^J O_J^\dagger H_J^d) \tag{4.21}$$

$$P_{L+1}^s \stackrel{def}{=} O_{L+1} (C_J^s - A^J O_J^\dagger H_J^s). \tag{4.22}$$

4.1.4 The projections used in DSR

There are two ways to explain the choice of projections in DSR. One is by considering the Extended State Space Model Equation (4.14). The other is presented here, considering the relationship between past and future data matrices Equation (4.19).

The terms defined by the future inputs and future noise, $H_{L+1}^d U_{J|L+1}$ and $H_{L+1}^s E_{J|L+1}$ respectively, do not have the same column space as the extended observability matrix. Therefore, these terms have to be removed. To do this future outputs are projected onto the matrix of instrumental variables to remove the effect of

future noise, $E_{J|L+1}$. The instrumental variables have to be uncorrelated with the future noise $E_{J|L+1}$ and sufficiently correlated with the informative part in the ESSM in order not to destroy information about the system order, for instance. An intuitive choice of instrumental variables are past data (past inputs and outputs). This choice ensures that the instruments are sufficiently correlated with the informative part of the signals and sufficiently uncorrelated with future noise. If only past inputs are used as instruments only the deterministic part of the model can be identified, Verhaegen (1994). By incorporating past outputs as instruments the stochastic part of the model can be identified too. The instrumental variables used in DSR are

$$\begin{bmatrix} U_{J|L+g} \\ U_{0|J} \\ Y_{0|J} \end{bmatrix}. \tag{4.23}$$

Projecting Equation (4.19) onto Equation (4.23) gives

$$\begin{aligned} Y_{J|L+1}/\begin{bmatrix} U_{J|L+g} \\ U_{0|J} \\ Y_{0|J} \end{bmatrix} &= [H^d_{L+1} \;\; P^d_{L+1} \;\; \tilde{A}^J_{L+1}] \begin{bmatrix} U_{J|L+g} \\ U_{0|J} \\ Y_{0|J} \end{bmatrix} \\ &+ P^s_{L+1} E_{0|J} \begin{bmatrix} U_{J|L+g} \\ U_{0|J} \\ Y_{0|J} \end{bmatrix} + dE_1 \end{aligned} \tag{4.24}$$

where

$$dE_1 \stackrel{def}{=} H^s_{L+1} E_{J|L+1}/\begin{bmatrix} U_{J|L+g} \\ U_{0|J} \\ Y_{0|J} \end{bmatrix}. \tag{4.25}$$

The error term Equation (4.25) turns to zero when K turns to infinity. Rearranging Equation (4.24) and get

$$\begin{aligned} Y_{J|L+1}/\begin{bmatrix} U_{J|L+g} \\ U_{0|J} \\ Y_{0|J} \end{bmatrix} &= [P^d_{L+1} \;\; \tilde{A}^J_{L+1} \;\; P^s_{L+1}] \begin{bmatrix} U_{0|J} \\ Y_{0|J} \\ Y_{0|J}/\begin{bmatrix} U_{J|L+g} \\ U_{0|J} \\ Y_{0|J} \end{bmatrix} \end{bmatrix} \\ &+ H^d_{L+1} U_{J|L+1} + dE_1. \end{aligned} \tag{4.26}$$

By using the projection

$$U^{\perp}_{k|L+g} = I_{K\times K} - U^T_{k|L+g}(U_{k|L+g}U^T_{k|L+g})^{\dagger}U_{k|L+g} \tag{4.27}$$

the effect of future inputs $U_{J|L+g}$ is removed and

$$(Y_{J|L+1}/\begin{bmatrix} U_{J|L+g} \\ U_{0|J} \\ Y_{0|J} \end{bmatrix})U^{\perp}_{J|L+g} = [P^d_{L+1}\ \tilde{A}^J_{L+1}\ P^s_{L+1}] \begin{bmatrix} \begin{matrix} U_{0|J} \\ Y_{0|J} \end{matrix} \\ Y_{0|J}/\begin{bmatrix} U_{J|L+g} \\ U_{0|J} \\ Y_{0|J} \end{bmatrix} \end{bmatrix} U^{\perp}_{J|L+g} + dE_2 \quad (4.28)$$

where

$$dE_2 \stackrel{def}{=} dE_1 U^{\perp}_{J|L+g}. \quad (4.29)$$

The error term Equation (4.29) turns to zero when K turns to infinity just like Equation (4.25) if $U_{J|L+g}$ is uncorrelated with $E_{J|L+1}$.

4.1.5 The extended observability matrix

As mentioned in Chapter 4 the estimation of the extended observability matrix O_{L+1} and the system order n from the column space of a known data matrix are the first step in subspace identification algorithms. The idea is to estimate O_{L+1} by computing a special projection matrix from known data. When O_{L+1} is estimated the system matrix A can be found using the shift invariant property, Di Ruscio (1995). Considering Equation (4.28) together with Equations (4.20)-(4.22) it is clear that the left-hand side of Equation (4.28) is proportional to the extended observability matrix. Then left-hand side of Equation (4.28) is such a special projection matrix. Defining the left-hand side of Equation (4.28) gives

$$Z_{J|L+1} \stackrel{def}{=} (Y_{J|L+1}/\begin{bmatrix} U_{J|L+g} \\ U_{0|J} \\ Y_{0|J} \end{bmatrix})U^{\perp}_{J|L+g}. \quad (4.30)$$

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

$$Z_{J|L+1} = O_{L+1} X^a_J \quad (4.31)$$

where

$$X^a_J \stackrel{def}{=} [A^J O^{\dagger}_J\ C^d_J - A^J O^{\dagger}_J H^d_J\ C^s_J - A^J O^{\dagger}_J H^s_J] \begin{bmatrix} \begin{matrix} Y_{0|J} \\ U_{0|J} \end{matrix} \\ E_{0|J}/\begin{bmatrix} U_{J|L+g} \\ U_{0|J} \\ Y_{0|J} \end{bmatrix} \end{bmatrix} U^{\perp}_{J|L+g} \quad (4.32)$$

The column space of the matrix $Z_{J|L+1}$ coincides with the column space of the extended observability matrix O_{L+1}. The system order n of the state space model is given as the dimension of the column space. X_J^a can be interpreted as a state sequence of an autonomous system. The rank of $Z_{J|L+1}$ is equal to the rank of X_J^a when the pair (A, D) is observable.

An alternative expression of X_J^a is

$$X_J^a = [A^J \quad C_J^d \quad C_J^s] \begin{bmatrix} X_0 / \begin{bmatrix} U_{J|L+g} \\ U_{0|J} \\ Y_{0|J} \end{bmatrix} \\ U_{0|J} \\ E_{0|J} / \begin{bmatrix} U_{J|L+g} \\ U_{0|J} \\ Y_{0|J} \end{bmatrix} \end{bmatrix} U_{J|L+g}^{\perp}. \tag{4.33}$$

The expression relates X_J^a to the reverse extended controllability matrix C_J^d for the pair (A, B) and the reverse extended controllability matrix C_J^s for the pair (A, C). It is stated that $rank(X_J^a) = n$ if

- the pair $(A, [B, C])$ is controllable,
- the state X_0 is sufficiently correlated with the instruments (past inputs and outputs) $U_{0|J}$ and $Y_{0|J}$,
- the input is persistently exciting with a sufficient high order,
- the past noise $E_{0|J}$ is sufficient correlated with the instruments (past inputs and outputs) $U_{0|J}$ and $Y_{0|J}$.

In the DSR software the system matrices are computed using QR-decomposition. The principal methods to estimate the system matrices are presented in the next section.

4.1.6 The basic steps in the DSR algorithm

1. **Calculate the orthogonal projections**

$$Z_{J|L} = (Y_{J|L} / \begin{bmatrix} U_{J|L+g} \\ U_{0|J} \\ Y_{0|J} \end{bmatrix}) U_{J|L+g}^{\perp} \tag{4.34}$$

and

$$Z_{J+1|L} = (Y_{J+1|L} / \begin{bmatrix} U_{J|L+g} \\ U_{0|J} \\ Y_{0|J} \end{bmatrix}) U^{\perp}_{J|L+g} \tag{4.35}$$

satisfying the autonomous matrix equation

$$Z_{J+1|L} = \tilde{A}_L Z_{J|L} \tag{4.36}$$

where the positive integer parameters L and J are given.

2. **Find the system order,** n

   Compute the Singular Value Decomposition (SVD)

$$Z_{J|L} = USV^T \tag{4.37}$$

   where, $U \in \mathbb{R}^{Lm \times Lm}$, $S \in \mathbb{R}^{Lm \times (m+r)}$ and $V \in \mathbb{R}^{(m+r) \times (m+r)}$ are given by

$$U = [U_1 \quad U_2] \quad S = \begin{bmatrix} S_1 & 0 \\ 0 & S_2 \end{bmatrix} \quad V = [V_1 \quad V_2] \tag{4.38}$$

   where, $S_n \in \mathbb{R}^{n \times n}$ and n is the number is the number of "non-zero" singular values of $Z_{J|L}$, which is equal to the system order. The system order, n, is determined by inspection of the diagonal elements of S or SS^T and is equal to the number of "non-zero" elements. The term $U_2 S_2 V_2$ represents the error by estimating the system order as the n first principal singular values.

3. **The extended observability matrix** O_L **for the pair (D,A)**

   The (extended) observability matrix can be taken directly as the first left-hand part in U, i.e U_1. Then we have

$$O_L = U(1 : Lm, 1 : n) = U_1. \tag{4.39}$$

4. **The system matrix A**

   The system matrix A can be computed from

$$A = O_L^T Z_{J+1|L} V \begin{bmatrix} S_n^{-1} \\ 0 \end{bmatrix} = U_1^T Z_{J+1|L} V_1 S_n^{-1}. \tag{4.40}$$

5. **The system output matrix D**
   The matrix D can be computed as the $m \times n$ upper sub-matrix in the observability matrix O_L like

$$D = U(1:m, 1:n). \tag{4.41}$$

6. **The extended system matrix $\tilde{A}_L$**
   This is computed from

$$\tilde{A}_L = O_L A (O_L^T O_L)^{-1} O_L^T = Z_{J+1|L} V_1 S_n^{-1} U_1^T. \tag{4.42}$$

7. **The system matrices B and E**
   Define

$$Z^d_{J|L+1} \stackrel{def}{=} Y_{J+1|L} / \begin{bmatrix} U_{J|L+g} \\ U_{0|J} \\ Y_{0|J} \end{bmatrix}. \tag{4.43}$$

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

$$Z^d_{J+1|L} = \tilde{A}_L Z^d_{J|L} + \tilde{B}_L U_{J|L+g}. \tag{4.44}$$

   Define from the linear Equation (4.44)

$$\mathcal{Y} \stackrel{def}{=} \tilde{B}_L \mathcal{U} \tag{4.45}$$

   where

$$\mathcal{Y} \stackrel{def}{=} Z^d_{J+1|L} - \tilde{A}_L Z^d_{J|L} \tag{4.46}$$

$$\mathcal{U} \stackrel{def}{=} U_{J|L+g}. \tag{4.47}$$

   Solve the least squares problem

$$\min_{B,E} \|\mathcal{Y} - \tilde{B}_L(B,E)\mathcal{U}\|_F^2. \tag{4.48}$$

8. **Compute the Kalman filter gain**
   The Kalman filter gain is not found by solving a Riccati equation, but by projections. The theory can be found in Di Ruscio (2003a) and Di Ruscio (2004).

Note that it may be difficult to estimate the system order as the non-zero or large singular values of the matrix $Z_{J|L}$ when the signal-to-noise ratio is "small". Very often both physical knowledge about the process and validation have to be used to find the correct model order.

4.2 N4SID

The basic steps in the robust version of N4SID, Van Overschee and De Moor (1996) are presented here. The notation introduced by Di Ruscio (1995)-(2004) will be used. Note that compared to DSR, it is not possible to choose the past horizon to be different from the future horizon. Comparing the notation used in the description of DSR and N4SID $i = L + 1$, this because N4SID does not have the structure parameter g like DSR. Hence the smallest i to be chosen in N4SID to identify a first order system is 2, unlike DSR where smallest value of the parameter L to be used for a first order system is 1.

1. **Calculate the oblique and orthogonal projections**

$$\mathcal{O}_{L+1} = Y_{L+1|L+1}/_{U_{L+1|L+1}} \begin{bmatrix} U_{0|L+1} \\ Y_{0|L+1} \end{bmatrix}, \tag{4.49}$$

$$Z_{L+1|L+1} = Y_{L+1|L+1}/ \begin{bmatrix} U_{0|L+1} \\ Y_{0|L+1} \\ U_{L+1|L+1} \end{bmatrix}, \tag{4.50}$$

$$Z_{L+2|L} = Y_{L+2|L}/ \begin{bmatrix} U_{0|L+2} \\ Y_{0|L+2} \\ U_{L+2|L} \end{bmatrix}. \tag{4.51}$$

2. **Calculate the SVD of the so-called weighted oblique projection**
   which actually is one oblique projection and one orthogonal projection

$$\mathcal{O}_{L+1} U^{\perp}_{L+1|L+1} = USV^T. \tag{4.52}$$

   The projection $U^{\perp}_{L+1|L+1}$ is not included in the original version of N4SID, but is added later in the robust version to remove the effect of future inputs on future outputs.

3. **Determine the system order**
   by inspecting the singular values in S and partition the SVD to obtain U_1 and S_1.

4. **Calculate O_{L+1} and O_L**

$$\begin{aligned} O_{L+1} &= U_1 S_1^{\frac{1}{2}}, && (4.53)\\ O_L &= O_{L+1}(1:Lm,:). && (4.54) \end{aligned}$$

5. **Solve the set of linear equations for A and D**

$$\begin{bmatrix} O_L^\dagger Z_{L+2} \\ Y_{L+1|1} \end{bmatrix} = \begin{bmatrix} A \\ D \end{bmatrix} O_{L+1}^\dagger Z_{L+1} - \mathcal{K} U_{L+1|L+1} + \begin{bmatrix} \rho_w \\ \rho_v \end{bmatrix} \tag{4.55}$$

where

$$\mathcal{K} \stackrel{def}{=} \begin{bmatrix} \begin{bmatrix} B & O_L^\dagger H_L^d \end{bmatrix} & -AO_{L+1}^\dagger H_{L+1}^d \\ [E \quad 0] & -DO_{L+1}^\dagger H_{L+1}^d \end{bmatrix}. \tag{4.56}$$

6. **Recompute O_{L+1} and O_L from A and D**

7. **Solve B and E from**

$$B, E = \arg\min \| \begin{bmatrix} O_L^\dagger Z_{L+2} \\ Y_{L+1|1} \end{bmatrix} - \begin{bmatrix} A \\ D \end{bmatrix} O_L^\dagger Z_{L+1} - \mathcal{K}(B,E) U_{L+1|L+1} \|_R^2 \tag{4.57}$$

8. **Compute the covariance matrices Q, S and R as**

$$\begin{bmatrix} Q & S \\ S^T & R \end{bmatrix} = E_j(\begin{bmatrix} \rho_w \\ \rho_v \end{bmatrix} \begin{bmatrix} \rho_w^T & \rho_v^T \end{bmatrix}) \tag{4.58}$$

where by definition

$$E_J(\bullet) \stackrel{def}{=} \lim_{j \to \infty} (\bullet). \tag{4.59}$$

9. **Compute the Kalman filter gain**
   A Riccati equation is solved to compute the Kalman filter gain.

A more detailed description of the N4SID algorithm can be found in Van Overschee and De Moor (1996). The implementation of N4SID algorithm is based on

QR-decomposition just like DSR.

The main differences between the robust N4SID algorithm and DSR is that the computation of the extended observability matrix is done by one oblique projection and one orthogonal projection in N4SID but in DSR only orthogonal projections are used. In N4SID, the Kalman filter gain is found by solving a Riccati equation. In DSR, projections are used in such a way that there is no need for solving Riccati equations to compute the Kalman filter gain, Di Ruscio (2003a).

The implementation of the robust N4SID done in the Matlab script **subid.m**, Van Overschee and De Moor (1996), differs from the N4SID implemented in Matlab 6.5. The implementations in Matlab 6.5 are an improvement. Therefore when methods are compared, the N4SID implementation in Matlab 6.5 will be used, even though it is not clear which modifications have been done. The only exception is in Section 4.4 where the different projections are compared. There the implementation of the robust N4SID, **subid.m**, is used, Van Overschee and De Moor (1996).

4.3 Error-in-variable based subspace identification

Subspace identification methods have also been presented that are supposed to work on systems operating in open loop and in addition handle error-in-variable (EIV) problems and systems operating in a closed loop. Here these kinds of methods will be referred to as EIV-based subspace identification methods. The reason for this name is that the EIV property is one of the main goals when the projections are chosen. The only one of these methods which is presented here is the Subspace based Identification using instrumental Variables (SIV) by Gustafsson (2001). The paper only handles the estimation of the extended observability matrix. SIV is an improvement of the MOESP variant proposed by Chou and Verhaegen (1997) which is supposed to be applicable to a closed loop and error-in-variable identification in addition to the open loop case.

In subspace identification, $[U_{0|J}^T \; Y_{0|J}^T \; U_{J|L+1}^T]^T$ is considered a good choice of instrumental variable matrix. But both Chou and Verhaegen (1997) and Gustafsson (2001) choose to use $[U_{0|J}^T \; Y_{0|J}^T]^T$. The argument by Gustafsson (2001) is that when the future inputs $U_{J|L+1}^T$ are included in the instrumental variable matrix the influence of noise does not vanish when the number of samples tend towards infinity in the EIV case. The reason for this is that the noise on the input is correlated to the the other noise terms on the states and the outputs. This

choice of instrumental variable is applicable to systems operating in a closed loop, Chou and Verhaegen (1997), assuming that the controller is causal, that there is at least a delay in the controller and that the closed loop is asymptotically stable.

The following presentation of the SIV estimate of the extended observability matrix is done using the notation in Di Ruscio (1995)-(2004). The projection is given by

$$Z_{J|L+1} = Y_{J|L+1} \left[\begin{array}{c} U_{0|J} \\ Y_{0|J} \end{array} \right]^T \left[U_{J|L+1} \left[\begin{array}{c} U_{0|J} \\ Y_{0|J} \end{array} \right]^T \right]^{\perp} W_R \tag{4.60}$$

where W_R is a weighting matrix which has to have full rank. It is also shown that in the SIV framework the weighting matrix W_R can be removed and replaced by a "pre-whitening" of the instrumental variable matrix. This means that the matrix $[U_{0|J}^T \ Y_{0|J}^T]^T$ has to be replaced by

$$\left(\left[\begin{array}{c} U_{0|J} \\ Y_{0|J} \end{array} \right] \left[\begin{array}{c} U_{0|J} \\ Y_{0|J} \end{array} \right]^T \right)^{-\frac{1}{2}} \left[\begin{array}{c} U_{0|J} \\ Y_{0|J} \end{array} \right]. \tag{4.61}$$

This "pre-whitening" of the instrumental variables is the difference between the algorithm by Chou and Verhaegen (1997) and Gustafsson (2001).

When SIV is used in the succeeding sections an implementation of the following steps is used

1. **Calculate the orthogonal projection**

$$Z_{J|L+1} = Y_{J|L+1} \left[R_{pp}^{-\frac{1}{2}} \left[\begin{array}{c} U_{0|J} \\ Y_{0|J} \end{array} \right] \right]^T \left[U_{J|L+1} \left[R_{pp}^{-\frac{1}{2}} \left[\begin{array}{c} U_{0|J} \\ Y_{0|J} \end{array} \right] \right]^T \right]^{\perp} \tag{4.62}$$

   where

$$R_{pp} \stackrel{def}{=} \left[\begin{array}{c} U_{0|J} \\ Y_{0|J} \end{array} \right] \left[\begin{array}{c} U_{0|J} \\ Y_{0|J} \end{array} \right]^T. \tag{4.63}$$

2. **Calculate the SVD of Z_{L+1}**

$$Z_{L+1} = USV^T. \tag{4.64}$$

3. **Determine the system order** by inspecting the singular values in S and partition the SVD accordingly to obtain U_1 and S_1

4. **Calculate** O_{L+1}

$$O_{L+1} = U_1. \tag{4.65}$$

5. **The system matrix A**
The system matrix A can be computed from

$$A = O_{L+1}(1 : Lm, :)^{\dagger} O_{L+1}(m + 1 : (L + 1)m, :) \tag{4.66}$$

using the shift invariance properties of O_{L+1}.

4.4 Test of Projections

In this section the quality of the projections used in DSR, N4SID and SIV will be compared to the estimates found by using the Prediction Error Method (PEM) implemented in Matlab 6.5. To do this, there will be a comparison of the ability to estimate the eigenvalues of the system presented in Section 3.1 from data from Monte Carlo simulations. All the projections are implemented by using the definitions. The state transition matrix A is found by using the shift invariance properties of the extended observability matrix. The system order $n = 2$ and the fact that the matrix E is the zero matrix is assumed known. PEM is used with default parameters and $nk = 1$. The horizons chosen in the rest of the methods are those that minimize the Squared Eigenvalue Error criterion, V^1, in each case.

A (Pseudo Random Binary Signal) PRBS of length N is constant for a random interval of T samples. The random interval T is bounded by a specified band, $T_{min} <= T <= T_{max}$. With "high frequent" PRBS it means that the signal changes rapidly. With "low frequent" PRBS it means that the signal changes slowly, here $T_{min} = 250$ and $T_{max} = 300$ are chosen. To illustrate the noise level the "low frequent" PRBS signal used as input, u_k^1 is plotted in Figure 4.1 with the corresponding output, y_k, for one particular noise realization.

DSR is used with $g = 0$, $L = 5$ and $J = 6$ to estimate the eigenvalues when u_k^1 is used as input on the system operating in an open loop . N4SID is used with a horizon that corresponds to $L = 5$ and $g = 0$. SIV is used with horizons that corresponds to $L = 8$, $J = 9$ and $g = 0$.

The eigenvalue estimates shown in Figure 4.2 indicate that there is no reason to state that the projections used in N4SID are superior to the ones used in DSR. Therefore, so far in the open loop case, there is no argument to use the oblique projection used in N4SID instead of the orthogonal projection used in DSR. SIV gives unbiased estimates like N4SID and DSR with open loop data, but there is marginally larger variance. The crucial test is the test on closed loop data.

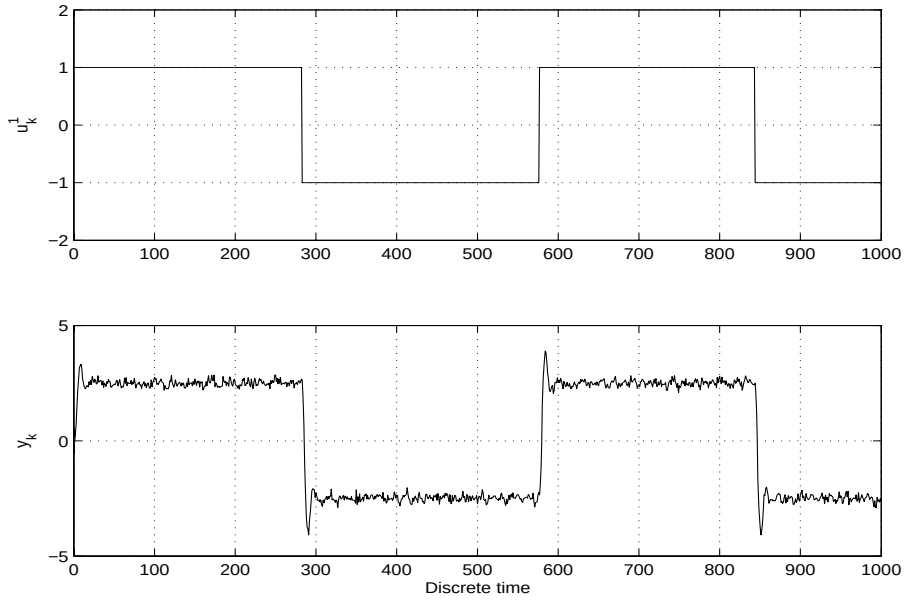

Figure 4.1: Input u_k^1 and corresponding output y_k for Example 1, Section 3.1.

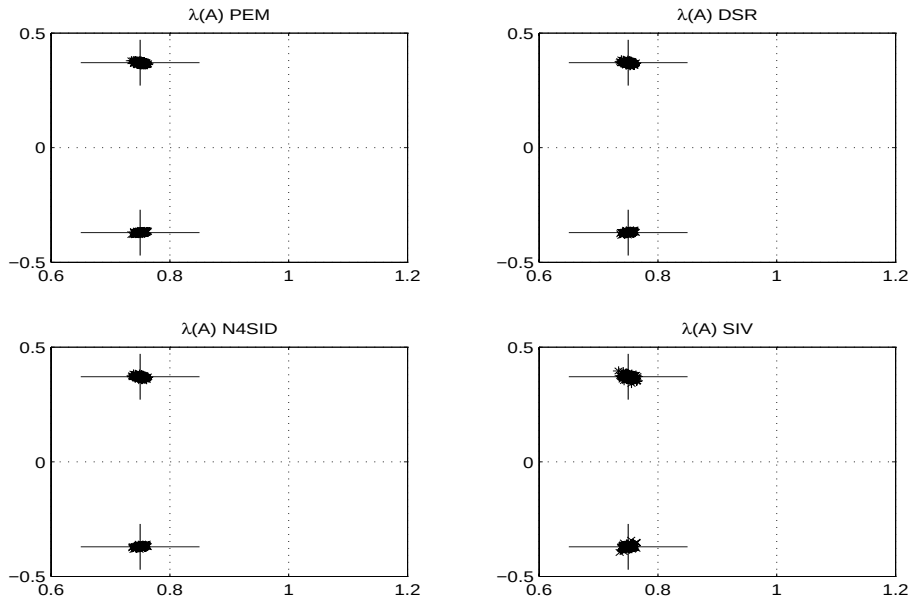

Figure 4.2: Eigenvalue estimates from a Monte Carlo simulation using a low frequent PRBS, u_k^1, as input signal in Example 1, Section 3.1.

Then the signal introduced earlier as u_k^1 is used as reference. Figure 4.3 shows the reference r_k^1 and corresponding input u_k and output y_k.

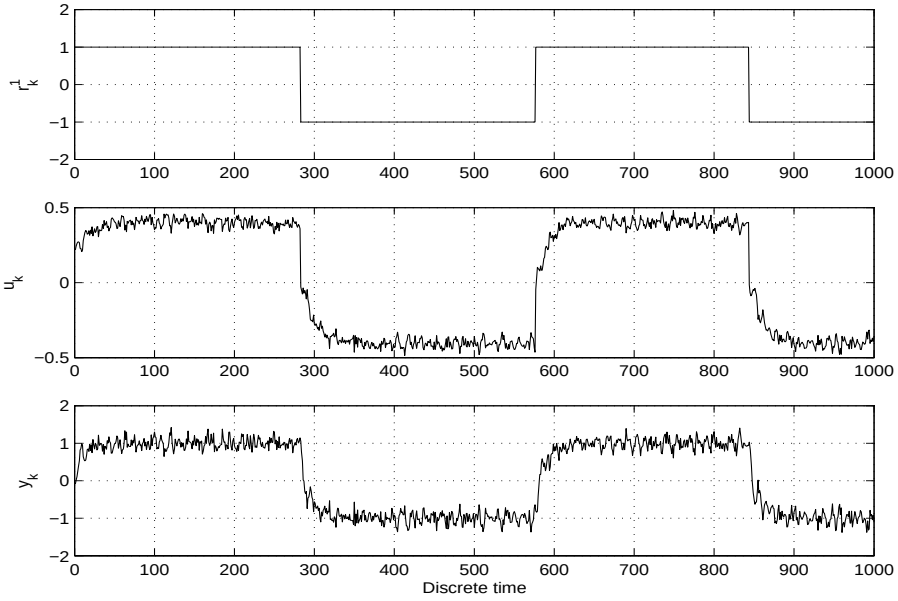

Figure 4.3: Reference r_k^1 and corresponding input u_k and output y_k for Example 1, Section 3.1.

DSR is used with $g = 0$, $L = 4$ and $J = 5$ to estimate the eigenvalues when r_k^1 is used as reference for the system operating in a closed loop. N4SID is used with a horizon that corresponds to $L = 5$ and $g = 0$. SIV is used with horizons that corresponds to $L = 5$, $J = 6$ and $g = 0$. The eigenvalue estimates are shown in Figure 4.4.

Even though the estimates from SIV do not have noticeable larger bias than N4SID or DSR the variance is considerable when r_k^1 is used as reference for the system operating in a closed loop. To check if the problem is that the reference signal does not have high enough order of persistent excitation a white noise sequence with unit variance is used as reference in the closed loop system. The estimated eigenvalues shown in Figure 4.5 are found by using DSR with $g = 0$, $L = 4$ and $J = 5$. N4SID is used with a horizon that corresponds to $L = 5$ and $g = 0$. SIV is used with horizons that correspond to $L = 5$, $J = 6$ and $g = 0$.

There is quite an improvement in the eigenvalue estimates presented in Figure 4.5 compared to Figure 4.4. Now all the methods are unbiased, but still SIV has estimates with much larger variance than the rest. All methods have estimates with reduced variance. What will happen when the process noise and measurement noise are 50 times higher? The eigenvalue estimates in this case are shown

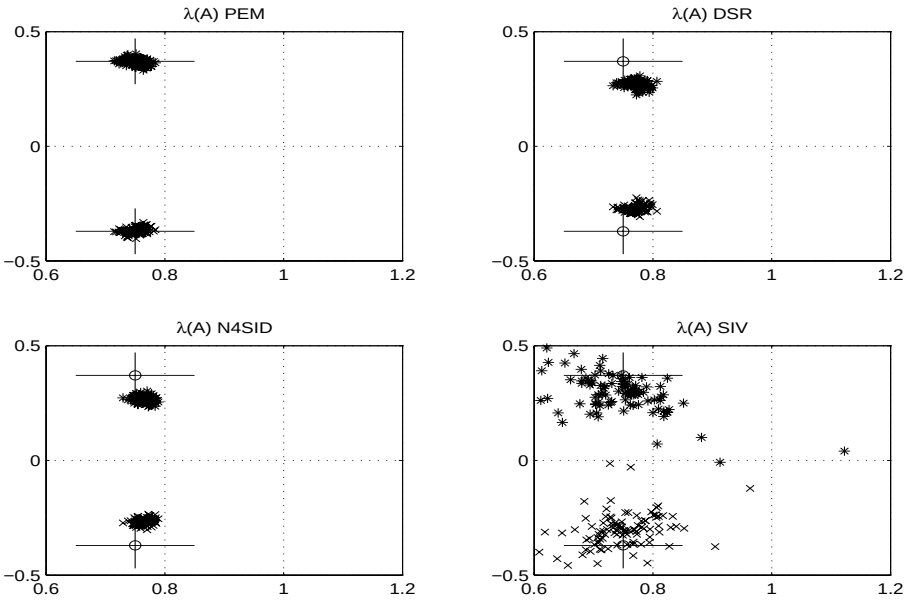

Figure 4.4: Eigenvalue estimates from a Monte Carlo simulation using a low frequent PRBS, r_k^1, as reference signal in the closed loop system, Example 1, Section 3.1.

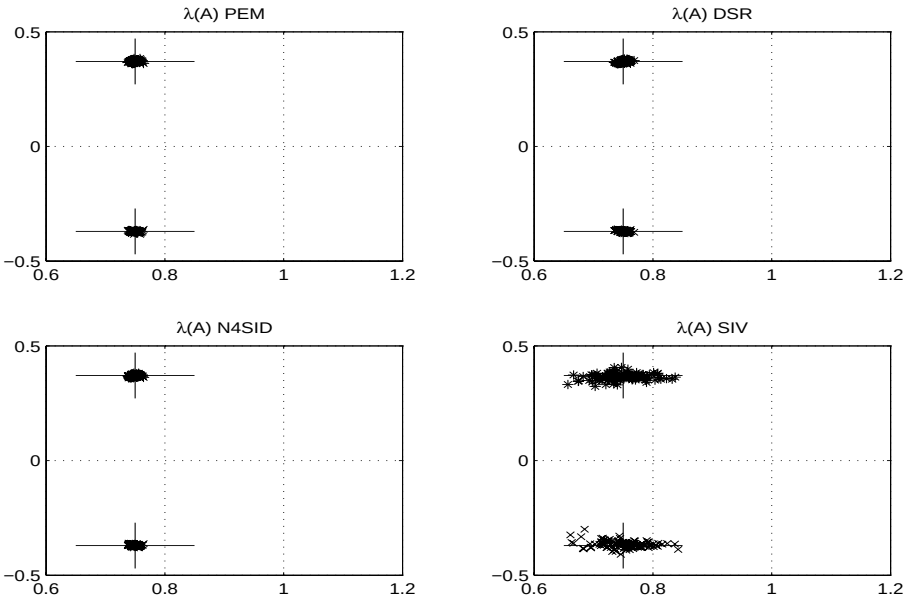

Figure 4.5: Eigenvalue estimates from a Monte Carlo simulation using a white noise sequence with unit variance as reference signal in the closed loop system, Example 1, Section 3.1.

in Figure 4.6.

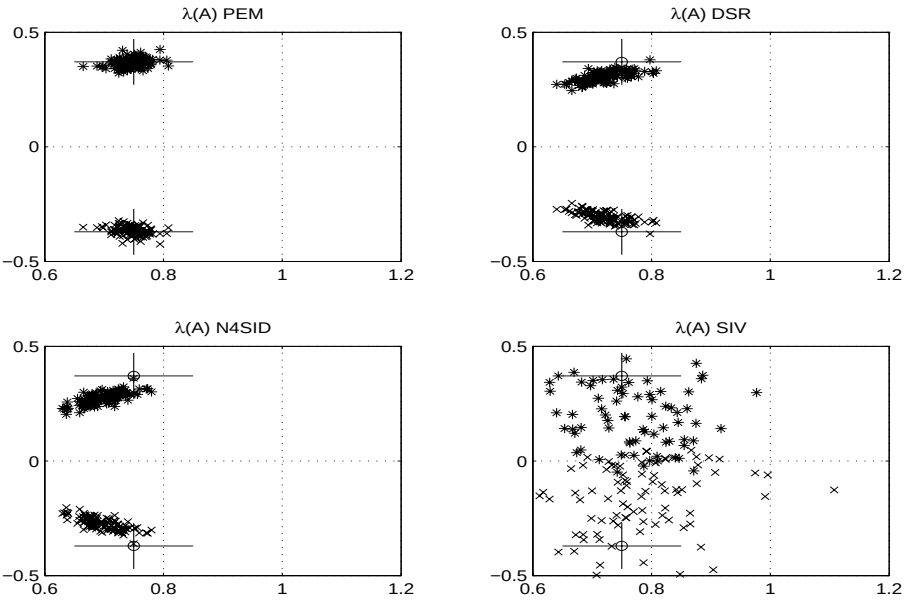

Figure 4.6: Eigenvalue estimates from a Monte Carlo simulation using a white noise sequence with unit variance as reference signal in the closed loop system when the noise level is increased 50 times, Example 1, Section 3.1.

The estimated eigenvalues shown in Figure 4.6 are found by using DSR with $g = 0$, $L = 6$ and $J = 7$. N4SID is used with a horizon that corresponds to $L = 7$ and $g = 0$. SIV is used with horizons that correspond to $L = 15$, $J = 16$ and $g = 0$. All the methods give estimates with increased variance when the noise level is increased. Now both N4SID and DSR have got a small bias. SIV gives estimates with a considerable bias and variance.

The simulations so far have shown that the EIV-based subspace identification method SIV might be used like N4SID or DSR in open loop cases. In closed loop cases SIV is not applicable even though the opposite is stated, and shown, by Chou and Verhaegen (1997) and Gustafsson (2001). One problem is that the assumption that the number of samples $N \to \infty$, but the data sets will always be finite. The requirement to persistence of excitation of the input signal is not clear either. But the fact that the term $U^{\perp}_{L+1|L+1}$ is "missing" as a projection indicates that the method will require white noise as a reference. This is similar to the comments by Di Ruscio (2003a) on the N4SID and the robust version of N4SID.

Since no reason has been found to apply the oblique projection used in N4SID instead of the orthogonal projections used in DSR, all further work will be based

on the DSR algorithm. The simulations done in this section show that with reasonable noise and an input signal with a high order of persistent excitation, like white noise, both N4SID and DSR give unbiased eigenvalue estimates from these finite closed loop data sets. Therefore it is interesting to evaluate how it is possible to improve the estimates when only signals with a low level of persistence of excitations are available.

4.5 A lucky parameter choice

The example in Section 3.3, Example 3, is a single input single output system introduced by Quin and Ljung (2003) to show that the PARSIM-E algorithm gives consistent eigenvalue estimates with closed loop data. The goal in this section is to show that when simulation results, like those in Quin and Ljung (2003), are presented without the appropriate parameter settings for the algorithm it may be suspected to be a lucky parameter choice, like the ones suggested later in this section when using DSR.

A simulation with a PRBS (Pseudo Random Binary Signal) as reference is carried out to get an indication of how much noise is superposed in the system. The result is shown in Figure 4.7.

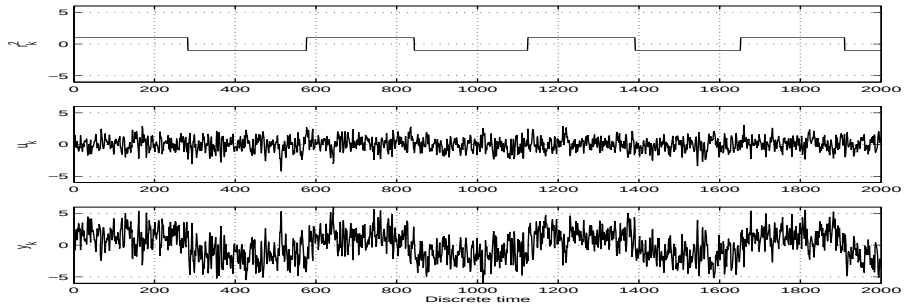

Figure 4.7: The reference signal, r_k^2 and the corresponding input, u_k, and output, y_k, for a particular noise realization e_k, Example 3, Section 3.3.

Figure 4.7 shows that the control system is not effective and there is very much noise. Using the Gaussian white noise with variance 4 as reference and performing a Monte Carlo simulation with 20 runs we get the eigenvalue estimates shown in Figure 4.8. The axes used are the same as in Quin and Ljung (2003).

Here DSR gives a bias that cannot be neglected. When the simulations are performed in Quin and Ljung (2003) the parameter settings used in the method are not presented. Knowing that the PARSIM-P algorithm by Quin and Ljung

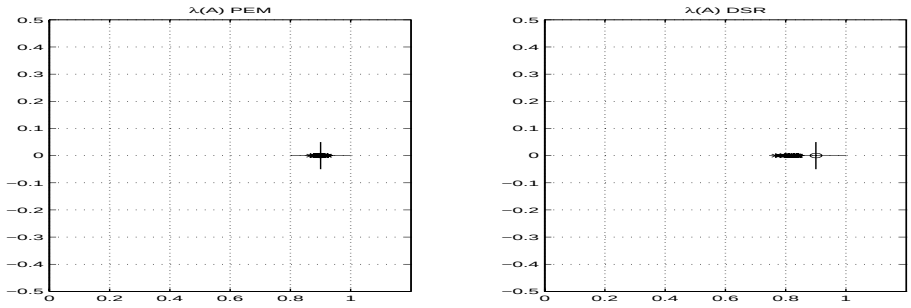

Figure 4.8: The eigenvalue estimates from DSR with $n = 1$, $g = 0$, $L = 1$ and $J = L + 1$ when using a Gaussian white noise with variance 4 as reference, Example 3, Section 3.3.

(2003) is biased for finite past horizon p it is chosen to use DSR with a larger J, in this case $J = 250$. The eigenvalue estimates are shown in Figure 4.9.

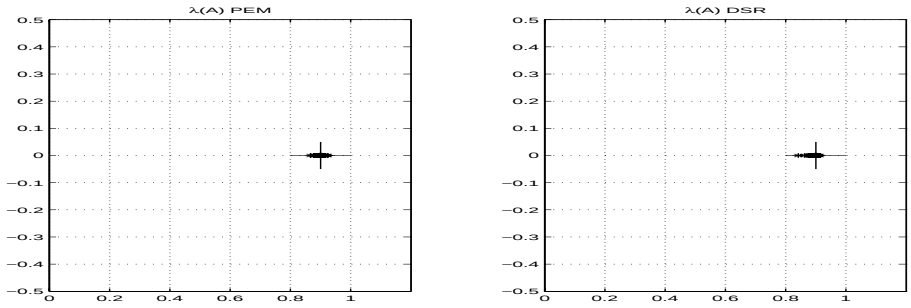

Figure 4.9: The eigenvalue estimates from DSR with $n = 1$, $g = 0$,$L = 1$ and $J = 250$ when using a Gaussian white noise with variance 4 as reference, Example 3, Section 3.3.

Now the bias is marginal, and hard to spot by visual inspection of Figure 4.9.

It has been stated by Di Ruscio (2003a) that using white noise as reference is not necessarily an optimal experiment for DSR when used on open loop data sets. With this in mind a search is performed to find a sinusoid signal with frequency ω and the corresponding parameters in DSR that minimize the Squared Eigenvalue Error criterion, V^1, when $J = L + 1$. When using DSR for identification, the reference signal $r_k = \sin(\omega k)$ with $\omega = 0.4750$ and the parameters $n = 1$, $g = 0$, $L = 10$ and $J = 11$ minimize the Squared Eigenvalue Error criterion, V^1. The eigenvalue estimates are shown in Figure 4.10.

In this case by choosing an appropriate sinusoid signal as reference together with an optimal parameter choice the eigenvalue estimates from DSR are unbiased like

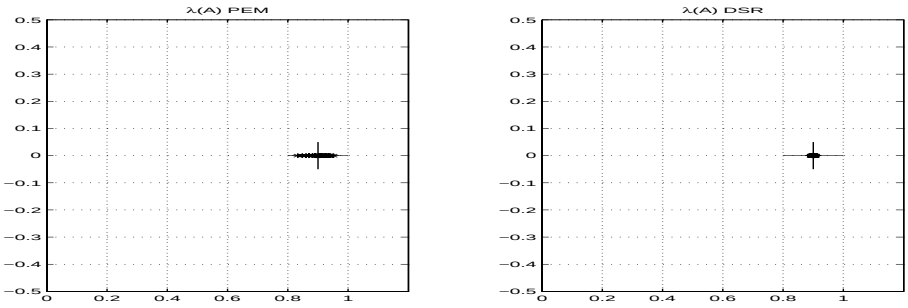

Figure 4.10: The eigenvalue estimates from the Monte Carlo simulation using $r_k = \sin(\omega k)$ with $\omega = 0.4750$ as reference signal when PEM is used with $n_k = 1$ and default parameters and DSR is used with $n = 1$, $g = 0$, $L = 10$ and $J = 11$ for identification, Example 3, Section 3.3.

the estimates from PEM. Which is quite remarkable. What is even more remarkable is that the variance of the eigenvalue estimates from DSR is smaller than the ones from PEM. The choice of reference signals will be investigated further in Section 4.6 together with the choice of parameter settings in DSR when used on closed loop data.

When results are presented it is of great importance that the parameter setting used are presented too. In addition, there ought to be an explanation of why the parameter setting are chosen, alternatively how they are found. The Monte Carlo simulations performed here consists of only 20 runs, which is the number used by Quin and Ljung (2003). Normally if 100 runs are used in the Monte Carlo simulations it is more likely to detect potential problems using the algorithm. In addition to this it is troublesome that methods like PARSIM-E are not implemented as software. Even though all the basic steps are presented in a publication the performance of a Matlab script implemented using the available publications is not guaranteed to work as well as the Matlab code used by the authors.

4.6 Dithering signals to improve the estimates

It has to be examined if an external dithering signal used in the reference or the input of a system operating in closed loop can lead to improved parameter estimates. This idea is presented in Di Ruscio (2003a). To do this the ability to estimate the eigenvalues of the system presented in Section 3.2 will be compared. The estimates from DSR will be compared to estimates from PEM (Prediction Error Method) in the system identification toolbox in Matlab 6.5. Dithering signals with little excitation will be used. With little excitation we do not only mean little excitation in the sense of persistent excitation but also in the level

of excitation. Throughout this section the stationary values in both inputs and outputs will be removed prior to identification. Initial parameter estimates to PEM are provided by using PEM on one extra initial run prior to the Monte Carlo simulation. This section is based on Nilsen and Di Ruscio (2004a) and Nilsen and Di Ruscio (2004b), and extended.

It is believed that DSR will be able to provide eigenvalue estimates with small variance and reasonable bias which is comparable to PEM if the parameter settings and the dithering signal are reasonable.

To get an indication of how much noise is superposed in the system, a simulation with a constant reference of value one superposed a PRBS (Pseudo Random Binary Signal) is carried out. The noise level is considered to be reasonable. The result is shown in Figure 4.11.

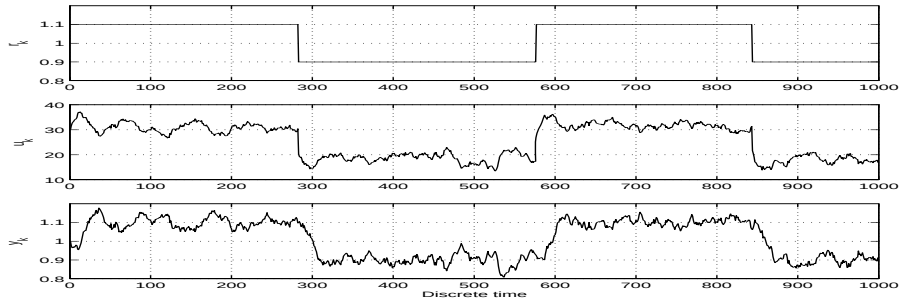

Figure 4.11: The reference signal, r_k and the corresponding input, u_k, and output, y_k, for two particular noise realizations v_k and w_k, Example 2, Section 3.2.

4.6.1 Using a sinusoid signal as a dithering signal in the reference

Closed loop eigenvalue estimates

A sinusoid signal with frequency $\omega = 0.725$ and with magnitude ± 0.1 is chosen as a dithering signal. The reference at time instant k is given by:

$$r_k^3 = 1 + 0.1 \cdot \sin(\omega k). \tag{4.67}$$

The system was simulated 100 times. The same reference was used each time but the noise realization was changed each time. Figure 4.12 shows the estimated eigenvalues from the Monte Carlo simulation. The linearized discrete system has multiple poles, actually two real eigenvalues at 0.875. The parameters in DSR are

chosen as: $n = 2$, $g = 0$, $L = 11$ and $J = 12$. PEM is used with default parameters and $n_k = 1$. Figure 4.12 shows that by using an appropriate dithering signal DSR can provide eigenvalue estimates comparable to PEM on data from a system operating in closed loop. Both estimates are unbiased but DSR has the largest variance on the imaginary part and PEM has the largest variance on the real part.

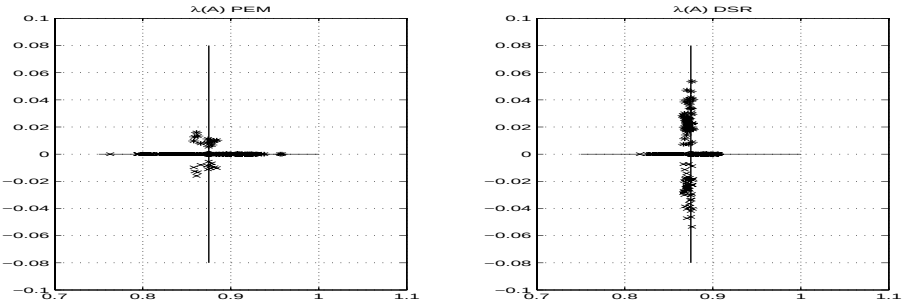

Figure 4.12: Monte Carlo simulation using a constant reference superposed a sinusoid signal as a dithering signal, $r_k^3 = 1 + 0.1 \cdot \sin(\omega k)$ with $\omega = 0.725$, using DSR with $n = 2$, $g = 0$, $L = 11$ and $J = 12$ and PEM with default parameters and $n_k = 1$ for identification in Example 2, Section 3.2.

Table 4.1 contains the mean and the standard deviation (Std) of the eigenvalues of the estimated system matrices from the Monte Carlo simulation when using DSR. Table 4.2 contains corresponding data from PEM. The data in Table 4.1 and Table 4.2 support the conclusion from the visual inspection of Figure 4.12, that DSR gives estimates which are comparable to PEM in this case.

Table 4.1: Mean and standard deviation from the Monte Carlo simulation using a constant reference superposed a sinusoid signal as a dithering signal, $r_k^3 = 1 + 0.1 \cdot \sin(\omega k)$ with $\omega = 0.725$, and using DSR with $n = 2$, $g = 0$, $L = 11$ and $J = 12$ for identification in Example 2, Section 3.2.

DSR	Pole 1	Pole 2
	$[Re \quad Im]$	$[Re \quad Im]$
Mean	[0.8829 0.0118]	[0.8583 -0.0118]
Std	[0.0119 0.0148]	[0.0152 0.0148]

The parameter estimates from DSR as a function of the prediction horizon and the frequency of the sinusoid signal used as a dithering signal in the reference

The experience so far in open loop cases is that the parameter L in DSR should be chosen as small as possible in order to reduce the variance of the estimates.

Table 4.2: Mean and standard deviation from the Monte Carlo simulation using a constant reference superposed a sinusoid signal as a dithering signal, $r_k^3 = 1 + 0.1 \cdot \sin(\omega k)$ with $\omega = 0.725$, and using PEM with default parameters and $n_k = 1$ for identification in Example 2, Section 3.2.

PEM	Pole 1	Pole 2
	$[Re \quad Im]$	$[Re \quad Im]$
Mean	[0.9033 0.0013]	[0.8405 -0.0013]
Std	[0.0198 0.0035]	[0.0253 0.0035]

This is especially important in cases of poorly excited input signals. The parameter J is usually chosen as $J = L$ or $J = L + 1$. It is normally not crucial which of these two alternatives are chosen.

Related to this it is interesting to consider the quality of the parameter estimates from DSR as a function of the frequency, ω, of the dithering signal and the identification horizon, L, in the algorithm when data is collected from a system operating in a closed loop. The parameter J is chosen as $J = L + 1$. The system used is still the chemical reactor operating in the closed loop introduced in Section 3.2. Now we do not only consider a Monte Carlo simulation at one frequency but Monte Carlo simulations in 0.25 steps from 0.25 up to the Nyquist frequency, which is the half of the sampling frequency. In order to evaluate the quality of the estimated model parameters we choose to use a quadratic criterion on the eigenvalues of the estimated model, the Squared Eigenvalue Error criterion, V^1.

Figure 4.13 shows V^1 as a function of ω and L. Each point in the figure is the sum given by V^1 from a Monte Carlo simulation at a specific ω with a specific L.

From Figure 4.13 it is obvious that the choice of ω is the most crucial parameter. Figure 4.13 shows that in this closed loop data set it is not favourable to choose L as small as possible as in open loop cases.

Comparing the parameter estimates from DSR and PEM as functions of the frequency of the sinusoid signal used as a dithering signal in the reference

In Figure 4.14, PEM and DSR are compared as functions of ω of the sinusoid signal used as a dithering signal in the reference. DSR is used with $L = 11$ and $J = L + 1$. It has to be noted that DSR is only comparable to PEM in a low frequent area.

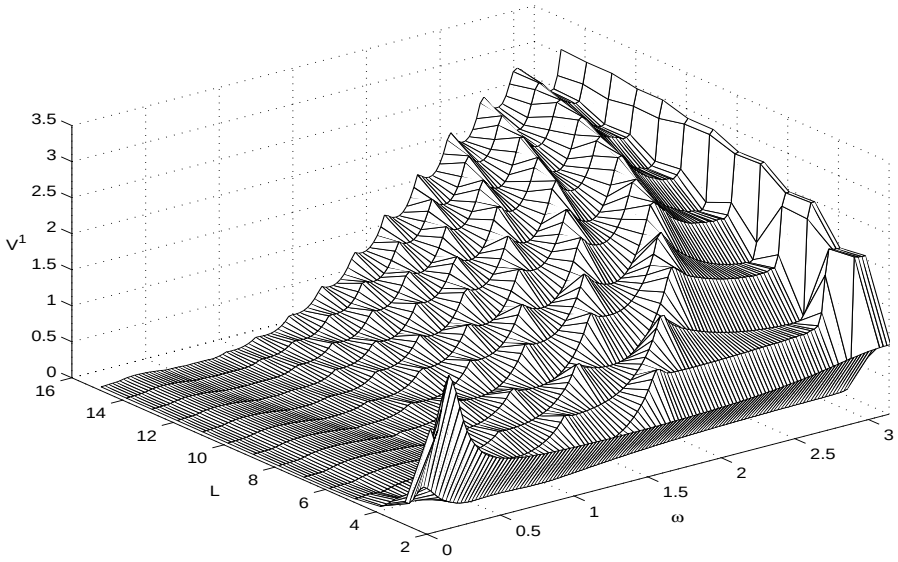

Figure 4.13: The Squared Eigenvalue Error criterion, V^1, as a function of ω and L when using a constant reference superposed a sinusoid signal as a dithering signal, $r_k^3 = 1 + 0.1 \cdot \sin(\omega k)$, and DSR with $n = 2$, $g = 0$ and $J = L + 1$ for identification in Example 2, Section 3.2.

Alternative closed loop quality measures

Transforming the state space model to observable canonical form gives system matrices with the following structure

$$A = \begin{bmatrix} 0 & 1 \\ a_{21} & a_{22} \end{bmatrix}, B = \begin{bmatrix} b_{11} \\ b_{21} \end{bmatrix}, D = \begin{bmatrix} 1 & 0 \end{bmatrix} \tag{4.68}$$

and E is still the zero matrix. The parameters to be estimated in the observable canonical form are collected in a parameter vector

$$\theta = [a_{21} \;\; a_{22} \;\; b_{11} \;\; b_{21}]. \tag{4.69}$$

The true parameter vector is

$$\theta = [-0.7656 \;\; 1.7500 \;\; -0.0010 \;\; -0.0005]. \tag{4.70}$$

In Figure 4.15 the estimated parameters in the parameter vector from DSR and PEM are compared as a function of the runs in the Monte Carlo simulation. The reference is given by Equation (4.67) with $\omega = 0.725$. DSR is used with the parameters $n = 2$, $g = 0$, $L = 11$ and $J = 12$. The parameter estimates from DSR all have small variance. The estimated parameters b_{11} and b_{21} have a bias, which also would have been detected if the estimated zeros had been considered. The

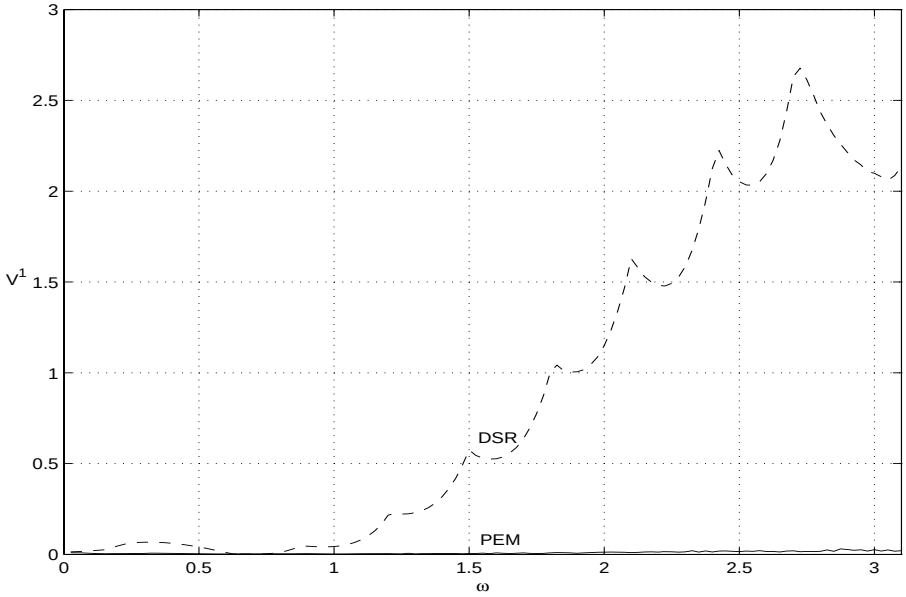

Figure 4.14: The Squared Eigenvalue Error criterion, V^1, as a function of ω, when using a constant reference superposed a sinusoid signal as a dithering signal $r_k^3 = 1 + 0.1 \cdot \sin(\omega k)$ in Example 2, Section 3.2, and the estimates from PEM with default parameters and $n_k = 1$, and DSR with $n = 2$, $g = 0$, $L = 11$ and $J = 12$. PEM is illustrated by the solid line and DSR is illustrated by the broken line.

parameter estimates of a_{21} and a_{22} from PEM are characterized by large variance compared to the estimates from DSR, which coincide with our observation of the estimated eigenvalues. The mean and standard deviation of the parameters in the parameter vector are listed in Table 4.3 and Table 4.4 for DSR and PEM, respectively.

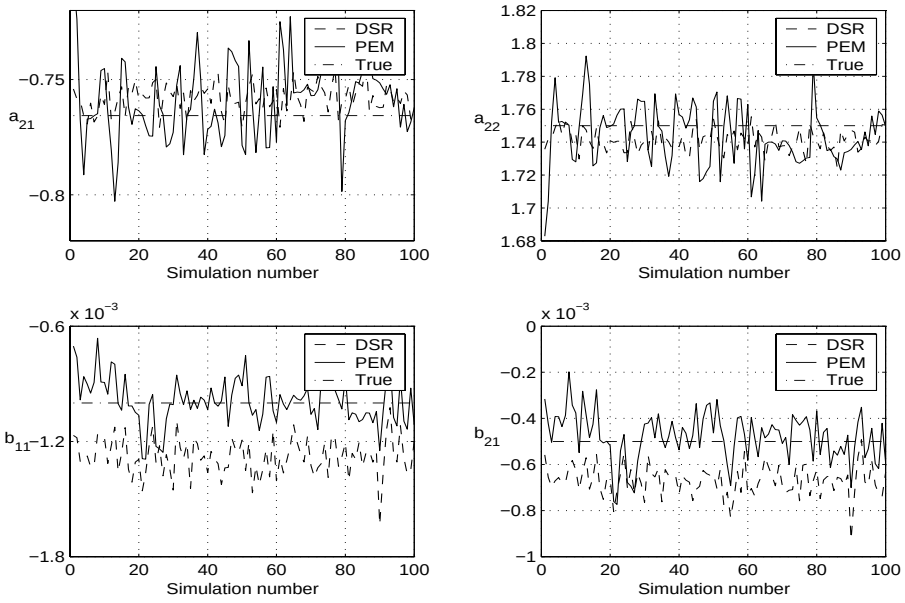

Figure 4.15: The parameter vector, Equation (4.69), as a function of the runs in the Monte Carlo simulation of Example 2, Section 3.2, using a constant reference superposed a sinusoid signal as a dithering signal, $r_k^3 = 1 + 0.1 \cdot \sin(\omega k)$ with $\omega = 0.725$, using DSR with $n = 2$, $g = 0$, $L = 11$ and $J = 12$ and PEM with default parameters and $n_k = 1$ for identification. PEM is illustrated by the solid line, DSR is illustrated by the broken line and the true value is illustrated by the dash dotted line.

Table 4.3: Mean and standard deviation from the Monte Carlo simulation of Example 2, Section 3.2, using a constant reference superposed a sinusoid signal as a dithering signal, $r_k^3 = 1 + 0.1 \cdot \sin(\omega k)$ with $\omega = 0.725$, and using DSR with $n = 2$, $g = 0$, $L = 11$ and $J = 12$ for identification.

DSR	a_{21}	a_{22}	b_{11}	b_{21}
Mean	-0.7580	1.7412	-0.0013	-0.0007
Std	0.0062	0.0066	0.0001	0.0001

Table 4.4: Mean and standard deviation from the Monte Carlo simulation of Example 2, Section 3.2, using a constant reference superposed a sinusoid signal as a dithering signal, $r_k^3 = 1 + 0.1 \cdot \sin(\omega k)$ with $\omega = 0.725$, and using PEM with default parameters for identification.

PEM	a_{21}	a_{22}	b_{11}	b_{21}
Mean	-0.7589	1.7438	-0.0010	-0.0005
Std	0.0167	0.0185	0.0001	0.0001

4.6.2 Using a PRBS as a dithering signal in the reference

Closed loop eigenvalue estimates

A PRBS (Pseudo Random Binary Signal) generated from **idinput(N,'prbs',[0 B])** is chosen as dithering signal. This function is a part of the system identification toolbox in Matlab. When choosing $B = 0.15$ in the **idinput** command line, as we choose here, the signal is constant over intervals of length $\frac{1}{0.15}$ (the clock period). An increase in B will therefore give a signal with higher frequency. The term higher frequency means that the signal is kept at respectively high or low level for shorter periods of time, and therefore is changing more frequently. The reference, $R^4 \in \mathbb{R}^N$, at time instant $1 \leq k \leq N$ is generated by

$$R^4 = \text{ones}(N, 1) + 0.1 \cdot \text{idinput}(N, 'prbs', [0\, B]). \tag{4.71}$$

The system was simulated 100 times. The same reference was used each time but the noise realization was changed each time. Figure 4.16 shows the estimated eigenvalues from the Monte Carlo simulation. The linearized discrete system has multiple poles, actually two real eigenvalues at 0.875. The parameters in DSR are chosen as: $n = 2$, $g = 0$, $L = 11$ and $J = 12$. PEM is used with default parameters and $n_k = 1$. Figure 4.16 shows that by using an appropriate PRBS as a dithering signal DSR can provide an eigenvalue estimate comparable to PEM on data from a system operating in a closed loop.

The mean and standard deviation are presented in Table 4.5 for DSR and Table 4.6 for PEM.

The parameter estimates from DSR as a function of the prediction horizon and the frequency of the PRBS used as a dithering signal in the reference

It is interesting to consider the quality of the parameter estimates from DSR as a function of the frequency, B, of the dithering signal and the identification

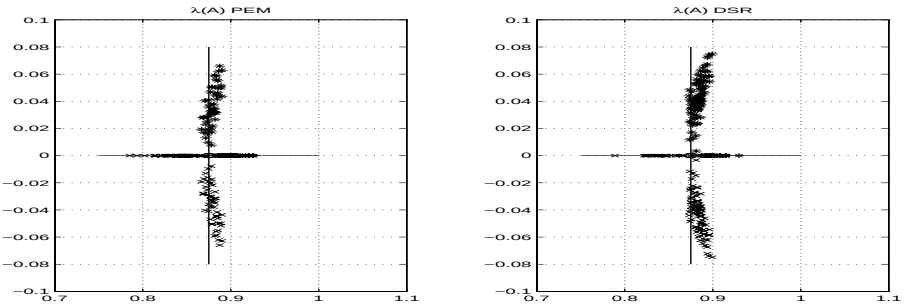

Figure 4.16: Monte Carlo simulation of Example 2, Section 3.2, using a constant reference superposed a PRBS as a dithering signal generated from $R^4 = \text{ones}(N,1) + 0.1 \cdot \text{idinput}(N,'prbs',[0\ B])$ with $B = 0.2$, using DSR with $n = 2$, $g = 0$, $L = 11$ and $J = 12$ and PEM with default parameters and $n_k = 1$ for identification.

Table 4.5: Mean and standard deviation from the Monte Carlo simulation of Example 2, Section 3.2, using a constant reference superposed a PRBS as a dithering signal generated from $R^4 = \text{ones}(N,1) + 0.1 \cdot \text{idinput}(N,'prbs',[0\ B])$ with $B = 0.2$, using DSR with $n = 2$, $g = 0$, $L = 11$ and $J = 12$ for identification.

DSR	Pole 1	Pole 2
	$[Re \quad Im]$	$[Re \quad Im]$
Mean	[0.8878 0.0338]	[0.8751 -0.0338]
Std	[0.01030.0223]	[0.0201 0.0223]

horizon, L, in the algorithm when data is collected from a system operating in a closed loop. The parameter J is chosen as $J = L + 1$. The system used is still the chemical reactor operating in a closed loop introduced in Section 3.2. Let us not only consider a Monte Carlo simulation at one constant B, using **idinput** to generate PRBS, but Monte Carlo simulations at 99 different frequencies. Actually 99 different B's, $0.01 \leq B \leq 0.99$. The Squared Eigenvalue Error criterion V^1, a quadratic criterion on the eigenvalues of the estimated model, is chosen to evaluate the quality of the estimated model parameters.

Figure 4.17 shows V^1 as a function of B and L. Each point in the figure is the sum given by V^1 from a Monte Carlo simulation at a specific B with a specific L.

From Figure 4.17 it is obvious that the choice of B is the most crucial parameter if the parameter L is chosen "large enough". This coincides with the observations made when a sinusoid signal was used as a dithering signal. Therefore the rule for choosing the prediction horizon L in DSR when performing closed loop identification will be to choose a horizon which is larger than when performing

Table 4.6: Mean and standard deviation from the Monte Carlo simulation of Example 2, Section 3.2, using a constant reference superposed a PRBS as a dithering signal generated from $R^4 = \text{ones}(N,1) + 0.1 \cdot \text{idinput}(N,'prbs',[0\ B])$ with $B = 0.2$, using PEM with default parameters and $n_k = 1$ for identification.

PEM	Pole 1	Pole 2
	$[Re \quad Im]$	$[Re \quad Im]$
Mean	[0.8903 0.0171]	[0.8590 -0.0171]
Std	[0.0134 0.0210]	[0.0236 0.0210]

open loop identification.

Comparing the parameter estimates from DSR and PEM as functions of the frequency of the PRBS used as a dithering signal in the reference

In Figure 4.18, PEM and DSR are compared when a PRBS is used as a dithering signal in the reference. The criterion V^1 is plotted as a function of B, in **idinput**. DSR is used with $L = 11$ and $J = L + 1$. PEM is used with default parameters and $n_k = 1$. There is no specific frequency area where DSR is much poorer than PEM. This differs from the case where a sinusoid signal was used as a dithering signal where a low frequent signal ought to be chosen as a dithering signal.

Alternative closed loop quality measures

For the system in observable canonical form the parameters in Equation (4.69) are considered with the values given by Equation (4.70). In Figure 4.19 the estimated parameters in the parameter vector from DSR and PEM are compared as a function of the runs in the Monte Carlo simulation. The reference is given by Equation (4.71) with $B = 0.2$. DSR is used with the parameters $n = 2$, $g = 0$, $L = 11$ and $J = 12$. The mean and standard deviation of the parameters in the parameter vector are listed in Table 4.15 and Table 4.16 for DSR and PEM, respectively.

Table 4.7: Mean and standard deviation from the Monte Carlo simulation of Example 2, Section 3.2, using a constant reference superposed a PRBS as a dithering signal, $R^4 = \text{ones}(N,1) + 0.1 \cdot \text{idinput}(N,'prbs',[0\ B])$ with $B = 0.2$, and using DSR with $n = 2$, $g = 0$, $L = 12$ and $J = 13$ for identification

DSR	a_{21}	a_{22}	b_{11}	b_{21}
Mean	-0.7784	1.7629	-0.0010	-0.0005
Std	0.0157	0.0161	0.0001	0.0001

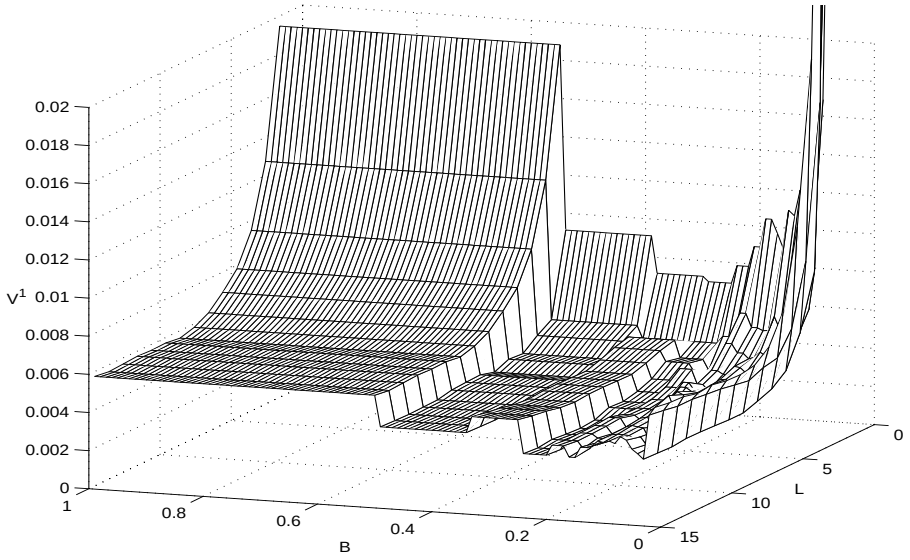

Figure 4.17: The Squared Eigenvalue Error criterion, V^1, as a function of B and L when using a constant reference superposed a PRBS as a dithering signal, $R^4 = \text{ones}(N,1) + 0.1 \cdot \text{idinput}(N,'prbs',[0\ B])$ in Example 2, Section 3.2, and DSR with $n = 2$, $g = 0$ and $J = L + 1$ for identification.

Table 4.8: Mean and standard deviation from the Monte Carlo simulation of Example 2, Section 3.2, using a constant reference superposed a PRBS as a dithering signal, $R^4 = \text{ones}(N,1) + 0.1 \cdot \text{idinput}(N,'prbs',[0\ B])$ with $B = 0.2$, and using PEM with default parameters and $n_k = 1$ for identification.

PEM	a_{21}	a_{22}	b_{11}	b_{21}
Mean	-0.7652	1.7492	-0.0010	-0.0005
Std	0.0142	0.0146	0.0001	0.0001

Now the parameter estimates from DSR are directly comparable to the estimates from PEM. The most important difference from using a sinusoid as a dithering signal is that the estimation of the parameters b_{11} and b_{21} by DSR are now unbiased. The reason for this is that the level of persistent excitation is increased.

4.6.3 Using a sinusoid signal as a dithering signal in the input

The previous sections have shown that when using a dithering signal in the reference of a chemical reactor operating in closed loop, DSR gives unbiased eigenvalue estimates if the parameter settings are reasonable. If the level of persistent excitation is sufficiently high the estimates of the zeros are unbiased too. Therefore

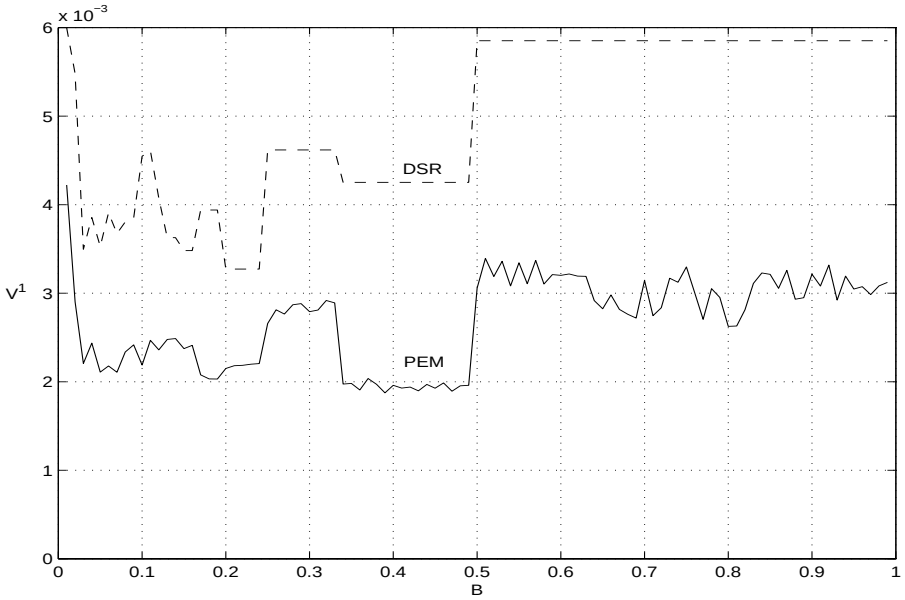

Figure 4.18: The Squared Eigenvalue Error criterion, V^1, as a function of B, when using a constant reference superposed a PRBS as a dithering signal $R^4 = \text{ones}(N,1) + 0.1 \cdot \text{idinput}(N,'prbs',[0\ B])$ in Example 2, Section 3.2, and the estimates from PEM with default parameters and $n_k = 1$, and DSR with $n = 2$, $g = 0$, $L = 11$ and $J = 12$. PEM is illustrated by the solid line and DSR is illustrated by the broken line.

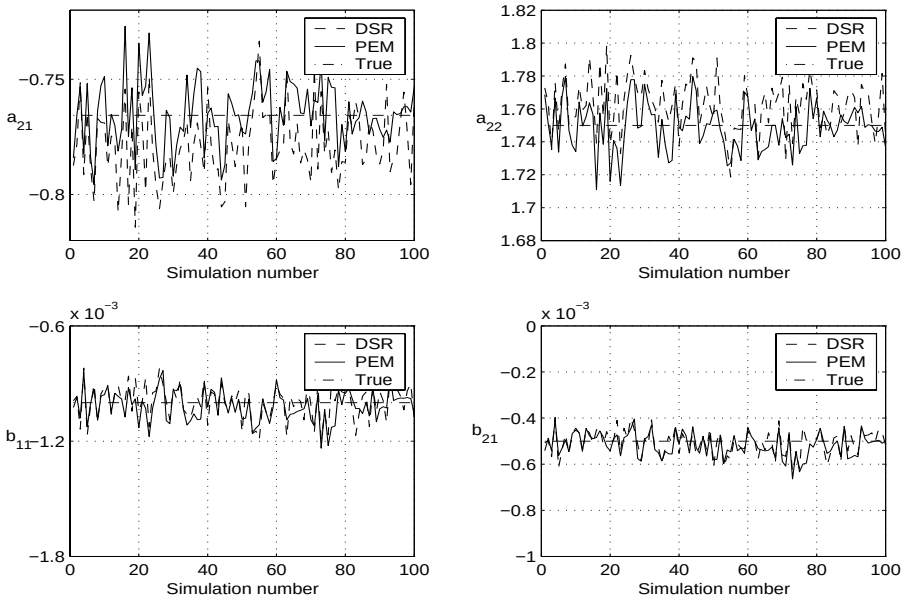

Figure 4.19: The parameter vector Equation (4.69) as a function of the runs in the Monte Carlo simulation of Example 2, Section 3.2, using a constant reference superposed a PRBS as a dithering signal, $R^4 = \text{ones}(N, 1) + 0.1 \cdot \text{idinput}(N, 'prbs', [0\ B])$ with $B = 0.2$, using DSR with $n = 2$, $g = 0$, $L = 11$ and $J = 12$ and PEM with default parameters and $n_k = 1$ for identification. PEM is illustrated by the solid line, DSR is illustrated by the broken line and the true value is illustrated by the dash dotted line.

it is interesting to consider the same system as used in the previous sections, the system presented in Section 3.2, to investigate if the same results occur when a constant reference is used and a dithering signal is added on the input.

Now the reference is a constant value given by

$$r_k^5 = 1. \tag{4.72}$$

To make it easier to compare the results, the same dithering signal will be used as when the dithering signal is used in the reference. But to make the influence of the dithering signal on the output approximately equal, the dithering signal now has to be scaled by the inverse of the discrete DC gain. The discrete DC gain is given by Equation (2.18) with $z = 1$. In this case $H^d(1) = 0.0160$.

Closed loop eigenvalue estimates

A sinusoid signal with frequency $\omega = 0.725$ and with magnitude ± 6.25 is chosen as a dithering signal. The dithering signal in the input at time instant k is given by

$$dU_k^1 = 6.25 \cdot \sin(\omega k). \tag{4.73}$$

The system was simulated 100 times. The same reference and dithering signal in the input was used each time but the noise realization was changed each time. Figure 4.20 shows the estimated eigenvalues from the Monte Carlo simulation. The parameters in DSR are chosen as: $n = 2$, $g = 0$, $L = 11$ and $J = 12$. PEM is used with default parameters and $n_k = 1$. Figure 4.20 shows that by using an appropriate dithering signal on the input DSR can provide eigenvalue estimates comparable to PEM on data from a system operating in closed loop. Both methods give unbiased estimates. In this case DSR has smaller variance than PEM. This was not the case in Section 4.6.1, when the dithering signal was used in the reference, then DSR had the largest variance on the imaginary part and PEM had the largest variance on the real part. Using the dithering signal in the input of the system instead of in the reference does not affect the ability of DSR to estimate the eigenvalues of the system. Regarding PEM, the variance of the imaginary part of the eigenvalue estimates is increased when the dithering signal is used in the input of the system instead of in the reference.

Table 4.9 contains the mean and the standard deviation (Std) of the eigenvalues of the estimated system matrices from the Monte Carlo simulation when using DSR. Table 4.10 contains corresponding data from PEM. The data in Table 4.9 and Table 4.10 support the conclusion from the visual inspection of Figure 4.20,

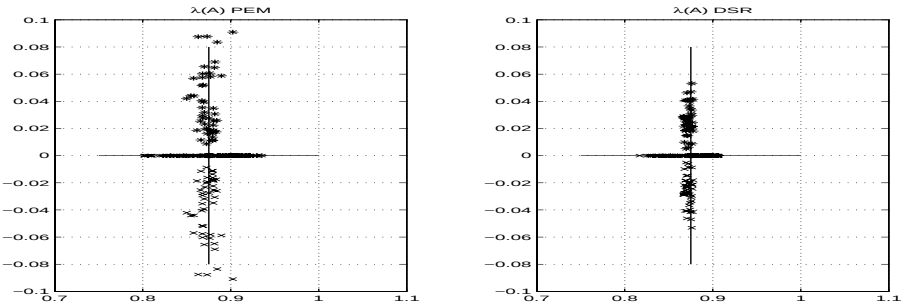

Figure 4.20: Monte Carlo simulation of Example 2, Section 3.2, using a constant reference, $r_k^5 = 1$, and the input superposed a sinusoid signal as a dithering signal, $dU_k^1 = 6.25 \cdot \sin(\omega k)$ with $\omega = 0.725$, using DSR with $n = 2$, $g = 0$, $L = 11$ and $J = 12$ and PEM with default parameters and $n_k = 1$ for identification.

that DSR gives estimates which are comparable to PEM, or slightly better, in this case.

Table 4.9: Mean and standard deviation from the Monte Carlo simulation of Example 2, Section 3.2, using a constant reference, $r_k^5 = 1$, and the input superposed a sinusoid signal as a dithering signal, $dU_k^1 = 6.25 \cdot \sin(\omega k)$ with $\omega = 0.725$, and using DSR with $n = 2$, $g = 0$, $L = 11$ and $J = 12$ for identification.

DSR	Pole 1	Pole 2
	$[Re \quad Im]$	$[Re \quad Im]$
Mean	[0.8827 0.0121]	[0.8585 -0.0121]
Std	[0.0119 0.0151]	[0.0152 0.0151]

The parameter estimates from DSR as a function of the prediction horizon and the frequency of the sinusoid signal used as a dithering signal in the input

Like in Section 4.6.1 it is interesting to consider the quality of the parameter estimates from DSR as a function of the frequency, ω, of the dithering signal and the identification horizon, L, in the algorithm when data is collected from a system operating in closed loop. The parameter J is chosen as $J = L + 1$. Monte Carlo simulations were performed in 0.25 steps from 0.25 up to the Nyquist frequency. The Squared Eigenvalue Error criterion, V^1, is used to evaluate the quality of the estimated model parameters.

Figure 4.21 shows V^1 as a function of ω and L. Each point in the figure is the sum given by V^1 from a Monte Carlo simulation at a specific ω with a specific L.

Table 4.10: Mean and standard deviation from the Monte Carlo simulation of Example 2, Section 3.2, using a constant reference, $r_k^5 = 1$, and the input superposed a sinusoid signal as a dithering signal, $dU_k^1 = 6.25 \cdot \sin(\omega k)$ with $\omega = 0.725$, and using PEM with default parameters and $n_k = 1$ for identification

PEM	Pole 1	Pole 2
	$[Re \quad Im]$	$[Re \quad Im]$
Mean	[0.8888 0.0177]	[0.8390 -0.0177]
Std	[0.0210 0.0248]	[0.1220 0.0248]

Visual inspection of Figure 4.21 shows that it is similar to Figure 4.13. Therefore the conclusion is the same. The choice of ω is the most crucial parameter. Figure 4.21 shows, as in Figure 4.13, that in this closed loop data set it is not favourable to choose L as small as possible as in open loop cases.

Comparing the parameter estimates from DSR and PEM as functions of the frequency of the sinusoid signal used as a dithering signal in the input

In Figure 4.22 PEM and DSR are compared as functions of ω of the sinusoid signal used as a dithering signal in the input. DSR is used with $L = 11$ and $J = L + 1$. It has to be noted that DSR is only comparable to PEM in a low frequent area, just like when the dithering signal is used in the reference as shown in Figure 4.14.

Alternative closed loop quality measures

When the state space model of the system is transformed to observable canonical form the parameters to be estimated can be collected in the parameter vector, Equation (4.69), with the true value given by Equation (4.70).

In Figure 4.23 the estimated parameters in the parameter vector from DSR and PEM are compared as a function of the runs in the Monte Carlo simulation. The reference is given by Equation (4.72) and the dithering signal on the input is given by Equation (4.73) with $\omega = 0.725$. DSR is used with the parameters $n = 2$, $g = 0$, $L = 11$ and $J = 12$.

Like in the case when the dithering signal is used in the reference, Figure 4.15, the parameter estimates from DSR all have small variance. The difference is that now the estimated parameters b_{11} and b_{21} are unbiased. The parameter estimates of a_{21} and a_{22} from PEM are characterized by large variance compared to the estimates from DSR, which of course coincide with our observation of the estimated eigenvalues. The mean and standard deviation of the parameters in the param-

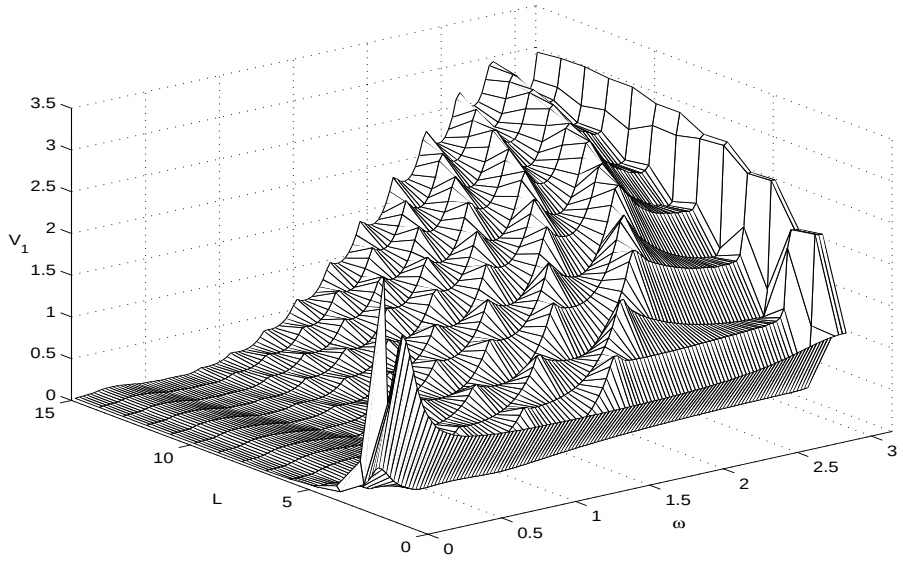

Figure 4.21: The Squared Eigenvalue Error criterion, V^1, as a function of ω and L when using a constant reference, $r_k^5 = 1$, and the input superposed a sinusoid signal as a dithering signal, $dU_k^1 = 6.25 \cdot \sin(\omega k)$ in Example 2, Section 3.2, and DSR with $n = 2$, $g = 0$ and $J = L + 1$ for identification.

eter vector are listed in Table 4.11 and Table 4.12 for DSR and PEM, respectively.

Table 4.11: Mean and standard deviation from the Monte Carlo simulation of Example 2, Section 3.2, using a constant reference, $r_k^5 = 1$, and the input superposed a sinusoid signal as a dithering signal, $dU_k^1 = 6.25 \cdot \sin(\omega k)$ with $\omega = 0.725$, and using DSR with $n = 2$, $g = 0$, $L = 11$ and $J = 12$ for identification.

DSR	a_{21}	a_{22}	b_{11}	b_{21}
Mean	-0.7580	1.7412	-0.0011	-0.0006
Std	0.0062	0.0066	0.0001	0.0000

4.6.4 Using a PRBS as a dithering signal in the input

When a sinusoid signal was used as a dithering signal in the reference the eigenvalue estimates from DSR were unbiased and the estimates of the zeros were biased. If a PRBS is used as a dithering signal in the reference instead, the estimates of the eigenvalues and the zeros are unbiased. When a sinusoid signal is used as a dithering signal in the input the estimates of the eigenvalues and the zeros are unbiased. Therefore it is interesting to investigate if using a PRBS as

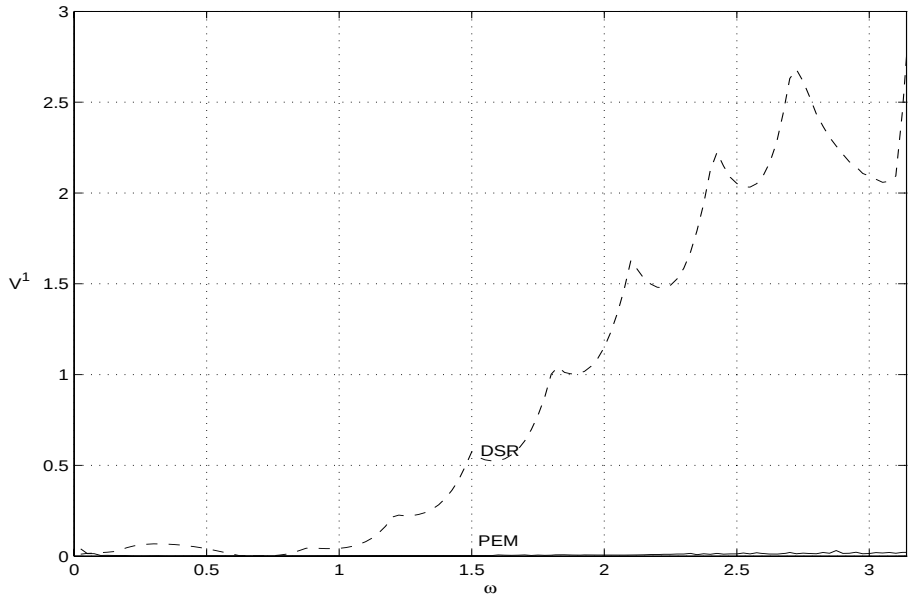

Figure 4.22: The Squared Eigenvalue Error criterion, V^1, as a function of ω, when using a constant reference, $r_k^5 = 1$, and the input superposed a sinusoid signal as a dithering signal, $dU_k^1 = 6.25 \cdot \sin(\omega k)$ in Example 2, Section 3.2, and the estimates from PEM with default parameters and $n_k = 1$, and DSR with $n = 2$, $g = 0$, $L = 11$ and $J = 12$. PEM is illustrated by the solid line and DSR is illustrated by the broken line

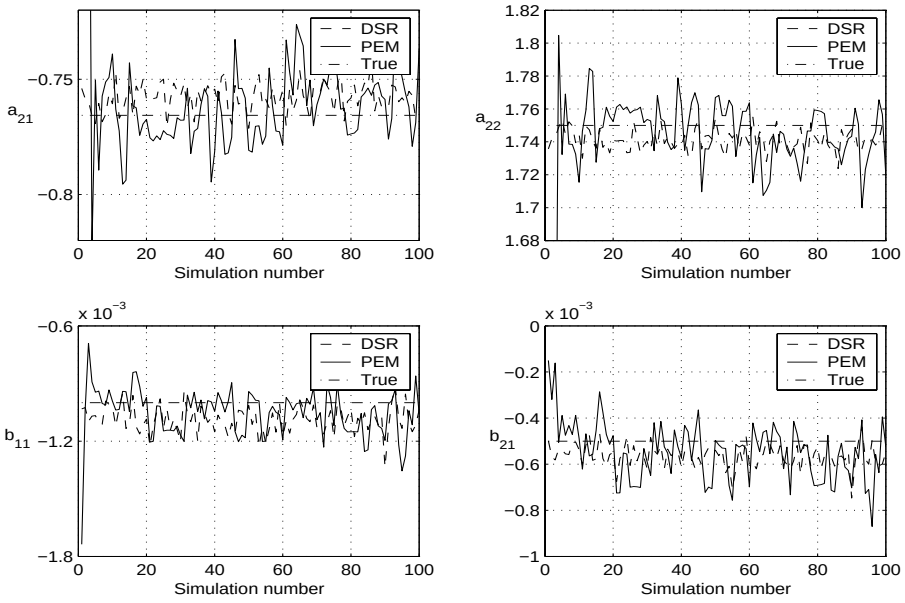

Figure 4.23: The parameter vector, Equation (4.69), as a function of the runs in the Monte Carlo simulation of Example 2, Section 3.2, using a constant reference, $r_k^5 = 1$, and the input superposed a sinusoid signal as a dithering signal, $dU_k^1 = 6.25 \cdot \sin(\omega k)$ with $\omega = 0.725$, using DSR with $n = 2$, $g = 0$, $L = 11$ and $J = 12$ and PEM with default parameters and $n_k = 1$ for identification. PEM is illustrated by the solid line, DSR is illustrated by the broken line and the true value is illustrated by the dash dotted line.

Table 4.12: Mean and standard deviation from the Monte Carlo simulation of Example 2, Section 3.2, using a constant reference, $r_k^5 = 1$, and the input superposed a sinusoid signal as a dithering signal, $dU_k^1 = 6.25 \cdot \sin(\omega k)$ with $\omega = 0.725$, and using PEM with default parameters for identification.

PEM	a_{21}	a_{22}	b_{11}	b_{21}
Mean	-0.7472	1.7278	-0.0011	-0.0005
Std	0.1077	0.1285	0.0001	0.0001

the dithering signal in the input will reduce the variance compared to when a sinusoid signal is used as the dithering signal in the input.

Closed loop eigenvalue estimates

The reference is still given by Equation (4.72) but the dithering signal on the input, $dU^2 \in \mathbb{R}^N$, at time instant $1 \leq k \leq N$ is generated by

$$dU^2 = 6.25 \cdot \text{idinput}(N, 'prbs', [0\, B]). \tag{4.74}$$

The system was simulated 100 times. The same reference and dithering signal on the input was used each time but the noise realization was changed each time. Figure 4.24 shows the estimated eigenvalues from the Monte Carlo simulation. The parameters in DSR are chosen as: $n = 2$, $g = 0$, $L = 11$ and $J = 12$. PEM is used with default parameters and $n_k = 1$. Figure 4.16 shows that by using an appropriate PRBS as the dithering signal, DSR can provide eigenvalue estimates comparable to PEM on data from a system operating in a closed loop, but the variance in the eigenvalue estimates from DSR is not reduced compared to when a sinusoid signal is used as the dithering signal in the input, Figure 4.20.

The mean and standard deviations are presented in Table 4.13 for DSR and Table 4.14 for PEM.

The parameter estimates from DSR as a function of the prediction horizon and the frequency of the PRBS used as the dithering signal in the input

Like in Section 4.6.1 it is interesting to consider the quality of the parameter estimates from DSR as a function of the frequency, B, of the PRBS used as the dithering signal and the identification horizon, L, in the algorithm when data is collected from a system operating in a closed loop. The parameter J is chosen as $J = L + 1$. Monte Carlo simulations at 99 different frequencies are performed with $0.01 \leq B \leq 0.99$. The Squared Eigenvalue Error criterion, V^1, is used to

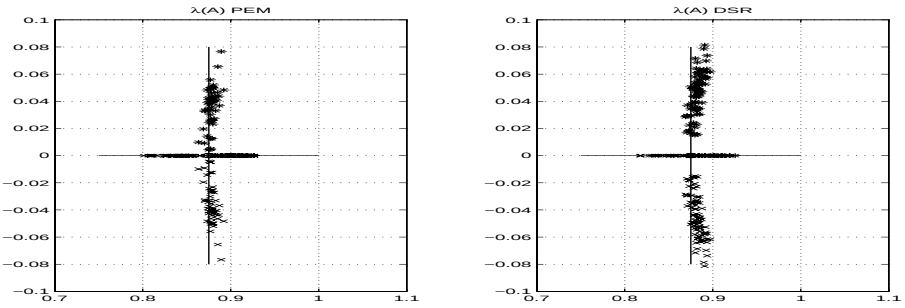

Figure 4.24: Monte Carlo simulation of Example 2, Section 3.2, using a constant reference, $r_k^5 = 1$, and the input superposed a PRBS as the dithering signal generated from $dU^2 = 6.25 \cdot \text{idinput}(N,'prbs',[0\ B])$ with $B = 0.2$, using DSR with $n = 2$, $g = 0$, $L = 11$ and $J = 12$ and PEM with default parameters and $n_k = 1$ for identification.

Table 4.13: Mean and standard deviation from the Monte Carlo simulation of Example 2, Section 3.2, using a constant reference, $r_k^5 = 1$, and the input superposed a PRBS as the dithering signal generated from $dU^2 = 6.25 \cdot \text{idinput}(N,'prbs',[0\ B])$ with $B = 0.2$, using DSR with $n = 2$, $g = 0$, $L = 11$ and $J = 12$ for identification.

DSR	Pole 1	Pole 2
	$[Re \quad Im]$	$[Re \quad Im]$
Mean	[0.8865 0.0349]	[0.8748 -0.0349]
Std	[0.0111 0.0239]	[0.0170 0.0239]

evaluate the quality of the estimated model parameters.

Figure 4.25 shows V^1 as a function of ω and L. Each point in the figure is the sum given by V^1 from a Monte Carlo simulation at a specific ω with a specific L.

Figure 4.25 shows that the choice of B is the most crucial parameter if the parameter L is chosen "large enough", just as when a PRBS is used as a dithering signal in the reference Figure 4.17. This coincides with the observations made when a sinusoid signal was used as a dithering signal. Therefore the rule for choosing the prediction horizon L in DSR when performing closed loop identification will be to choose a horizon which is larger than when performing open loop identification.

Comparing the parameter estimates from DSR and PEM as functions of the frequency of the PRBS used as a dithering signal on the input

In Figure 4.26, PEM and DSR are compared as functions of B of the PRBS used as a dithering signal on the input. DSR is used with $L = 11$ and $J = L+1$. There

Table 4.14: Mean and standard deviation from the Monte Carlo simulation of Example 2, Section 3.2, using a constant reference, $r_k^5 = 1$, and the input superposed a PRBS as the dithering signal generated from $dU^2 = 6.25 \cdot \text{idinput}(N,'prbs',[0\ B])$ with $B = 0.2$, using PEM with default parameters and $n_k = 1$ for identification.

PEM	Pole 1	Pole 2
	$[Re \quad Im]$	$[Re \quad Im]$
Mean	[0.8907 0.0178]	[0.8589 -0.0178]
Std	[0.0158 0.0210]	[0.0228 0.0210]

is no specific frequency area where DSR is much poorer than PEM. This is the same observation as when the PRBS is used as a dithering signal, but it differs from the cases where a sinusoid signal was used as a dithering signal, Figure 4.13 and Figure 4.25, respectively in the reference and on the input. But in all cases a low frequent signal ought to be chosen as a dithering signal.

Alternative closed loop quality measures

When the state space model of the system is transformed to observable canonical form the parameters to be estimated can be collected in the parameter vector, Equation (4.69), with the true value given by Equation (4.70).

In Figure 4.27 the estimated parameters in the parameter vector from DSR and PEM are compared as a function of the runs in the Monte Carlo simulation. The reference is given by Equation (4.72) and the dithering signal on the input is given by Equation (4.74) with $B = 0.2$. DSR is used with the parameters $n = 2$, $g = 0$, $L = 11$ and $J = 12$. The mean and standard deviation of the parameters in the parameter vector are listed in Table 4.15 and Table 4.16 for DSR and PEM, respectively. All the estimates are unbiased. The variance on the estimates of b_{11} and b_{21} is reduced compared to when a sinusoid signal is used as dithering signal on the input, Figure 4.23.

4.7 DSR with two sets of horizons

In Section 4.6.1 a sinusoid signal was used as a dithering signal in the reference. When an appropriate frequency was chosen in the dithering signal the eigenvalue estimates were unbiased if the identification horizon, L, was chosen large enough. Unfortunately the estimates of the parameters in the matrix B in the state space model were biased and therefore the estimates of the zeros were biased too. It is

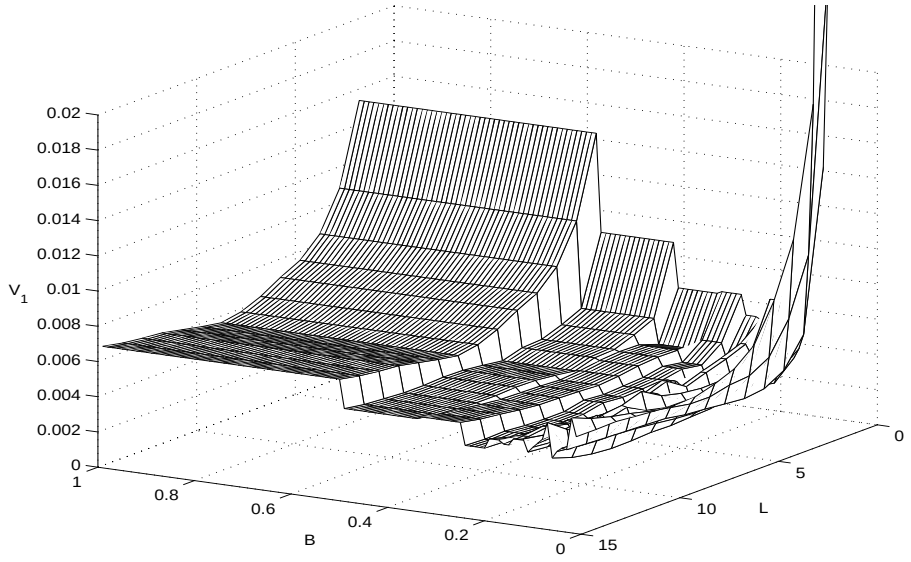

Figure 4.25: The Squared Eigenvalue Error criterion, V^1, as a function of B and L when using a constant reference, $r_k^5 = 1$, and the input superposed a PRBS as a dithering signal generated from $dU^2 = 6.25 \cdot \text{idinput}(N,'prbs',[0\ \ B])$ in Example 2, Section 3.2, and DSR with $n = 2$, $g = 0$ and $J = L + 1$ for identification.

well known that in open loop cases with a low level of persistent excitation on the input signal it is favourable to choose the identification, L, as small as possible. Combining this knowledge with the simulation results from Section 4.6.1 it is interesting to investigate if there will be an improvement in the estimation of the zeros if DSR is implemented with two sets of horizons. Long horizons for the identification of eigenvalues and short horizons for the identification of zeros.

For simplicity the method is named DSR_2LJ. One set of horizons, L_e and J_e, is used for identification of the extended observability matrix and therefore the matrices A and D. The other set of horizons, L_z and J_z, is used for identification of matrix B and will have an effect on the identification of the zeros. This

Table 4.15: Mean and standard deviation from the Monte Carlo simulation of Example 2, Section 3.2, using a constant reference, $r_k^5 = 1$, and the input superposed a PRBS as a dithering signal generated from $dU^2 = 6.25 \cdot \text{idinput}(N,'prbs',[0\ \ B])$, with $B = 0.2$, and using DSR with $n = 2$, $g = 0$, $L = 12$ and $J = 13$ for identification

DSR	a_{21}	a_{22}	b_{11}	b_{21}
Mean	-0.7772	1.7613	-0.0010	-0.0005
Std	0.0134	0.0136	0.0001	0.0000

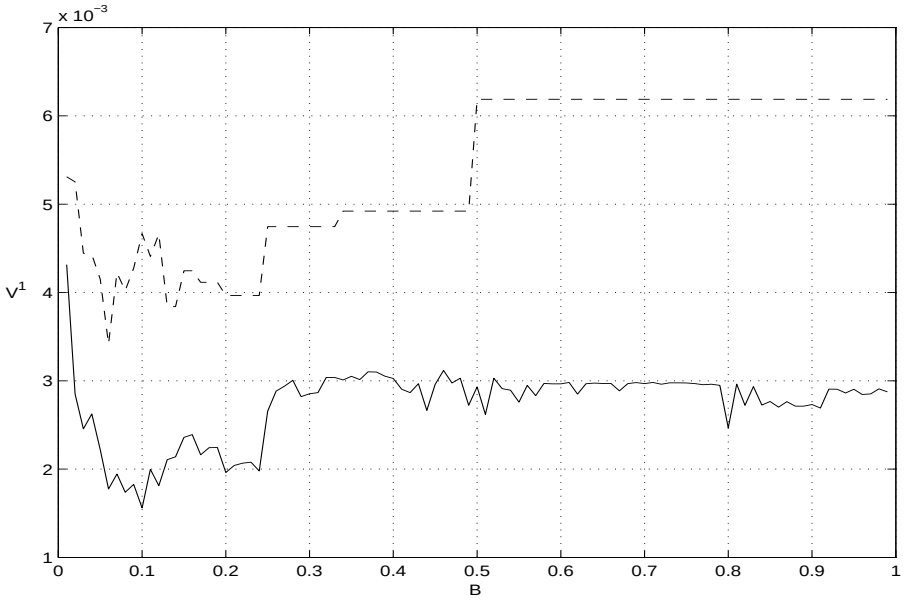

Figure 4.26: The Squared Eigenvalue Error criterion, V^1, as a function of B, when using a constant reference, $r_k^5 = 1$, and the input superposed a PRBS as a dithering signal generated from $dU^2 = 6.25 \cdot \text{idinput}(N,'prbs',[0\ \ B])$ in Example 2, Section 3.2, and the estimates from PEM with default parameters and $n_k = 1$, and DSR with $n = 2$, $g = 0$, $L = 11$ and $J = 12$. PEM is illustrated by the solid line and DSR is illustrated by the broken line.

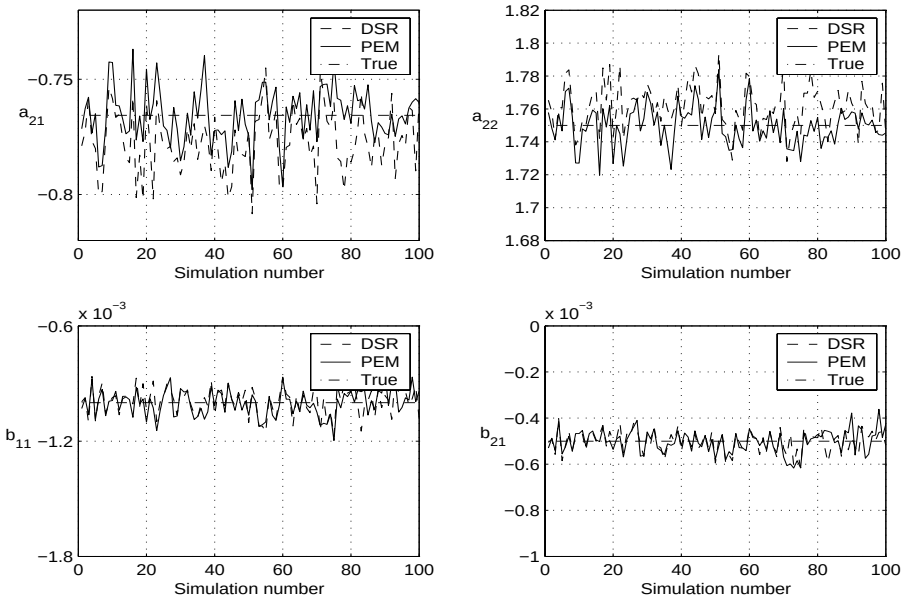

Figure 4.27: The parameter vector, Equation (4.69), as a function of the runs in the Monte Carlo simulation of Example 2, Section 3.2, using a constant reference, $r_k^5 = 1$, and the input superposed a PRBS as a dithering signal generated from $dU^2 = 6.25 \cdot \text{idinput}(N,'prbs',[0\ B])$, with $B = 0.2$, using DSR with $n = 2$, $g = 0$, $L = 11$ and $J = 12$ and PEM with default parameters and $n_k = 1$ for identification. PEM is illustrated by the solid line, DSR is illustrated by the broken line and the true value is illustrated by the dash dotted line.

Table 4.16: Mean and standard deviation from the Monte Carlo simulation of Example 2, Section 3.2, using a constant reference, $r_k^5 = 1$, and the input superposed a PRBS as a dithering signal generated from $dU^2 = 6.25 \cdot \text{idinput}(N,' prbs', [0\ B])$, with $B = 0.2$, and using PEM with default parameters and $n_k = 1$ for identification.

PEM	a_{21}	a_{22}	b_{11}	b_{21}
Mean	-0.7655	1.7496	-0.0010	-0.0005
Std	0.0117	0.0119	0.0001	0.0000

algorithm needs approximately twice as many computations as the original DSR because two sets of data matrices have to be defined and a large amount of the computations have to be performed twice.

4.7.1 The estimation of the parameters in matrix B as a function of the identification horizon of the zeros

In order to evaluate if the use of two sets of horizons in the algorithm named DSR_2LJ leads to improved estimates of the zeros the ability to estimate the zeros of the system presented in Section 3.2 has to be considered as a function of the horizons, L_z and J_z, used for identification of the matrix B. When using a constant reference superposed a sinusoid signal as the dithering signal, Equation (4.67) with $\omega = 0.725$, the estimation of the zeros is biased if DSR is used with the horizons L and J needed to get unbiased eigenvalue estimates. Figure 4.28 shows the Squared Zero Error criterion, V^2, as a function of the identification horizon, L_z, when $J_z = L_z + 1$ and a constant reference superposed a sinusoid signal is used as a dithering signal, Equation (4.67) with $\omega = 0.725$, using DSR_2LJ with $n = 2$, $g = 0$, $Le = 11$ and $Je = 12$.

The assumption that the horizon, L_z, for identification of zeros has to be chosen as small as possible when inputs with a low level of persistent excitation are used is wrong. Choosing the horizon L_z smaller than L_e gives an improvement in the estimation of the zeros.

4.7.2 Alternative closed loop quality measures

In the previous sections it has not been focused directly on the estimation of zeros, rather indirectly by considering the parameters to be estimated when the state space model of the system is transformed to observable canonical form. The parameters to be estimated can be collected in the parameter vector, Equation (4.69), with the true value given by Equation (4.70).

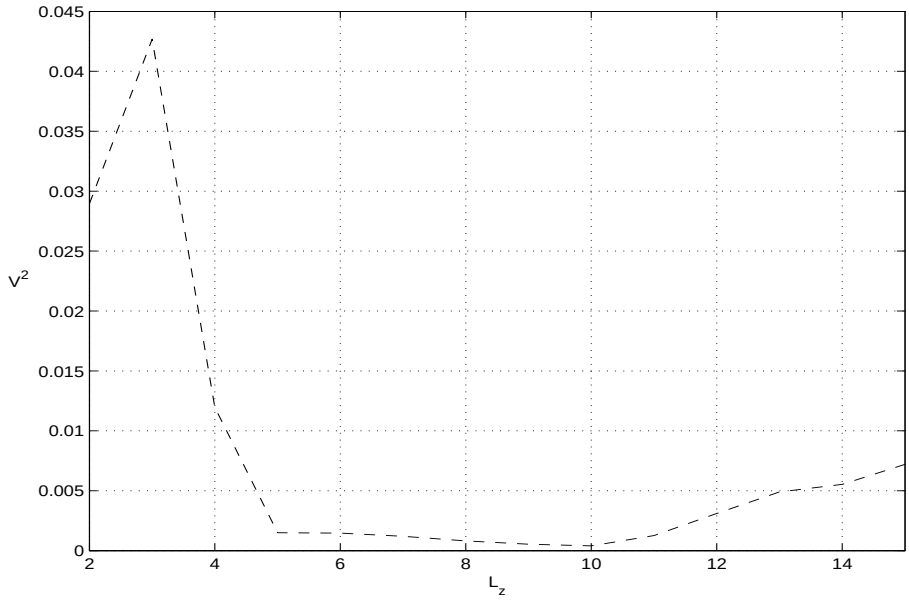

Figure 4.28: The Squared Zero Error criterion, V^2, as a function of the identification horizon, L_z, when using a constant reference superposed a sinusoid signal as the dithering signal, $r_k^3 = 1 + 0.1 \cdot \sin(\omega k)$ with $\omega = 0.725$ in Example 2, Section 3.2, using DSR_2LJ with $n = 2$, $g = 0$, $Le = 11$ and $Je = 12$.

In Figure 4.29 the estimated parameters in the parameter vector from DSR and DSR_2LJ are compared as a function of the 100 runs in the Monte Carlo simulation. The reference is given by Equation (4.67) with $\omega = 0.725$. DSR is used with the parameters $n = 2$, $g = 0$, $L = 11$ and $J = 12$. DSR_2LJ is used with the parameters $n = 2$, $g = 0$, $L_e = 11$, $J_e = 12$, $L_z = 6$ and $J_z = 7$.

Using DSR_2LJ for identification, the estimated parameters b_{11} and b_{21} are unbiased and therefore also the zeros, if the eigenvalues estimates are unbiased. The estimates of the zeros are not unbiased when DSR is used. The parameter estimates of a_{21} and a_{22} are the same in both methods due to the implementation of DSR_2LJ. The mean and standard deviation of the parameters in the parameter vector are listed in Table 4.3 and Table 4.17 for DSR and DSR_2LJ, respectively.

The simulations so far indicate that the DSR_2LJ method is a good alternative for closed loop data sets. But unfortunately when the level of persistent excitation of the dithering signal in the reference is increased the estimate of the parameters in the matrix B is biased, Figure 4.30.

This indicates that the choice L_z depends on the dithering signal. Figure 4.31 shows the Squared Zero Error criterion, V^2, as a function of the identification horizon, L_z, when $J_z = L_z + 1$ and a constant reference superposed a PRBS used

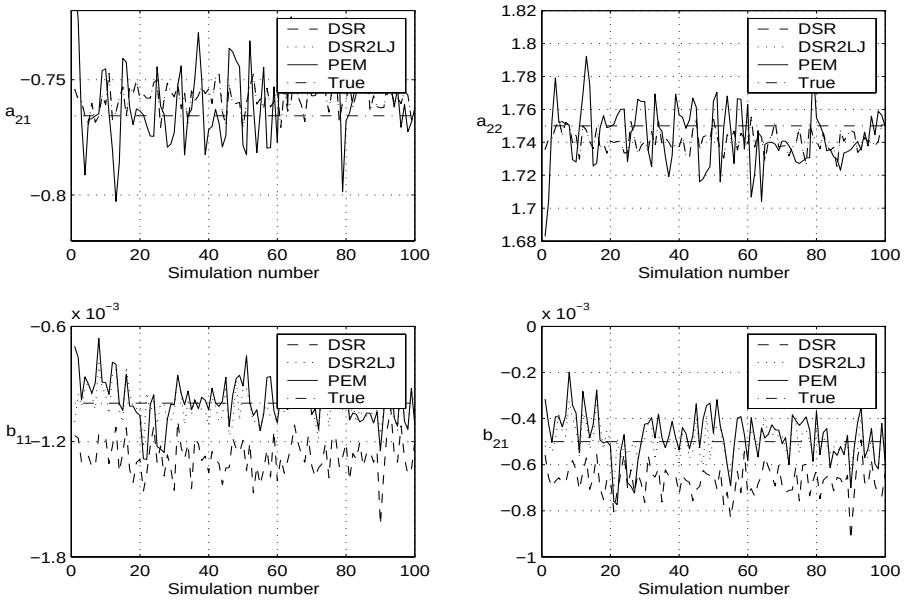

Figure 4.29: The parameter vector, Equation (4.69), as a function of the runs in the Monte Carlo simulation of Example 2, Section 3.2, using a constant reference superposed a sinusoid signal as the dithering signal, $r_k^3 = 1 + 0.1 \cdot \sin(\omega k)$ with $\omega = 0.725$. DSR is used with the parameters $n = 2$, $g = 0$, $L = 11$ and $J = 12$. DSR_2LJ is used with the parameters $n = 2$, $g = 0$, $L_e = 11$, $J_e = 12$, $L_z = 6$ and $J_z = 7$. PEM is used with default parameters and $n_k = 1$ for identification. PEM is illustrated by the solid line, DSR is illustrated by the broken line, DSR_2LJ is illustrated by the dotted line and the true value is illustrated by the dash dotted line.

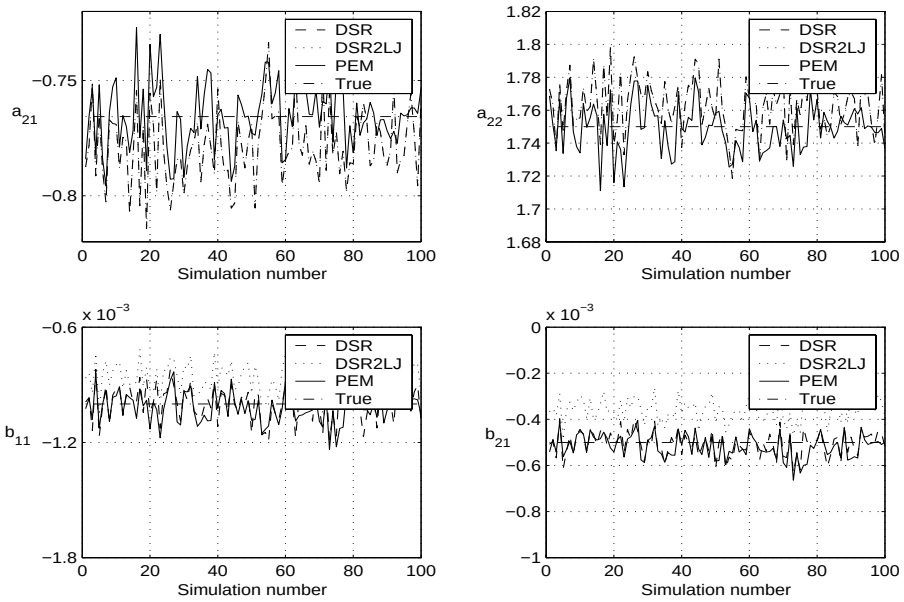

Figure 4.30: The parameter vector, Equation (4.69), as a function of the runs in the Monte Carlo simulation of Example 2, Section 3.2, using a constant reference superposed a PRBS as the dithering signal generated from $R^4 = \text{ones}(N,1) + 0.1 \cdot \text{idinput}(N,'prbs',[0\ B])$ with $B = 0.2$. DSR is used with the parameters $n = 2$, $g = 0$, $L = 11$ and $J = 12$. DSR_2LJ is used with the parameters $n = 2$, $g = 0$, $L_e = 11$, $J_e = 12$, $L_z = 6$ and $J_z = 7$. PEM is used with default parameters and $n_k = 1$ for identification. PEM is illustrated by the solid line, DSR is illustrated by the broken line, DSR_2LJ is illustrated by the dotted line and the true value is illustrated by the dash dotted line.

Table 4.17: Mean and standard deviation from the Monte Carlo simulation of Example 2, Section 3.2, using a constant reference superposed a sinusoid signal as the dithering signal, $r_k^3 = 1 + 0.1 \cdot \sin(\omega k)$ with $\omega = 0.725$, and using DSR_2LJ with the parameters $n = 2$, $g = 0$, $L_e = 11$, $J_e = 12$, $L_z = 6$ and $J_z = 7$ for identification

DSR	a_{21}	a_{22}	b_{11}	b_{21}
Mean	-0.7580	1.7412	-0.0010	-0.0005
Std	0.0062	0.0066	0.0001	0.0001

as dithering signal, Equation (4.71) with $B = 0.2$, using DSR_2LJ with $n = 2$, $g = 0$, $Le = 11$ and $Je = 12$.

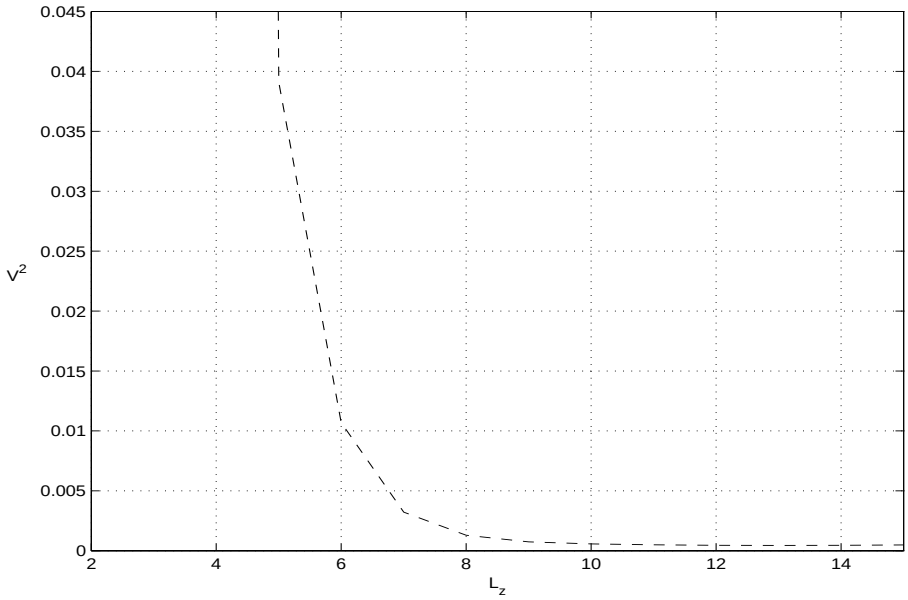

Figure 4.31: The Squared Zero Error criterion, V^2, as a function of the identification horizon, L_z, when using a constant reference superposed a PRBS as the dithering signal generated from $R^4 = \text{ones}(N, 1) + 0.1 \cdot \text{idinput}(N, 'prbs', [0\ B])$ with $B = 0.2$ in Example 2 Section 3.2, using DSR_2LJ with $n = 2$, $g = 0$, $Le = 11$ and $Je = 12$.

Comparing Figure 4.31 with Figure 4.28 shows that when a dithering signal with high order of persistent is used in the reference the lower limit of the choice of L_z is higher than when a dithering signal with low order of persistent excitation is used. This is the opposite of what is the intuitive choice, than when the level of persistent excitation is reduced the horizon can be reduced. This means that the method can be difficult to use on real data.

Chapter 5

Modification of the control loop

Consider once more the reason for the problems that can occur when applying subspace identification (SID) algorithms for direct identification of closed loop data. Future inputs and the noise on the outputs are assumed to be uncorrelated. If future inputs and the noise on the outputs are correlated the projection of the future outputs onto the future inputs will cause a bias. The simulations performed in Chapter 4 have shown that when an appropriate dithering signal is chosen the eigenvalue estimates are unbiased if the identification horizon, L, is chosen large enough. Identification of zeros is a harder task. Two solutions to help reduce the bias in the zeros was found. An increase in the level of persistent excitation of the dithering signal helped to reduce the bias in the zeros. Using a dithering signal on the input instead of the reference also helped to reduce the bias.

Another approach to reduce the bias problem will be considered in this chapter. Here the closed loop will be modified to either reduce the noise in the feedback or make the noise through the feedback uncorrelated to the noise on the output. Di Ruscio (2003a) has already shown that using a filter in the feedback loop is a solution to reduce the bias on the estimated eigenvalues from SID algorithms caused by the error term that occur when the future inputs are correlated with the future noise on the output.

In this section the effect of using different types of filters in the feedback loop will be investigated. The example used in Section 4.4, which is introduced in Section 3.1, will be used in this section. DSR will be compared to PEM and N4SID. The system order $n = 2$ and the fact that the matrix E is the zero matrix is assumed known. PEM and N4SID are used with default parameters and $nk = 1$. The parameter L chosen in DSR is the value which minimizes the Squared Eigenvalue Error criterion, V^1, when $J = L + 1$ and $g = 0$. Monte Carlo simulations with 100 runs will be performed in each case. Focus will be placed on the following properties of the estimates: the eigenvalues of the system matrix A,

$\lambda(A)$, the deterministic steady state gain $H^d(1)$, the eigenvalues of the Kalman filter $\lambda(A - KD)$ and the stochastic steady state gain $H^s(1)$.

5.1 Feedback from the noise free output

An intuitive thought is that an optimal filter will remove all the noise on the output. This means that the measurement equation, Equation (2.14), in the Kalman filter in innovation form is replaced by

$$y_k = Dx_k. \tag{5.1}$$

In this section the feedback to the PI-controller will come from the "filtered" output found by Equation (5.1). The system will be identified from the input and output data as before, and the system will still be affected by both process noise and measurement noise.

When using the "low frequent" PRBS, shown in Figure 4.3 page 33, as reference the eigenvalue estimates from DSR shown in Figure 4.4, page 34, are biased. The estimates from PEM, DSR and N4SID when the feedback to the regulator is calculated from Equation (5.1) are shown in Figure 5.1. PEM and N4SID are used with default parameters and $nk = 1$. DSR is used with $g = 0$, $L = 4$ and $J = 5$.

In Figure 5.1 the bias is so small that the estimates can be considered as unbiased. The estimates from DSR are comparable to the estimates from PEM. The estimates from N4SID have considerably larger variance than the ones from PEM and DSR. To investigate what happens when the noise is increased the simulation is repeated with 50 times larger noise variance. Figure 5.2 shows the estimates when PEM and N4SID are used with default parameters and $nk = 1$. DSR is used with $g = 0$, $L = 13$ and $J = 14$.

Now both the bias and variance of the eigenvalue estimates from all the methods are considerable. This should not happen according to the thought that the output used in the feedback is noise free. This means that the problem is more complex than initially assumed. Factors that may have influence are that there is still noise on the outputs which is used for identification. It is necessary to consider what happens if a reference signal with higher order of persistent excitation is used. Figure 5.3 shows a more "high frequent" PRBS, r_k^6, used as reference and the corresponding input, u_k, and output, y_k, when the system is operating in a closed loop without a filter in the feedback.

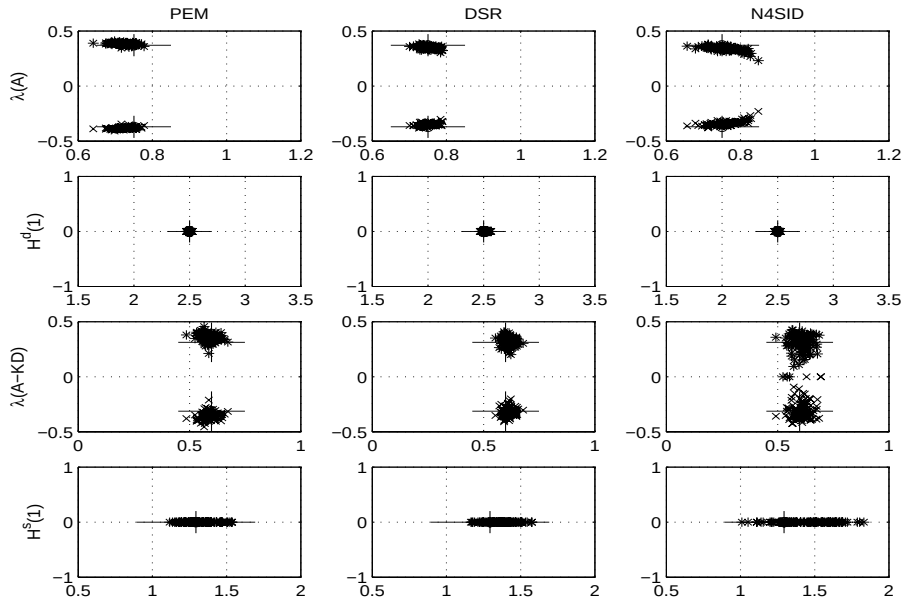

Figure 5.1: Estimates using low frequent PRBS, r_k^1 page 33, as a reference signal and noise-free feedback in the closed loop system Example 1, Section 3.1. PEM and N4SID are used with default parameters and $nk = 1$. DSR is used with $g = 0$, $L = 4$ and $J = 5$.

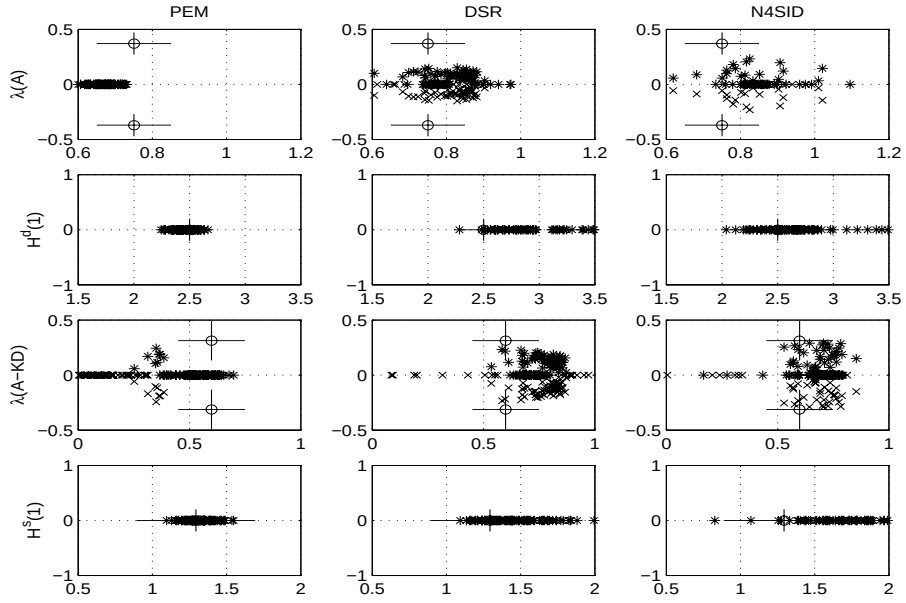

Figure 5.2: Estimates using low frequent PRBS, r_k^1 page 33, as a reference signal and noise-free feedback in the closed loop system, Example 1, Section 3.1, when noise variance is increased by 50 times. PEM and N4SID are used with default parameters and $nk = 1$. DSR is used with $g = 0$, $L = 13$ and $J = 14$.

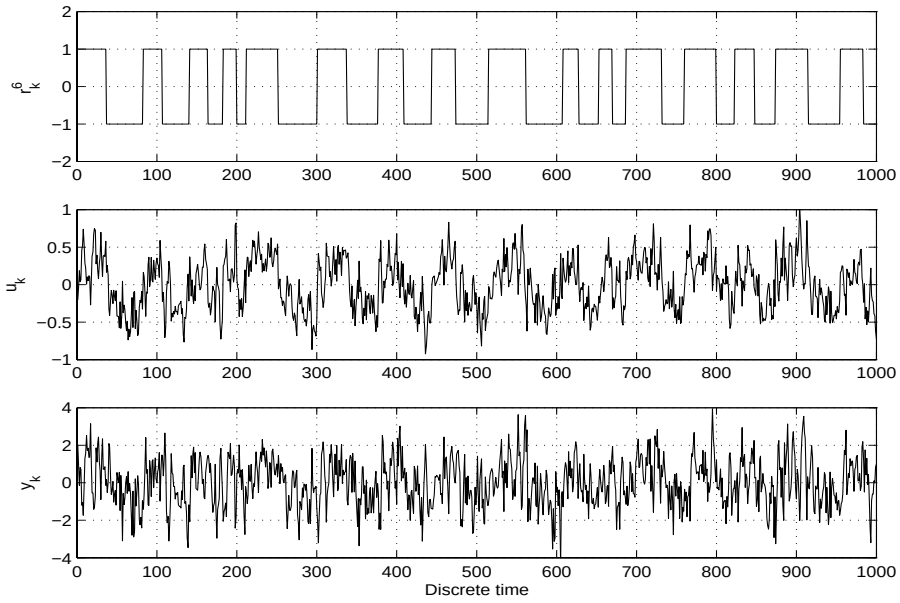

Figure 5.3: Reference r_k^6 and corresponding input, u_k, and output, y_k, when the system in Example 1, Section 3.1, is operating in a closed loop without a filter in the feedback and the noise variance is increased 50 times

Figure 5.4 shows the estimates when r_k^6 is used as a reference. PEM and N4SID are used with default parameters and $nk = 1$. DSR is used with $g = 0$, $L = 4$ and $J = 5$.

It must be noted that all three methods give biased estimates here, but N4SID has the largest variance on the estimates. The bias is reduced when r_k^6 is used as the reference signal instead of r_k^1, page 33, but it does not give unbiased estimates. It is no surprise that an increase in the order of persistent excitation of the reference signal can compensate for an increase in the noise level.

Comparing Figure 5.4 and Figure 5.5 where the feedback is from a noise-free output and the actual output, respectively, it has to be noted that for PEM it is not favourable to have the feedback from the noise free output. This coincides with the fact that for PEM the variance on the parameter estimates from direct closed loop identification is reduced when the variance on the measurement noise is increased, Forssell et al. (1997).

It is also important to be aware that when using the noise-free feedback the noise is only removed from the feedback and not from the output data which is used for modelling. This means that the noise level on the output is the same but the correlation between the input and the noise on the output is removed. Another effect from using the noise-free feedback is that the noise on the output does not

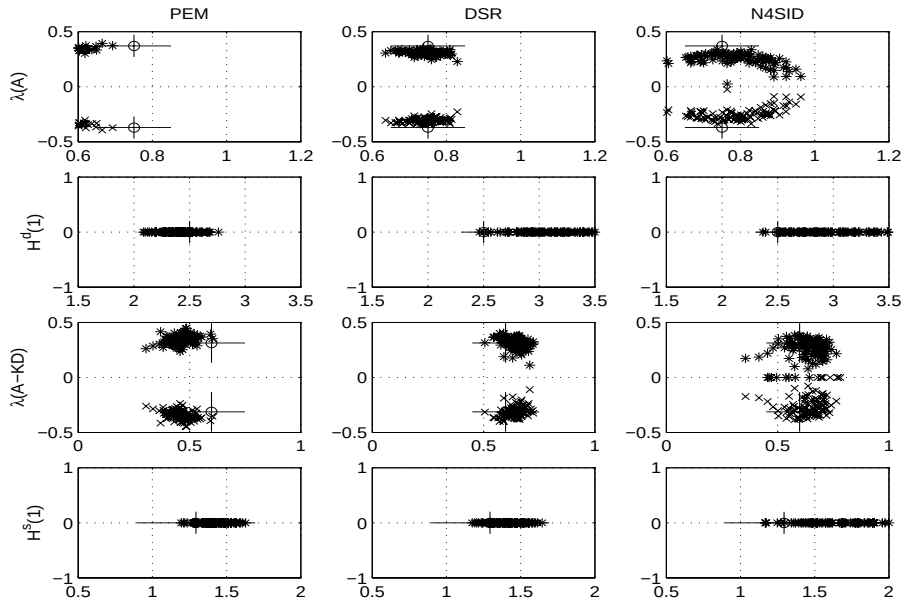

Figure 5.4: Estimates using high frequent PRBS, r_k^6, as a reference signal and noise-free feedback in the closed loop system, Example 1, Section 3.1, when the noise variance is increased 50 times. PEM and N4SID are used with default parameters and $nk = 1$. DSR is used with $g = 0$, $L = 4$ and $J = 5$.

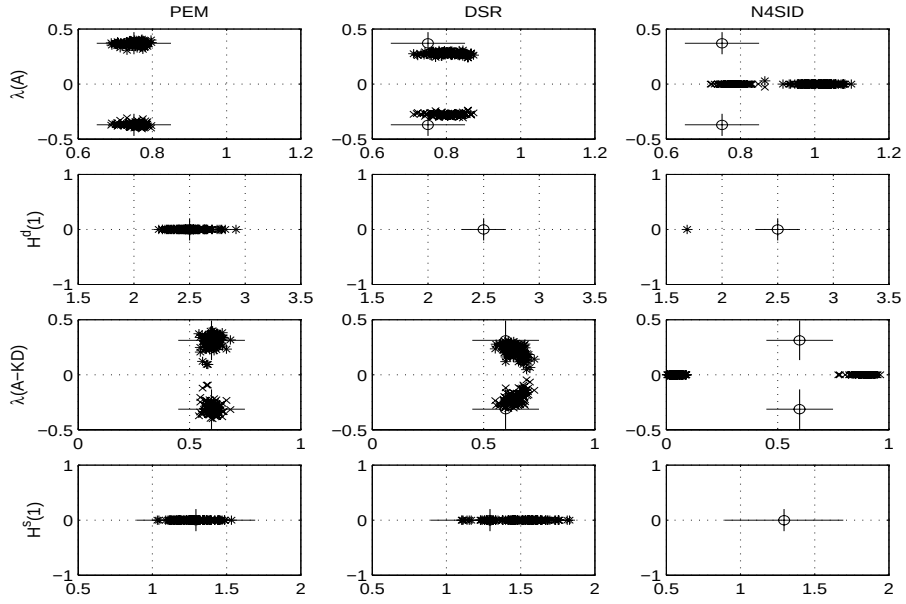

Figure 5.5: Estimates using high frequent PRBS, r_k^6, as a reference signal when the system is operating in closed loop, Example 1, Section 3.1, without a filter in the feedback when the noise variance is increased 50 times. PEM and N4SID are used with default parameters and $nk = 1$. DSR is used with $g = 0$, $L = 4$ and $J = 5$.

help to excite the input.

5.2 Feedback from a customized filter

The intuitive thought presented in Section 5.1 that an optimal filter will remove all the noise on the output has to be modified. Let us use the fact that for PEM the variance on the parameter estimates from direct closed loop identification is reduced when the variance on the measurement noise is increased, Forssell et al. (1997). Assume that some of the reason for this is that the noise helps to excite the input. For SID algorithms future inputs and the noise on the outputs are assumed to be uncorrelated, therefore there is no direct connection. But if the noise level is reduced sufficiently or the future inputs and the noise on the outputs are sufficiently uncorrelated, the noise on the output will help to excite the input without interfering with the assumptions for the SID algorithms. Therefore a 1^{st} order low-pass filter is introduced in the feedback. The question is therefore: Will the choice of filter constant in the low-pass filter in the feedback reduce the noise on the output, or make the input sufficiently uncorrelated with the noise on the output, in a suitable way such that the bias is reduced or preferably removed? The filter used is a 1^{st} order low-pass filter given by

$$\bar{y}_{k+1} = \bar{y}_k + K_f(y_k - \bar{y}_k). \tag{5.2}$$

The parameter L chosen in DSR and the filter constant K_f is the values which minimize the Squared Eigenvalue Error criterion, V^1, when $J = L + 1$ and $g = 0$.

When using the "low frequent" PRBS, shown in Figure 4.3 page 33, as reference the eigenvalue estimates from DSR shown in Figure 4.4, page 34, are biased. When using a "noise free" feedback the bias is not observable with visual inspection, Figure 5.1. The estimates from PEM, DSR and N4SID when the feedback to the regulator is filtered through a 1^{st} order low-pass filter, Equation (5.2), is shown in Figure 5.6. PEM and N4SID are used with default parameters and $nk = 1$. DSR is used with $g = 0$, $L = 7$ and $J = 8$. The filter constant used is $K_f = 0.05$.

In Figure 5.6 the bias is so small that it is not observable with visual inspection, just like when the noise-free feedback is used, Figure 5.1. The bias and variance of the estimates from DSR is comparable to the estimates from PEM. N4SID also gives unbiased estimates, but the variance is larger. To investigate what happens when the noise is increased the simulation is repeated with 50 times larger noise variance. Figure 5.7 shows the eigenvalue estimates when PEM and N4SID are used with default parameters and $nk = 1$. DSR is used with $g = 0$, $L = 6$ and

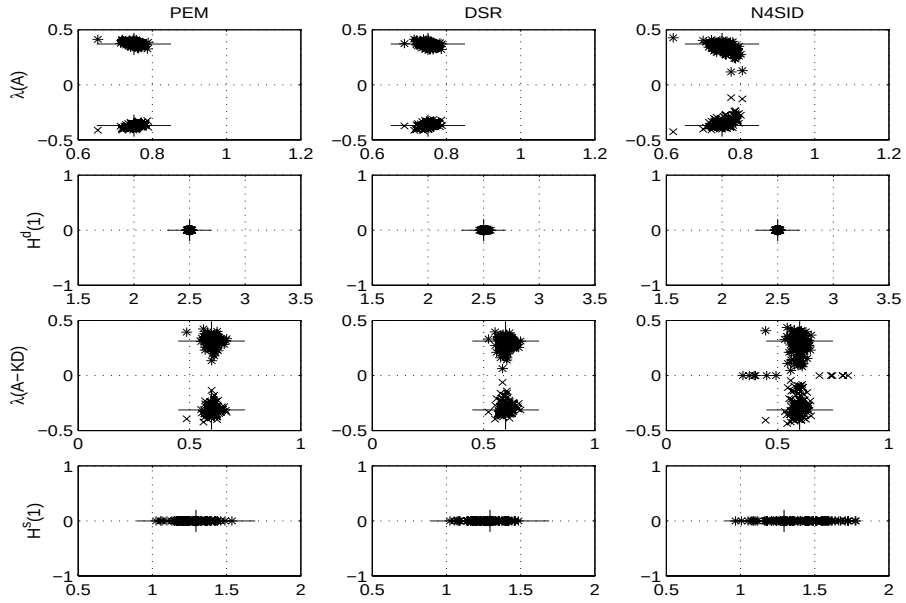

Figure 5.6: Estimates using low frequent PRBS, r_k^1 page 33, as a reference signal and low-pass filtered feedback in the closed loop system, Example 1, Section 3.1. PEM and N4SID are used with default parameters and $nk = 1$. DSR is used with $g = 0$, $L = 7$ and $J = 8$. The filter constant used is $K_f = 0.05$.

$J = 7$. The filter constant is $K_f = 0.05$.

Now the estimates from DSR have a bias, but PEM still provides estimates with a bias which is so small that it is not observable by visual inspection. The estimates from N4SID are now considerably poorer than the ones from DSR. The results when the feedback is filtered through the customized filter, Figure 5.7, is considerable better than when the noise-free output is used as feedback, Figure 5.2, when the same reference and noise level is used. However, it is necessary to consider what happens if a reference signal r_k^6, page 72, with a higher order of persistent excitation is used.

Figure 5.8 shows the estimates when PEM and N4SID are used with default parameters and $nk = 1$. DSR is used with $g = 0$, $L = 6$ and $J = 7$ and the filter constant used is $K_f = 0.05$.

An increase in the order of persistent excitation of the reference signal can compensate for an increase in the noise level. Comparing Figure 5.8 and Figure 5.5 where the feedback is filtered through a filter and the actual output, respectively, it has to be noted that for PEM it is not necessarily favourable to have the feedback from the output filtered. This coincides with the the properties presented by Forssell et al. (1997), that for PEM the variance on the parameter estimates

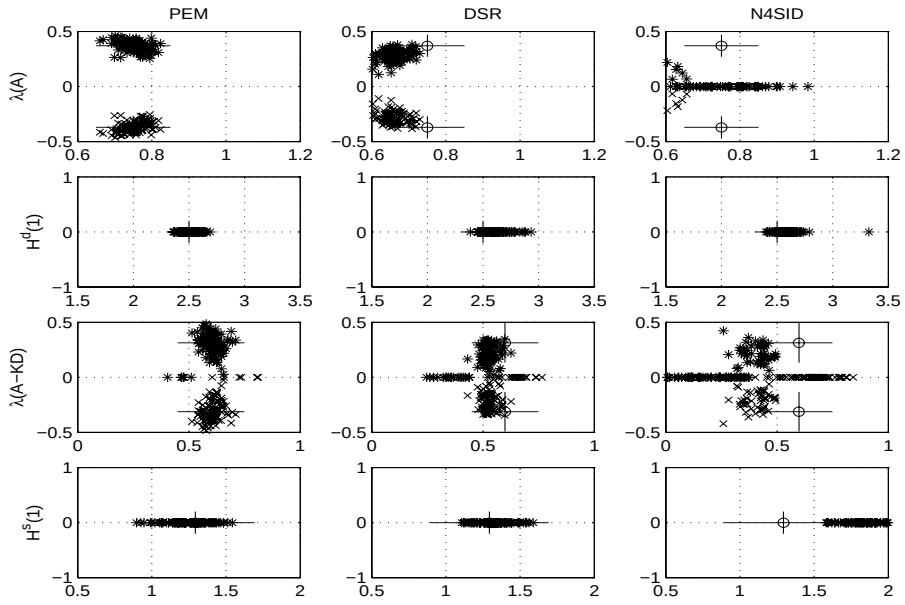

Figure 5.7: Estimates using low frequent PRBS, r_k^1 page 33, as a reference signal and low-pass filtered feedback in the closed loop system, Example 1, Section 3.1, when the noise variance is increased 50 times, PEM and N4SID are used with default parameters and DSR is used with $g = 0$, $L = 6$ and $J = 7$ and the filter constant used is $K_f = 0.05$.

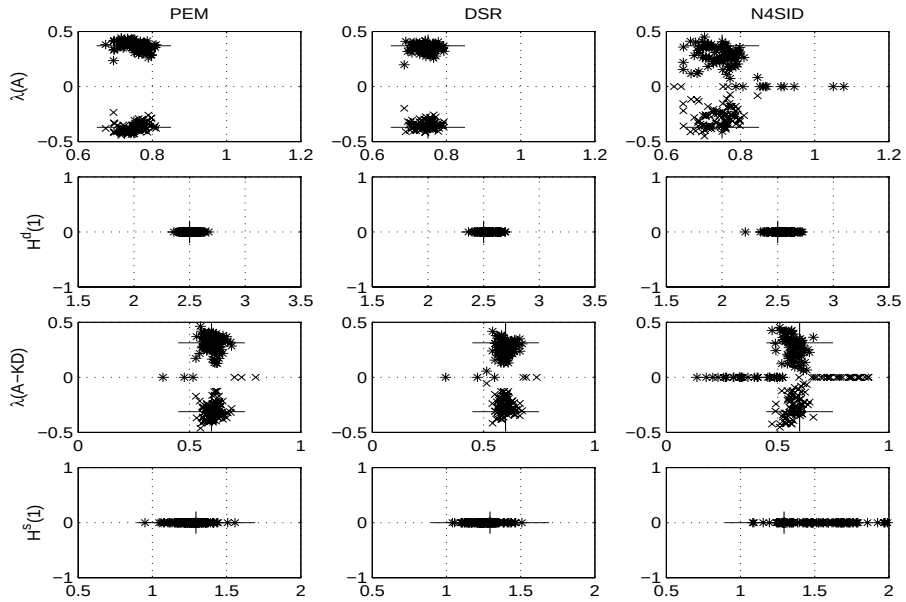

Figure 5.8: Estimates using high frequent PRBS, r_k^6 page 72, as a reference signal and low-pass filtered feedback in the closed loop system, Example 1, Section 3.1, when the noise variance is increased 50 times. PEM and N4SID are used with default parameters and $nk = 1$. DSR is used with $g = 0$, $L = 6$ and $J = 7$. The filter constant used is $K_f = 0.05$.

from direct closed loop is reduced when the variance on the measurement noise is increased.

5.3 Feedback from the Kalman filter output

The Kalman filter is known to be the optimal filter and the minimum variance estimator. Since the use of a 1^{st} order low-pass filter in the feedback in Section 5.2 gave such an improvement in the estimates from the SID algorithms compared to not using a filter, or using a noise free feedback, it is interesting to check if the use of a Kalman filter instead can give even better results.

Figure 5.9 shows the estimates from PEM, DSR and N4SID when the feedback to the regulator is filtered through a Kalman filter and the "low frequent" PRBS, shown in Figure 4.3 page 33, is used as a reference. PEM and N4SID are used with default parameters and $nk = 1$. DSR is used with $g = 0$, $L = 5$ and $J = 6$.

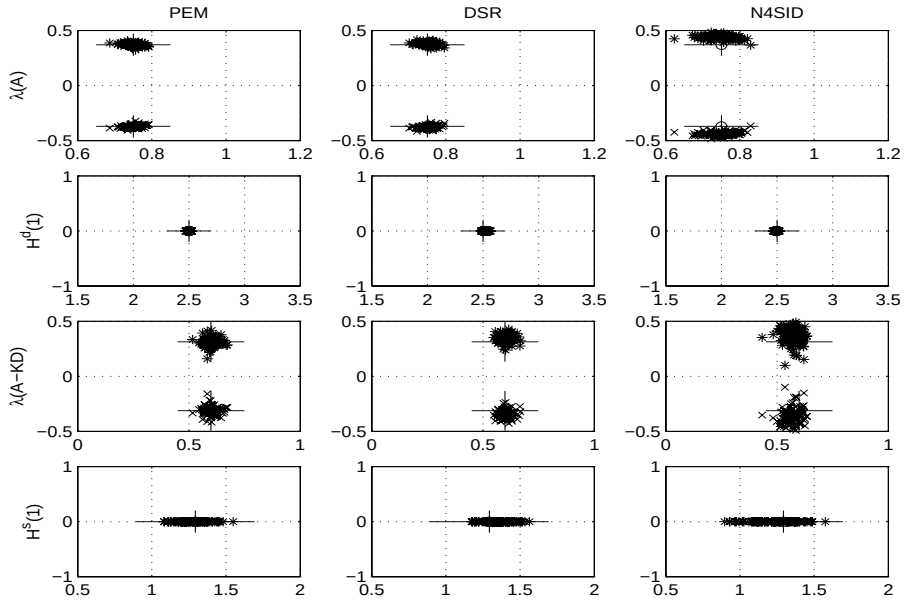

Figure 5.9: Estimates using low frequent PRBS, r_k^1 page 33, as a reference signal and feedback from a Kalman filter in the closed loop system, Example 1, Section 3.1. PEM and N4SID are used with default parameters and $nk = 1$. DSR is used with $g = 0$, $L = 5$ and $J = 6$.

In Figure 5.9 the bias on the estimates from PEM and DSR is so small that it is not observable with visual inspection, just like in Figure 5.1 and Figure 5.6. The exception is N4SID which gives estimates with bias and larger variance than

DSR and PEM. To investigate what happens when the noise is increased the simulation is repeated with 50 times larger noise variance. Figure 5.10 shows the estimates when PEM and N4SID are used with default parameters and $nk = 1$. DSR is used with $g = 0$, $L = 5$ and $J = 6$.

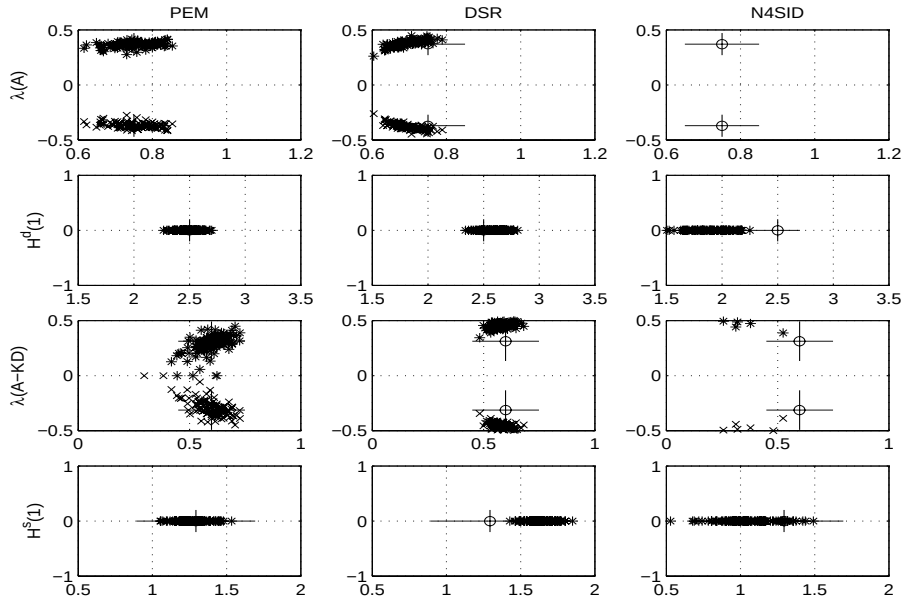

Figure 5.10: Estimates using low frequent PRBS, r_k^1 page 33, as a reference signal and feedback from a Kalman filter in the closed loop system, Example 1, Section 3.1, when the noise variance is increased 50 times. PEM and N4SID are used with default parameters and $nk = 1$. DSR is used with $g = 0$, $L = 5$ and $J = 6$.

The estimates from DSR shown in Figure 5.10 only have a small bias, except the stochastic steady state gain $H^s(1)$ which has got an increase in the bias. All the estimates from DSR have reduced variance compared to the case when the feedback is filtered through a low-pass filter, Figure 5.7. This has to be considered as an improvement compared to the case when the feedback is filtered through a low-pass filter when the same reference and noise are used. When using the Kalman filter in the feedback PEM seems to have a larger bias than DSR, Figure 5.10. Also in this case N4SID gives the poorest results. It is also necessary to consider what happens if a reference signal, r_k^6 page 72, with higher order of persistent excitation is used. Figure 5.11 shows the estimates when PEM and N4SID are used with default parameters and $nk = 1$. DSR is used with $g = 0$, $L = 6$ and $J = 7$.

The variance is considerably reduced in Figure 5.11 where r_k^6 is used as the reference compared to Figure 5.10 where r_k^1 is used as the reference, except of the

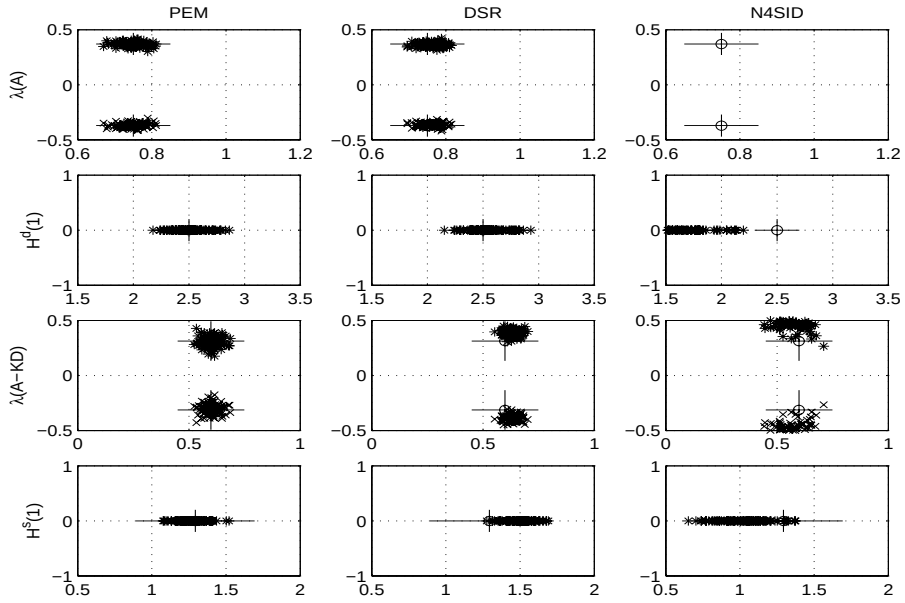

Figure 5.11: Estimates using high frequent PRBS, r_k^6 page 72, as a reference signal and feedback from a Kalman filter in the closed loop system, Example 1, Section 3.1, when the noise variance is increased 50 times. PEM and N4SID are used with default parameters and $nk = 1$. DSR is used with $g = 0$, $L = 6$ and $J = 7$.

deterministic steady state gain $H^d(1)$. Once again, N4SID gives the poorest results. Now the bias on the estimates from both PEM and DSR are so small that the estimates can be considered as unbiased.

The simulations performed in Section 5 showed that using a filter in the feedback to the controller can reduce and in some cases remove the bias when DSR is used for estimation. The optimal filter used in the feedback is not the noise-free output or a 1^{st} order low-pass filter but the Kalman filter. The reason for this is that the Kalman filter gives an estimate of the output which contains the optimal amount of information. There is also reason to believe that measurement noise can lead to an increased excitation of the input of the system and in this way help to reduce bias and variance on the estimates from SID methods, if the input and the noise on the output are uncorrelated.

The simulations done so far using the Kalman filter are based on knowledge of all the system matrices and the noise variance. If these parameters are known it is no use of a performing a system identification procedure. Therefore a Monte Carlo simulation is performed where all the inputs to the calculations in the Kalman filter are reduced by 10%. Figure 5.12 shows the estimates when PEM and N4SID are used with default parameters and $n_k = 1$. DSR is used with $g = 0$, $L = 6$

and $J = 7$.

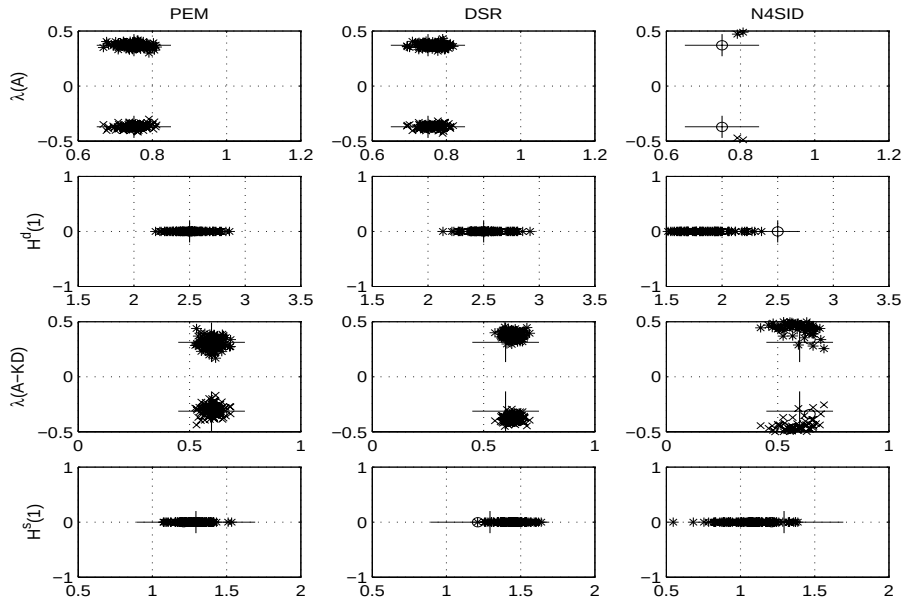

Figure 5.12: Estimates using high frequent PRBS, r_k^6 page 72, as a reference signal and feedback from a Kalman filter, with 10% deviation, in the closed loop system, Example 1, Section 3.1, when noise variance is increased 50 times. PEM and N4SID are used with default parameters and $nk = 1$. DSR is used with $g = 0$, $L = 6$ and $J = 7$.

The estimates shown in Figure 5.12 are comparable to the case where the Kalman filter is correct, Figure 5.11.

5.3.1 Subspace identification and feedback from an analytic point of view

Di Ruscio (2003a) has shown that when the extended observability matrix O_{L+1} is estimated from the column space of the projection matrix $Z_{J|L+1}$ as defined in Equation (4.31) there is an error term in the projection

$$Z_{J|L+1} = O_{L+1} X_J^a + dZ \tag{5.3}$$

where the error term dZ is given by

$$dZ = H_{L+1}^s \left(E_{J|L+1} / \begin{bmatrix} U_{J|L+1} \\ U_{0|J} \\ Y_{0|J} \end{bmatrix} \right) U_{J|L+g}^{\perp}$$

$$\begin{aligned} &= H^s_{L+1} E_{J|L+1} U^{\perp}_{J|L+g} W_p^T (W_p U^{\perp}_{J|L+g} W_p^T)^{-1} W_p U^{\perp}_{J|L+g} \qquad (5.4) \\ &\approx -H^s_{L+1} E_{J|L+1} / U_{J|L+g} W_p^T (W_p U^{\perp}_{J|L+g} W_p^T)^{-1} W_p U^{\perp}_{J|L+g} \end{aligned}$$

where

$$W_p = \begin{bmatrix} U_{0|J} \\ Y_{0|J} \end{bmatrix}. \qquad (5.5)$$

In the last expression in Equation (5.4) it is made use of the fact that $E_{J|L+1} W_p^T / K \approx 0$ when the number of columns K tends towards infinity. The remaining projection in the error term is then $E_{J|L+1}/U_{J|L+g}$. This term is approximately zero for open loop problems. The term $E_{J|L+1}/U_{J|L+g}$ in some closed loop problems is non-zero and causes biased estimates. This is the problem in feedback systems where the control is directly proportional to the innovation noise. It is stated by Di Ruscio (2003a) that *It is believed that SID of the 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 innovation noise. There are no problems 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 Equation (4.25) small. The statements by Di Ruscio are exactly what is observed here. There is no difference if there is feedback from the "noise-free" state $\bar{x}$ in the Kalman filter or the the "noise-free" output $\bar{y} = D\bar{x}$ of the Kalman filter from the SID point of view. From the SID point of view the goal is to have a feedback without innovation noise. This is also the reason why the simulations where the Kalman filter is not correct works so well.

5.4 Feedback from a Kalman filter estimated by DSR

In Section 5.3 a Kalman filter was used in the feedback to reduce, and preferably eliminate, the bias when DSR is used for closed loop system identification. A simulation study was also performed where all the inputs to the calculation of the Kalman filter were reduced by 10%. The results were comparable to when the correct Kalman filter was used. This leads to the idea of a new three-step algorithm based on the DSR algorithm and the Kalman filter properties, Jazwinski (1970). The results in this section are published in Modeling, Identification and Control, Nilsen and Di Ruscio (2006).

By definition the minimum variance estimator (Kalman filter) minimizes the error norm $E((x_k - \overline{x}_k)(x_k - \overline{x}_k)^T)$. The Orthogonal Projection Lemma, Jazwinski

(1970), gives a condition that is equivalent to the minimization of the error norm. The Orthogonal Projection Lemma states that the error is orthogonal to the approximation space Y_k. In the Kalman filter problem the approximation space is $Y_k = \{space\ spanned\ by\ y_1, \ldots, y_k\}$. This means that

$$E(\bar{y}_k \varepsilon_k^T) = 0. \tag{5.6}$$

Therefore we want to suggest a new closed loop subspace identification algorithm based on feedback from a Kalman filter found by DSR from closed loop data.

Algorithm 5.4.1

- *Step 1. Identification of the Kalman filter using DSR*
- *Step 2. Implementation of the Kalman filter identified in Step 1*
- *Step 3. Identification of an unbiased model using DSR*

The model found by DSR in Step 1 may have a bias, when the system is operating in a closed loop and there is noise present. The output, $\bar{y}_k$, from a non-optimal Kalman filter will have some level of correlation to the innovation, ε_k. The idea is that the Kalman filter found by DSR in Step 1 will give an output, $\bar{y}_k$, which is sufficiently uncorrelated with the noise on the output of the actual process, and in this way reduce or eliminate the bias problem.

A block diagram of the algorithm is shown in Figure 5.13. In Step 1 the switch in the figure is in position 1. In all other cases the switch is in position 2.

5.4.1 Single Input Single Output simulation example

Example 1 introduced in Section 3.1 is still used as an example. The "low frequent" PRBS, r_k^1 shown in Figure 4.3 page 33, is used as a reference. The results from direct open loop closed loop identification is shown in Figure 5.14, where DSR is used with $g = 0$, $L = 5$ and $J = 6$. PEM and N4SID are used with default parameters and $nk = 1$. Figure 5.14 shows the eigenvalues of the system $\lambda(A)$, the deterministic steady state gain $H^d(1)$, the eigenvalues of the Kalman filter $\lambda(A - KD)$ and the stochastic steady state gain $H^s(1)$, which are the properties of the estimates that will be focused on in this section.

The estimates from PEM are unbiased and the estimates from the SID algorithms are biased. Comparing the SID algorithms it is clear that the estimates from N4SID have a much larger bias than the estimates from DSR. Therefore it is not advisable to use N4SID in Step 1 in the algorithm introduced in Section 5.4.

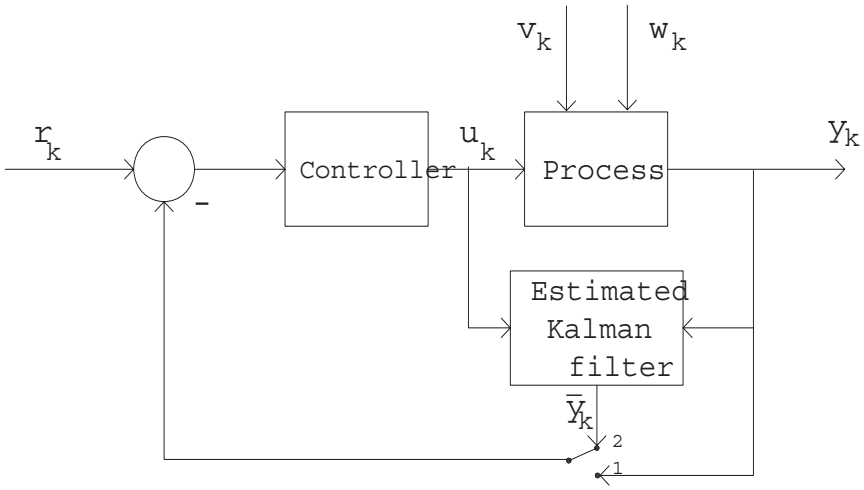

Figure 5.13: Block diagram of the algorithm. The switch is in position 1 in Step 1, else the switch is in position 2.

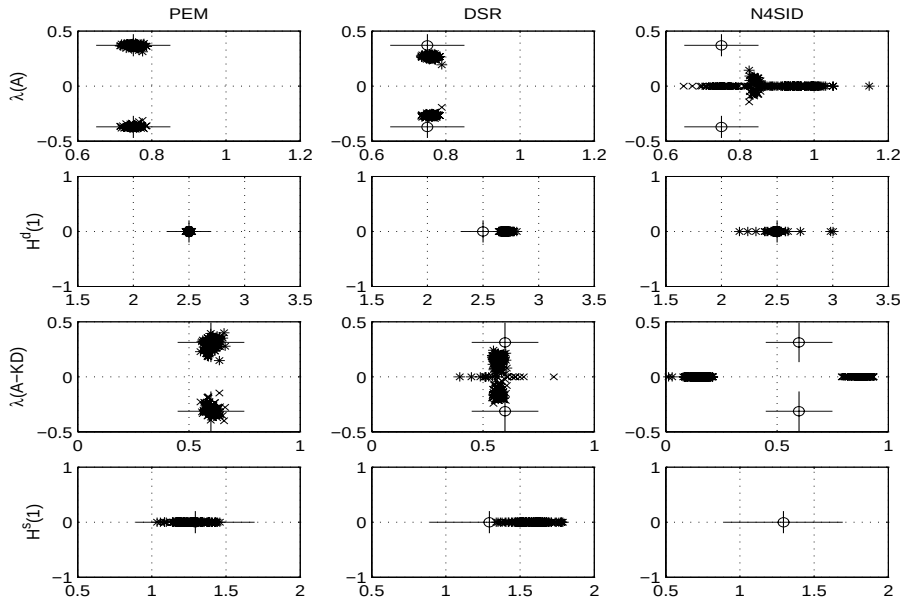

Figure 5.14: Estimates from closed loop Monte Carlo simulation using r_k^1, page 33, as a reference in Example 1, Section 3.1. PEM and N4SID are used with default parameters and $nk = 1$. DSR is used with $g = 0$, $L = 5$ and $J = 6$.

In order to evaluate the quality of the algorithm introduced in Section 5.4, Step 1 is performed by a single simulation using r_k^1 as a reference to identify a (biased) model using DSR. The Kalman filter found by DSR with $g = 0$, $L = 5$ and $J = 6$ is given by

$$A = \begin{bmatrix} 0 & 1 \\ -0.6508 & 1.5257 \end{bmatrix}, B = \begin{bmatrix} 0.9134 \\ 0.8175 \end{bmatrix},$$

$$D = \begin{bmatrix} 1 & 0 \end{bmatrix}, K = \begin{bmatrix} 0.3634 & 0.2675 \end{bmatrix}^T. \tag{5.7}$$

The reference r_k^1 is plotted in Figure 5.15 with the corresponding input, u_k, and the output, y_k, for two particular noise realizations with the same noise level as in the previous simulations, to illustrate the effect on the noise level when using feedback filtered through the Kalman filter found by DSR, Equation (5.7).

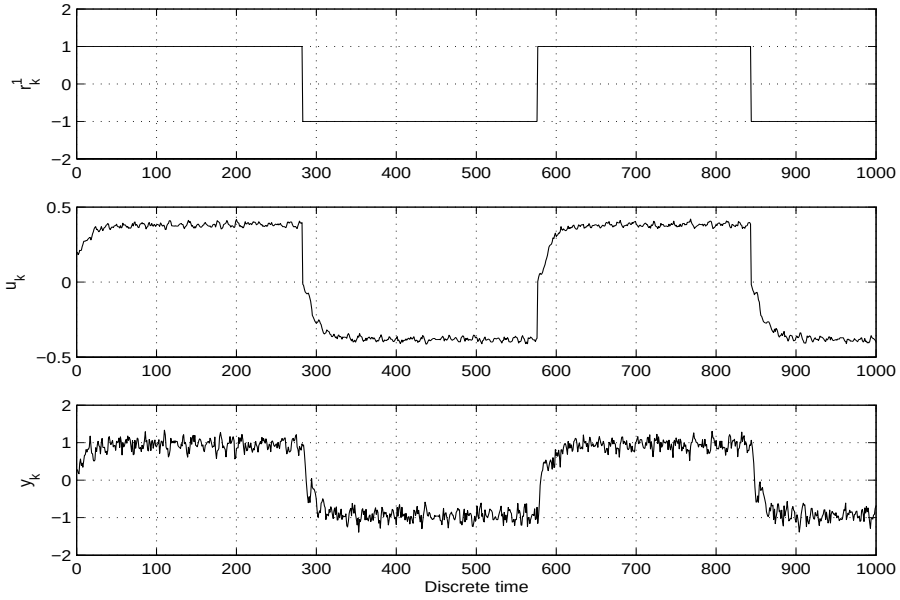

Figure 5.15: The reference signal, r_k^1, with corresponding input, u_k, and output, y_k, for two particular noise realizations, v_k and w_k in Example 1 Section 3.1, when the feedback is filtered through a Kalman filter found by DSR, Equation (5.7), in an initial step. PEM and N4SID are used with default parameters and $nk = 1$. DSR is used with $g = 0$, $L = 5$ and $J = 6$.

The noise level in Figure 5.15 is reduced compared to Figure 4.3. The reason is that the input is a function of the filtered output, but this was not the main goal. The main goal was to generate an input, u_k, which is uncorrelated with the noise

on the output, y_k.

Steps 2 and 3 of the algorithm introduced in Section 5.4 are evaluated by a Monte Carlo simulation with 100 runs with a different noise realization in each run carried out using r_k^1 as a reference signal, when the feedback is filtered through the Kalman filter found by DSR, Equation (5.7), in an initial simulation. As in previous simulations the system order $n = 2$ is assumed known. PEM and N4SID are used with default parameters and $nk = 1$. DSR is used with $g = 0$, $L = 5$ and $J = 6$. Figure 5.16 shows the estimates with no iterations performed to improve the Kalman filter.

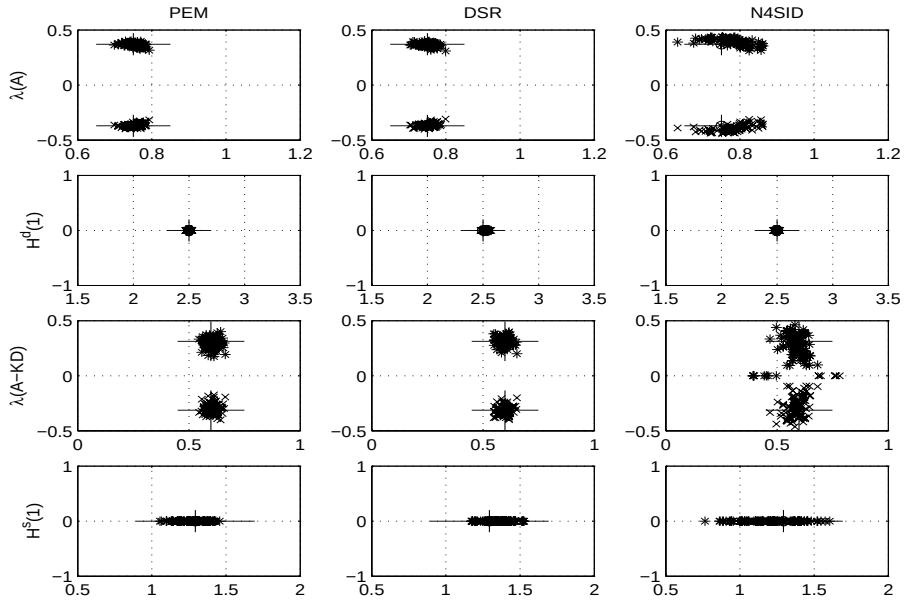

Figure 5.16: Estimates from closed loop Monte Carlo simulation using r_k^1, page 33, as a reference when the feedback is filtered through a Kalman filter found by DSR, Equation (5.7), in an initial step. For direct closed loop system identification PEM and N4SID are used with default parameters and $nk = 1$. DSR is used with $g = 0$, $L = 5$ and $J = 6$.

Now when the feedback is filtered through the Kalman filter found by DSR, Equation (5.7), all the methods give unbiased estimates, but the estimates from N4SID have considerable larger variance than the others. It is quite satisfactory that the estimates from PEM do not have any observable increase in variance when the feedback is filtered through the Kalman filter found by DSR, Figure 5.16, is compared to direct closed loop identification, Figure 5.14. It indicates that the Kalman filter estimated in Step 1 in the algorithm does not have to be very accurate to have the desired effect.

We are pleased to observe that the control function shown in Figure 5.15 is still satisfactory when the feedback is filtered through the Kalman filter found by DSR in the initial step.

Figure 5.9 shows the reference r_k^1 with the corresponding input, u_k, and the output, y_k, for two particular noise realizations with the same noise level as in the previous simulations when the feedback is filtered through the correct Kalman filter. The performance is not significantly better compared to when the feedback is filtered through the Kalman filter found by DSR in an initial step, Figure 5.16. Like in the previous sections the simulation is repeated with 50 times larger noise variance to investigate what happens when the noise level is increased.

Step 1 is performed by a single simulation using r_k^1 as a reference to identify a (biased) model using DSR. The Kalman filter found by DSR with $g = 0$, $L = 6$ and $J = 7$ is given by:

$$A = \begin{bmatrix} 0 & 1 \\ -0.5784 & 1.4607 \end{bmatrix}, B = \begin{bmatrix} 1.0056 \\ 0.9373 \end{bmatrix},$$

$$D = \begin{bmatrix} 1 & 0 \end{bmatrix}, K = \begin{bmatrix} 0.1374 & 0.1291 \end{bmatrix}^T. \tag{5.8}$$

PEM and N4SID are used with default parameters and $nk = 1$. DSR is used with $g = 0$, $L = 6$ and $J = 7$. Figure 5.17 shows the estimates with no iterations performed to improve the Kalman filter.

The increase in noise level leads to an increase in the variance of the estimates. This increase makes it hard to state if a bias is present by visual inspection. Compared to when the correct Kalman filter is used in the feedback, Figure 5.10, the variance is increased when a Kalman filter found by DSR is used in the feedback, Figure 5.17. Here it is also necessary to investigate the effect of using a reference signal, r_k^6 page 72, with a higher order of persistent excitation.

Step 1 is performed by a single simulation using r_k^6, page 72, as a reference to identify a (biased) model using DSR. The Kalman filter found by DSR with $g = 0$, $L = 6$ and $J = 7$ is given by

$$A = \begin{bmatrix} 0 & 1 \\ -0.6102 & 1.4983 \end{bmatrix}, B = \begin{bmatrix} 1.0873 \\ 1.1306 \end{bmatrix},$$

$$D = \begin{bmatrix} 1 & 0 \end{bmatrix}, K = \begin{bmatrix} 0.2694 & 0.2239 \end{bmatrix}^T. \tag{5.9}$$

PEM and N4SID are used with default parameters and $nk = 1$. DSR is used with $g = 0$, $L = 6$ and $J = 7$. Figure 5.18 shows the estimates with no iterations

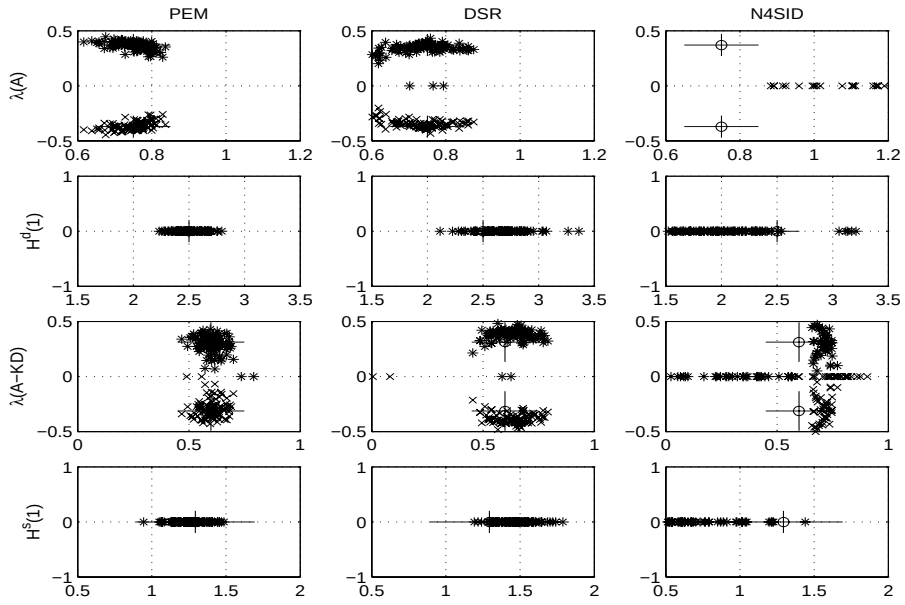

Figure 5.17: Estimates from closed loop Monte Carlo simulation of Example 1, Section 3.1, using low frequent PRBS, r_k^1 page 33 , as a reference when the noise variance is increased 50 times and the feedback is filtered through a Kalman filter found by DSR, Equation (5.8), in an initial step. PEM and N4SID are used with default parameters and $nk = 1$. DSR is used with $g = 0$, $L = 6$ and $J = 7$.

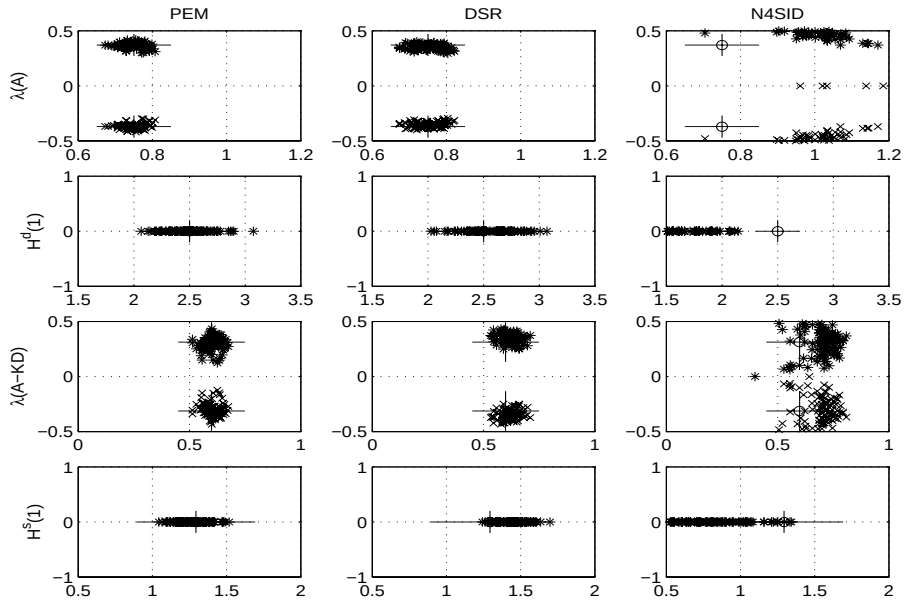

Figure 5.18: Estimates from closed loop Monte Carlo simulation of Example 1, Section 3.1, using a high frequent PRBS, r_k^6 page 72, as a reference when the noise variance is increased 50 times and the feedback is filtered through a Kalman filter found by DSR, Equation (5.9), in an initial step. PEM and N4SID are used with default parameters and $nk = 1$. DSR is used with $g = 0$, $L = 6$ and $J = 7$.

performed to improve the Kalman filter.

Using a reference signal, r_k^6 page 72, with a higher order of persistent excitation reduces the variance. PEM is unbiased and DSR has an insignificant bias on the estimates of the stochastic steady state gain of the Kalman filter. The rest of the estimates from DSR are unbiased. Compared to the case where the correct Kalman filter is used in the feedback, Figure 5.11, the use of the Kalman filter identified by DSR, Figure 5.18, leads to an increase in variance.

5.4.2 Multiple Input Multiple Output simulation example

The system introduced in Section 3.5 is used as a multiple input multiple output example. Time series of N=1000 discrete data points, $k = 0, 1, \ldots, N-1$, are generated. r_k^7 and r_k^8 are used as reference signals for output 1 and output 2, respectively. Figure 5.19 shows the reference signals, r_k^7 and r_k^8, plotted together with the corresponding outputs for two particular noise realizations, v_k and w_k, when the system is operating in a closed loop.

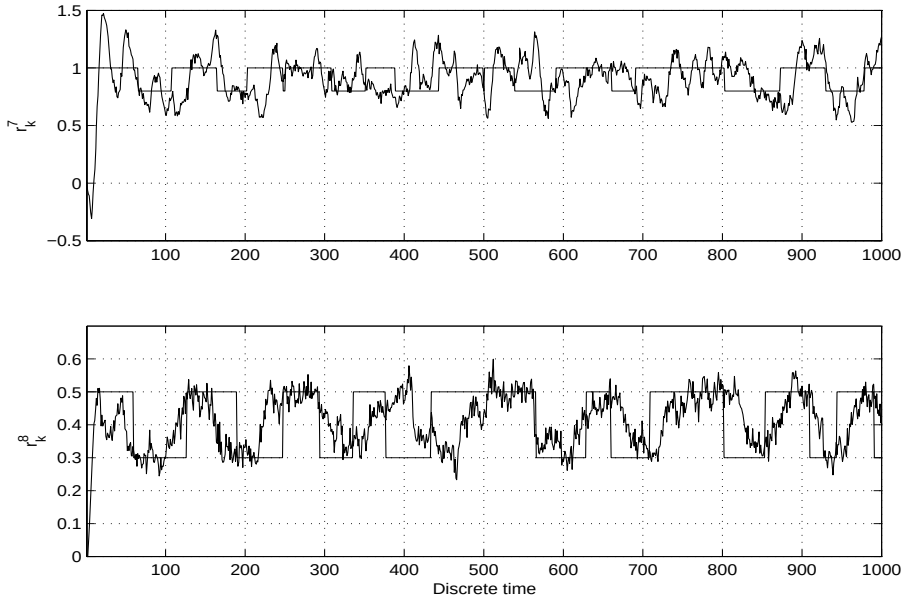

Figure 5.19: The reference signals, r_k^7 and r_k^8, plotted together with the corresponding outputs for two particular noise realizations, v_k and w_k, when the system, Example 5, Section 3.5, is operating in a closed loop.

In this example we assume that the system order $n = 3$ is known and that there is no direct feedthrough term from input to output. Figure 5.20 shows the estimates from a Monte Carlo simulation with 100 runs where r_k^7 and r_k^8 are used as the reference signals with different noise realizations in each run where direct

closed loop identification is performed. As in the single input single output case we observe that PEM gives unbiased estimates and both the SID algorithms give biased estimates. Regarding the estimation of the zeros two comments have to be made. The first is that when the eigenvalue estimates from the SID algorithms are so poor, as they are here, it is just a coincidence that the estimation of the zeros, compared to PEM, is so good. The second is that when PEM has one or more estimates which seem like "outliers", the zeros are hard to estimate.

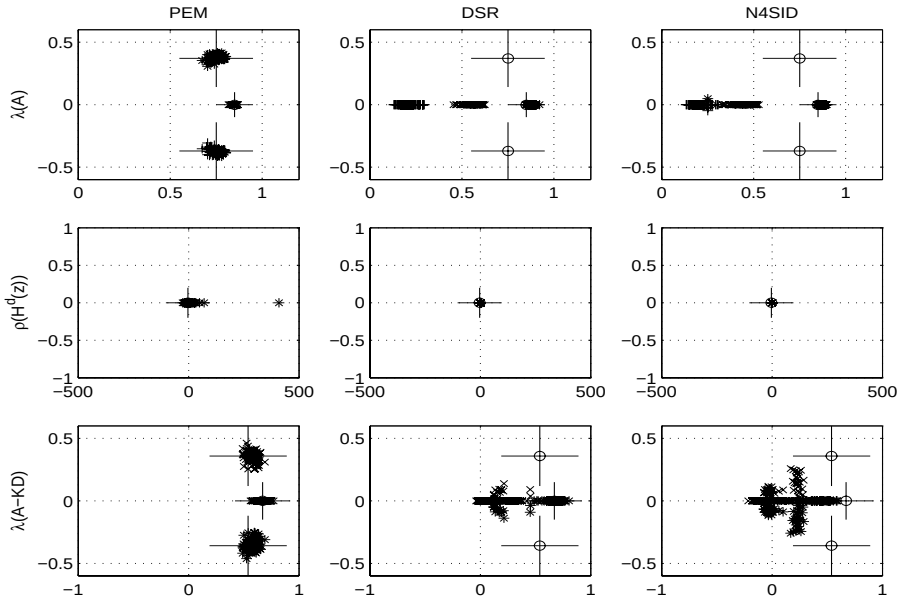

Figure 5.20: Estimates from closed loop Monte Carlo simulation of Example 5, Section 3.5, using r_k^7 and r_k^8, page 90, as references. PEM and N4SID are used with default parameters and $nk = 1$. DSR is used with $g = 0$, $L = 8$ and $J = 9$.

In order to evaluate the quality of the algorithm introduced in Section 5.4, Step 1 is performed by a single simulation using r_k^7 and r_k^8, page 90, as references to identify a (biased) model using DSR. The Kalman filter found by DSR with $g = 0$, $L = 8$ and $J = 9$ is given by

$$A = \begin{bmatrix} 0.7872 & 0.4514 & 0.2434 \\ 0.0435 & 0.6936 & -0.0203 \\ 0.0184 & 0.1157 & 0.1970 \end{bmatrix}, B = \begin{bmatrix} -4.1692 & -8.9423 \\ -2.2756 & 2.6672 \\ 1.7208 & -0.5849 \end{bmatrix},$$

$$D = \begin{bmatrix} -0.6301 & 0.5400 & 0.3128 \\ 0.0072 & -0.3117 & 0.8906 \end{bmatrix}, K = \begin{bmatrix} -0.7383 & -0.3019 \\ -0.0802 & -0.1532 \\ -0.0350 & 0.2059 \end{bmatrix}. \quad (5.10)$$

The references r_k^7 and r_k^8 are plotted in Figure 5.21 with the corresponding outputs, for two particular noise realizations with unchanged noise level, to illustrate

the effect on the noise level when using feedback filtered through the Kalman filter found by DSR, Equation (5.10). There is no significant reduction in the noise level. This is not a problem because the goal is to generate a feedback which is sufficiently uncorrelated with the noise on the output of the actual process.

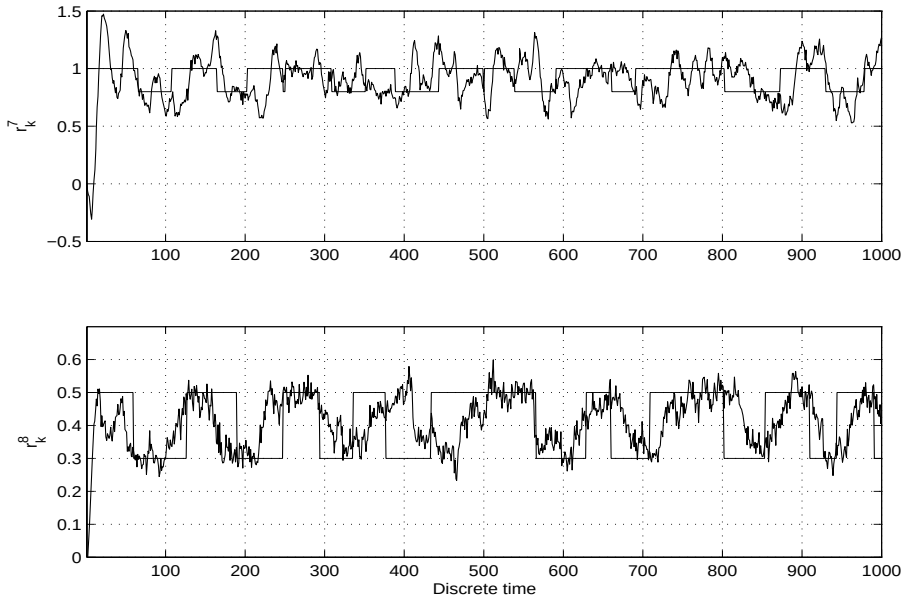

Figure 5.21: The reference signals, r_k^7 and r_k^8, with corresponding outputs from Example 5, Section 3.5, for two particular noise realizations, v_k and w_k, when the feedback is filtered through a Kalman filter found by DSR, Equation (5.10), in an initial step.

Now when the feedback is filtered through the Kalman filter found by DSR, Equation (5.10), all the methods give unbiased estimates, but the estimates from N4SID have considerably larger variance than the others. It is quite satisfactory that the estimates from PEM do not have any observable increase in variance, except the zeros, when the feedback is filtered through the Kalman filter found by DSR, Figure 5.21, compared to direct closed loop identification, Figure 5.20. It supports the observations in the single input single output example, Section 5.4.1, that the Kalman filter estimated in Step 1 in the algorithm does not have to be very accurate to have the desired effect.

As in the example in Section 5.4.1 we observe that the control function still is satisfactory when the feedback is filtered through the Kalman filter found by DSR in the initial step.

Figure 5.23 shows the references r_k^7 and r_k^8 with the corresponding outputs for

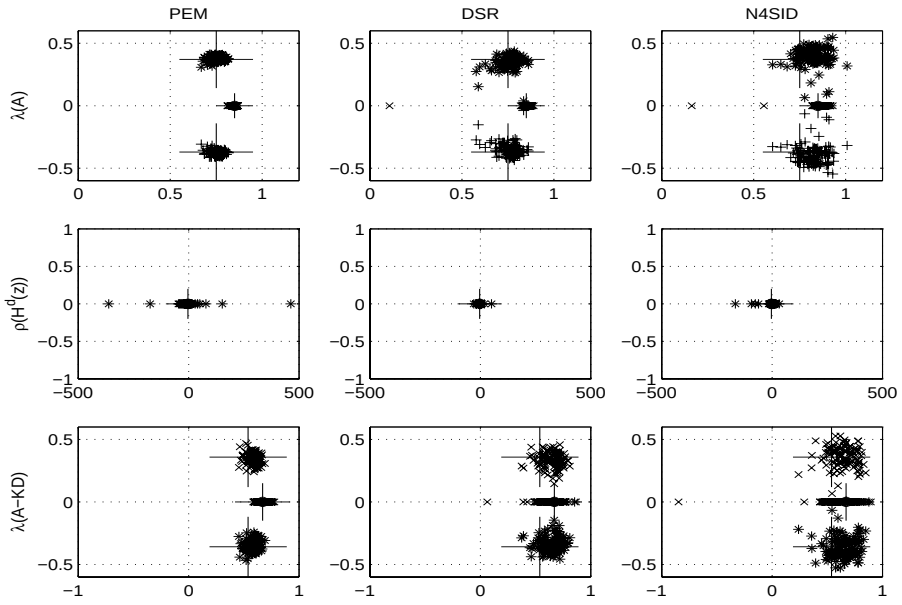

Figure 5.22: Estimates from closed loop Monte Carlo simulation of Example 5, Section 3.5, using r_k^7 and r_k^8, page 90, as references when the feedback is filtered through a Kalman filter found by DSR, Equation (5.10), in an initial step. PEM and N4SID are used with default parameters and $nk = 1$. DSR is used with $g = 0$, $L = 8$ and $J = 9$.

two particular noise realizations with the same noise level as in the previous simulations when the feedback is filtered through the correct Kalman filter.

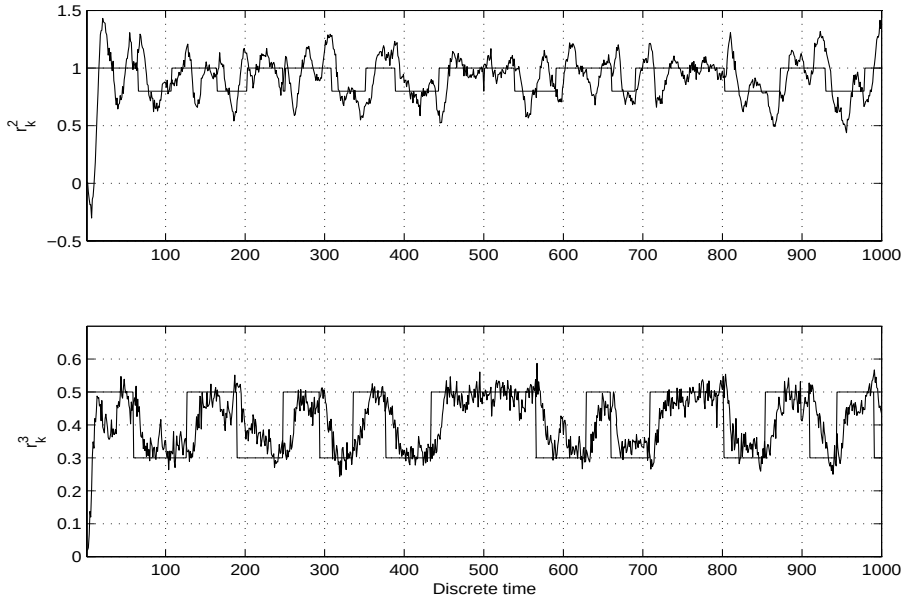

Figure 5.23: The reference signals, r_k^7 and r_k^8, with corresponding outputs from Example 5, Section 3.5, for two particular noise realizations, v_k and w_k, when the feedback is filtered through the correct Kalman filter.

Figure 5.24 shows the estimates when the feedback is filtered through the correct Kalman filter. As in Section 5.4.1 it has to be noted that there is no significant improvement of the performance compared to when the feedback is filtered through the Kalman filter found by DSR in an initial step, Figure 5.22.

5.4.3 Comments on the algorithm

A new three-step closed loop subspace identification algorithm based on the DSR algorithm and the Kalman filter properties is presented. In an initial step DSR is used for identification of the process model, including the Kalman filter gain. This model may have a bias when the system is operating in closed loop and there is noise present. The next step is to implement the Kalman filter in the feedback in such a way that the controller uses the filtered output from the filter, not the actual process measurement. The idea is that the Kalman filter found by DSR will give an output which is sufficiently uncorrelated with the noise on the output of the actual process, and in this way reduce or eliminate the bias problem. The final step is to use DSR to identify the process model when the feedback is filtered through the Kalman filter. This model will be unbiased if the Kalman filter is correct.

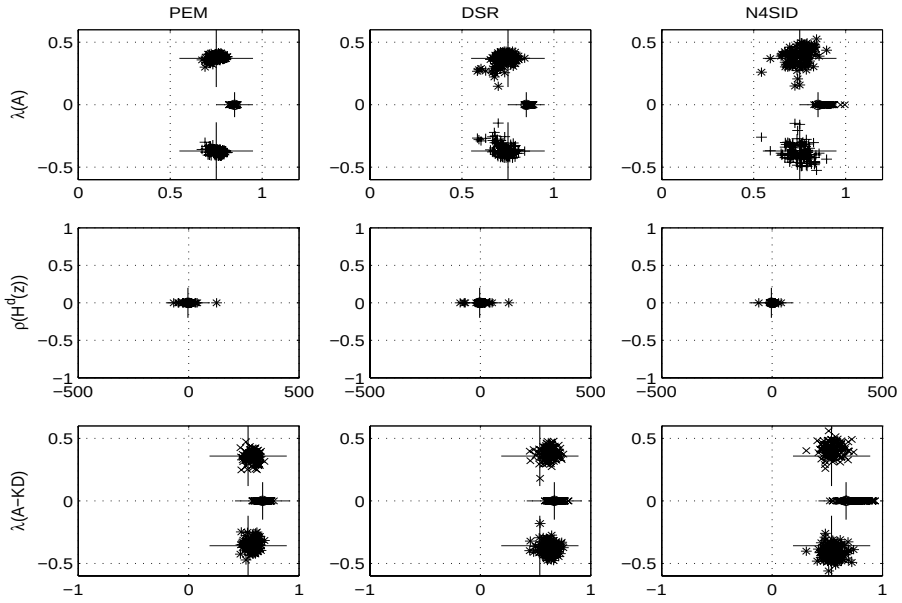

Figure 5.24: Estimates from closed loop Monte Carlo simulation of Example 5, Section 3.5, using r_k^7 and r_k^8 as references when the feedback is filtered through the correct Kalman filter. PEM and N4SID are used with default parameters and $nk = 1$. DSR is used with $g = 0$, $L = 8$ and $J = 9$.

Our simulation studies have shown that even when a Kalman filter with a bias is used the estimated model in the final step is unbiased.

The initial idea was that any subspace identification algorithm which estimates the full state space model, inclusive of the Kalman filter gain, should be applicable for this algorithm. The simulation study performed showed that it is not advisable to use N4SID in the initial step in the algorithm due to poor results. N4SID can be used in the final step in the algorithm, but it is not advisable because the variance is much larger than when DSR is used.

Chapter 6

Closed Loop Subspace Identification

So far when considering SID algorithms for closed loop system identification it has been focused on the existing algorithm DSR, and compared to N4SID and PEM. Two main topics have been treated, the use of DSR on finite closed loop data sets and how to modify the control loop to avoid the possible bias problem using DSR for direct closed loop SID.

The projections used in DSR to estimate the extended observability matrix, and the eigenvalues, have been compared to the projections used in other SID algorithms. The results shown in Section 4.4 give no indications that any of the other projections should be used instead of the ones used in DSR, both for use with open and closed loop data sets. A special case has been presented in Section 4.5 where a special choice of parameters eliminates the bias. When DSR is used for closed loop SID the optimal parameter choice is different from the open loop case, Section 4.6. When using a dithering signal in the reference, or on the input, of a system operating in closed loop it is favourable to use a signal with high order of persistent excitation if there is significant noise present. It is especially the estimation of the zeros which is hard using the classic SID algorithms for direct closed loop identification. When using a dithering signal on the input, instead of in the reference, the bias is reduced.

The modification of the control loop that has been considered is filtering the feedback. Different filter types have been tried. The best effect was obtained by using a Kalman filter in the feedback. A three-step algorithm based on the DSR algorithm and the Kalman filter properties was introduced.

The goal for closed loop SID algorithms must be to introduce algorithms for direct identification which is as easy to use on finite closed loop data sets as the original SID algorithms are used on finite open loop data sets. In this Chapter

two methods will be presented. Jansson (2003) has introduced a SID algorithm which first estimates a higher order ARX model to get estimates of the impulse response coefficients. Then projections are performed and the matrices in the state space model are estimated. In Section 6.1 a system identification method based on model reduction of a higher order ARX model is presented. The method is presented to show that when a higher order ARX model is identified there is no need for additional projections to estimate the system matrices. It is sufficient to perform a model reduction step. This also means that the method presented in Section 6.1 is not a SID algorithm and therefore the method by Jansson (2003) can hardly be called a SID algorithm either. The main part of this section will treat a new closed loop SID algorithm named DSR_e, Di Ruscio (2004), and simulation studies of it. The DSR_e algorithm is a modification of the existing DSR algorithm.

6.1 A simple algorithm

A method to identify a state space model from given inputs and outputs can be described by the following three steps. The first step is to identify a higher order ARX model by for instance using the ARX function in Matlab. The second step is to convert the ARX model to a state space model. The third step is a model reduction technique based on the properties of (block) Hankel matrices constructed from the impulse responses. The model reduction technique is described in Di Ruscio (2003b). Only the equations needed to do the model reduction are presented here. The impulse responses are given by

$$H_i = DA^{i-1}[B \;\; K]. \tag{6.1}$$

The submatrices needed to form the (block) Hankel matrix constructed from the impulse responses are given by

$$H_{1|L} = \begin{bmatrix} H_1 & H_2 & H_3 & \dots & H_J \\ H_2 & H_3 & H_4 & \dots & H_{J+1} \\ H_3 & H_4 & H_5 & \dots & H_{J+2} \\ \vdots & \vdots & \ddots & \ddots & \vdots \\ H_L & H_{L+1} & \dots & \dots & H_{L+J-1} \end{bmatrix} \mathbb{R}^{m\cdot L \times r\cdot J}. \tag{6.2}$$

Where L is the future horizon and J is the past horizon.

$$H_{2|L} = \begin{bmatrix} H_2 & H_3 & H_4 & \dots & H_{J+1} \\ H_3 & H_4 & H_5 & \dots & H_{J+2} \\ H_4 & H_5 & H_6 & \dots & H_{J+3} \\ \vdots & \vdots & \ddots & \ddots & \vdots \\ H_{L+1} & H_{L+2} & \dots & \dots & H_{L+J} \end{bmatrix} \mathbb{R}^{m\cdot L \times r\cdot J}. \tag{6.3}$$

The singular value decomposition (SVD) of the finite (block) Hankel matrix $H_{1|L}$ have to be computed to obtain the system matrices

$$H_{1|L} = USV^T = [U_1 \quad U_2] \begin{bmatrix} S_1 & 0 \\ 0 & S_2 \end{bmatrix} \begin{bmatrix} V_1 \\ V_2 \end{bmatrix}. \tag{6.4}$$

The order of the state space model is equal to the number of the non-zero singular values in the singular value decomposition of $H_{1|L}$, Equation (6.4).

Utilizing that

$$H_{1|L} = O_L C_J \tag{6.5}$$

where O_L is given by Equation (4.7) and

$$C_J \stackrel{def}{=} [[B \;\; K] \quad A[B \;\; K] \quad \ldots \quad A^{J-1}[B \;\; K]] \in \mathbb{R}^{n \times L(r+m)} \tag{6.6}$$

and choosing output normal equation

$$\begin{aligned} O_L &= U_1 && (6.7) \\ C_J &= S_1 V_1^T && (6.8) \end{aligned}$$

the following system matrices can be found by

$$\begin{aligned} D &= O_L(1:m,:) && (6.9) \\ B &= C_J(:,1:r) && (6.10) \\ K &= C_J(:,r+1:r+m). && (6.11) \end{aligned}$$

Utilizing that

$$H_{2|L} = O_L A C_J \tag{6.12}$$

gives

$$A = (O_L^T O_L)^\dagger O^T H_{2|L} C_J^T (C_J C_J^T)^\dagger \tag{6.13}$$

where $\dagger$ denotes the Moore-Penrose pseudo inverse.

This method does not have its own name, to my knowledge. For simplicity it will be presented as h_arx_mr (Higher order ARX model with Model Reduction).

6.2 The DSR_e algorithm

A SID algorithm for open and closed loop systems is presented in Di Ruscio (2004) and implemented in the DSR_e Matlab function in the DSR Toolbox for Matlab. In this section the algorithm will be analysed further. The algorithm is based on the fact that the noise innovation process can be identified directly from the data in a first step, Di Ruscio (1995), Di Ruscio (2001). The estimation of the noise innovation process is consistent, both in the case of open and closed loop data. This section, section 6.1 and section 6.3 is an extended version of Nilsen and Di Ruscio (2004c).

Before introducing the method some basic definitions used in Di Ruscio (2003a) and Di Ruscio (2004) have to be introduced. When describing the fundamentals of the DSR algorithm, Section 4.1, a necessary assumption is that the states are a function of the past, Section 4.1.3. Let us consider the discrete time Kalman filter in innovation form, Equations (2.15) and (2.16). Given a number of N data points

$$\left.\begin{matrix} u_k \\ y_k \end{matrix}\right\} \ \forall \ k = 0, 1, 2, \ldots, N-1 \tag{6.14}$$

a matrix equation of the predicted state at time $k+J$ of a Kalman filter with the initial predicted state at time k is

$$\bar{X}_{k+J} = \tilde{C}_J^s Y_{k|J} + \tilde{C}_J^d U_{k|J} + (A-KD)^J \bar{X}_k. \tag{6.15}$$

$\tilde{C}_J^s = C_J(A-KD, K)$ 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 $(A-KD, B-KE)$. The definition of the extended controllability matrix is given by Equation (4.8). $\bar{X}_k$ is the initial predicted state (estimate) at the initial discrete time k. The past horizon, J, is introduced in Section 4.1.1. For simplicity of notation the following definition is introduced $X = \bar{X}$ which gives

$$X_J = \tilde{C}_J^s Y_{0|J} + \tilde{C}_J^d U_{0|J} + (A-KD)^J X_0, \tag{6.16}$$

when the initial discrete time $k = 0$. By using the the extended output matrix Equation (4.12) and setting the initial discrete time to J gives

$$Y_{J|L} = O_L X_J + H_L^d U_{L|L+g-1} + H_L^s E_{J|L}. \tag{6.17}$$

Combining Equations (6.16) and (6.17) gives the relationship between the past and the future

$$Y_{J|L} = [H_L^d \ O_L\tilde{C}_J^d \ O_L\tilde{C}_J^s] \begin{bmatrix} U_{J|L+g-1} \\ U_{0|J} \\ Y_{0|J} \end{bmatrix} + O_L(A-KD)^J X_0 + H_L^s E_{J|L}. \quad (6.18)$$

This equation describes the same properties as Equation (4.19), the relationship between the past and the future, and is the main equation needed to understand the DSR_e algorithm. It is important to note which terms which are proportional with the extended observability matrix O_L. Both Equations (4.19) and (6.18) show that the effect of 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. The choice of projections used in DSR have already been presented in Section 4.1.4.

In the following it will be assumed that there is no direct feedthrough term in the closed loop systems considered, which gives $E = 0$, which also is the case in closed loop control systems due to process dynamics. Using Equation (6.18) with $g = 0$, $L = 1$ and letting $J \to \infty$ gives

$$Y_{J|1} = D[\tilde{C}_J^d \ \tilde{C}_J^s] \begin{bmatrix} U_{0|J} \\ Y_{0|J} \end{bmatrix} + FE_{J|1}. \quad (6.19)$$

Hence, the innovation is given by

$$Z_{J|1}^s \stackrel{def}{=} FE_{J|1} = Y_{J|1} - Y_{J|1}/\begin{bmatrix} U_{0|J} \\ Y_{0|J} \end{bmatrix}, \quad (6.20)$$

and the innovations sequence in Equations (2.15)-(2.16) is given by

$$[\varepsilon_J \ \varepsilon_{J+1} \ \ldots \ \varepsilon_{N-1}] = Z_{J|1}^s \in \mathbb{R}^{m\times(N-1-J)}. \quad (6.21)$$

This approach is valid for both open and closed loop systems since the past data, $U_{0|J}$ and $Y_{0|J}$, are uncorrelated with the future noise $E_{J|1}$.

When the innovation is known the remaining problem is to solve a deterministic SID problem to obtain the system order, n, and the system matrices A, B, K and D. The problem is solved by using the DSR algorithm directly to identify the following model

$$x_{k+1} = Ax_k + [B \ K] \begin{bmatrix} u_k \\ \varepsilon_k \end{bmatrix}, \quad (6.22)$$

$$y_k - \varepsilon_k = Dx_k. \quad (6.23)$$

Note that, when the data matrices are formed the right-hand side of Equation (6.23) equals DX_J, which also can be found on the right-hand side of Equation (6.20). Defining

$$Z^d_{J|1} \stackrel{def}{=} Y_{J|1} / \begin{bmatrix} U_{0|J} \\ Y_{0|J} \end{bmatrix} \tag{6.24}$$

gives

$$DX_J = [Dx_J \;\; Dx_{J+1} \ldots Dx_{N-1}] = Z^d_{J|1}. \tag{6.25}$$

This gives

$$[y_J - \varepsilon_J \quad y_{J+1} - \varepsilon_{J+1} \quad \ldots \quad y_{N-1} - \varepsilon_{N-1}] = Z^d_{J|1}. \tag{6.26}$$

The theory presented in Section 4.1 can be used directly on deterministic systems by zeroing out the matrices $U_{0|J}$ and $Y_{0|J}$ from the projections. This gives that the extended observability matrix, O_{L+1}, and thus matrices A and D, can be found from the projection equation

$$Z_{J|L+1} = Y_{J|L+1} \begin{bmatrix} U_{J|L} \\ E_{J|L+1} \end{bmatrix}^{\perp} \approx O_{L+1} X^a_J. \tag{6.27}$$

Using the theory in Section 4.1, matrices B and K can be found from the projection equation

$$Y_{J+1|L} = \tilde{A}_L Y_{J|L} + [\tilde{B}_L \;\; \tilde{C}_L] \begin{bmatrix} U_{J|L} \\ E_{J|L+1} \end{bmatrix}. \tag{6.28}$$

6.2.1 The basic steps in the DSR_e algorithm

1. **Estimate the future innovation matrix**

   The future innovation matrix is estimated from Equation (6.20)

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

   where a large past horizon, J, has to be chosen.

2. **Form the block Hankel matrix** $E_{J|L+1}$

   The block Hankel matrix $E_{J|L+1}$ is formed directly from the innovation sequence $\varepsilon_k \;\forall\; k = J, J+1, \ldots, N-1$ in the Kalman filter, Equation (6.21).

3. **Form the block Hankel matrices $U_{J|L}$ and $Y_{J|L+1}$**

   The block Hankel matrices $U_{J|L}$ and $Y_{J|L+1}$ are formed directly from the known input and output data sequence u_k and y_k $\forall$ $k = J, J+1, \ldots, N-1$.

4. **Estimate the extended observability matrix**
   The extended observability matrix O_{L+1} is estimated from Equation (6.27)

$$Z_{J|L+1} = Y_{J|L+1} \begin{bmatrix} U_{J|L} \\ E_{J|L+1} \end{bmatrix}^{\perp} \approx O_{L+1} X_J^a.$$

   The matrices A and D are found from the extended observability matrix as described in Section 4.1.

5. **Estimate the matrices B and K**
   Matrices B and K are estimated from Equation (6.28)

$$Y_{J+1|L} = \tilde{A}_L Y_{J|L} + [\tilde{B}_L \ \tilde{C}_L] \begin{bmatrix} U_{J|L} \\ E_{J|L+1} \end{bmatrix}.$$

6. **Compute the initial state X_0**

6.3 Simulation examples

This section contains simulation examples to test the performance of the DSR_e algorithm, introduced in Section 6.2, and the algorithm named h_arx_mr, introduced in Section 6.1. The algorithms will be compared to PEM implemented in Matlab. The reason for this is that the results from a closed loop SID algorithm have to be comparable to the results from PEM in order to be considered as an alternative to PEM. The results using the classic SID algorithms DSR, N4SID and MOESP, Verhaegen (1994), are presented to show the improvement in DSR_e compared to the classic SID algorithms.

The examples used have been presented in Section 3. The estimates presented are the eigenvalue of the system matrix A, $\lambda(A)$, the deterministic transition zeros of the system, $\rho(A, B, D, E)$, and the eigenvalues of the Kalman filter, $\lambda(A - KD)$. Focus is placed on reference signals with a low level of persistent excitation. The reason for this is that poor projections are revealed by the need of rich excitation signals and sensitivity to noise.

The system order and the fact that the matrix E is the zero matrix is assumed known. PEM and N4SID are used with default parameters and $nk = 1$. The parameter chosen for the rest of the methods are the values which minimize the Squared Eigenvalue Error criterion, V^1, unless other values specified.

6.3.1 Example 1

Using a low frequent PRBS as reference signal

The example used in this section was introduced in Section 3.1. The noise level used is the noise variance presented in Section 3.1. The reference used is the low frequent PRBS introduced in Section 4.4, Figure 4.3 page 33. PEM and N4SID are used with default parameters and $n_k = 1$. DSR_e is used with $g = 0$, $L = 5$ and $J = 7$. In h_arx_mr the model order of the higher order ARX model used is 6 together with $L = 6$ and $J = 6$. MOESP is used with $s = 10$. DSR is used with $g = 0$, $L = 4$ and $J = 4$. The estimated eigenvalues are shown in Figure 6.1. The estimated zeros are presented in Figure 6.2 and the estimated eigenvalues of the Kalman filter are shown in Figure 6.3.

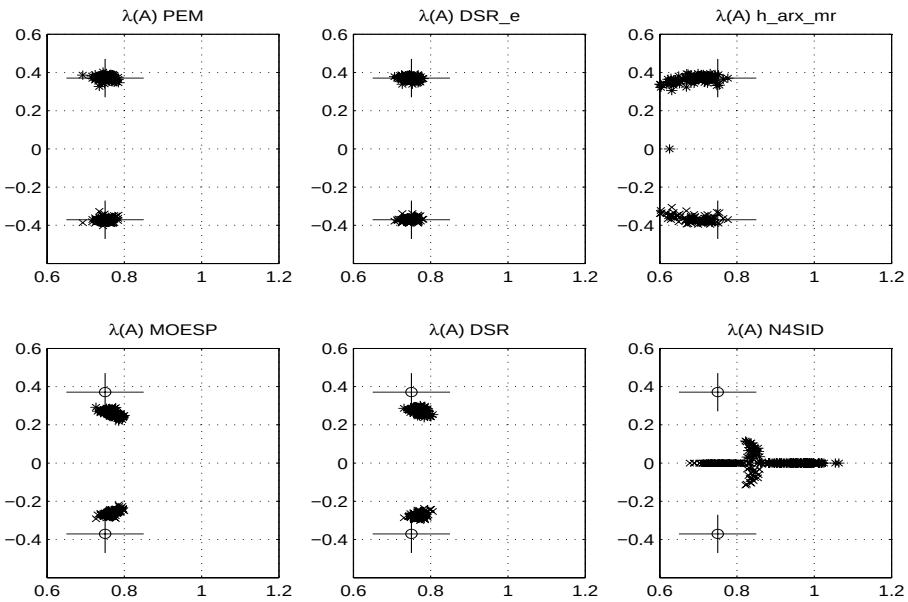

Figure 6.1: Eigenvalue estimates using low frequent PRBS, r_k^1 page 33, as the reference signal in the closed loop system, Example 1, Section 3.1. PEM and N4SID are used with default parameters and $nk = 1$. DSR_e is used with $g = 0$, $L = 5$ and $J = 7$. In h_arx_mr the model order of the higher order ARX model used is 6 together with $L = 6$ and $J = 6$. MOESP is used with $s = 10$. DSR is used with $g = 0$, $L = 4$ and $J = 4$.

The only methods that give unbiased eigenvalue estimates are PEM and DSR_e. The methods h_arx_mr, MOESP and DSR all have a bias, but h_arx_mr has larger variance than the others. The reason for this is that the methods need excitation with a higher order of persistent excitation to give unbiased results with reasonable variance. N4SID gives the poorest eigenvalue estimates.

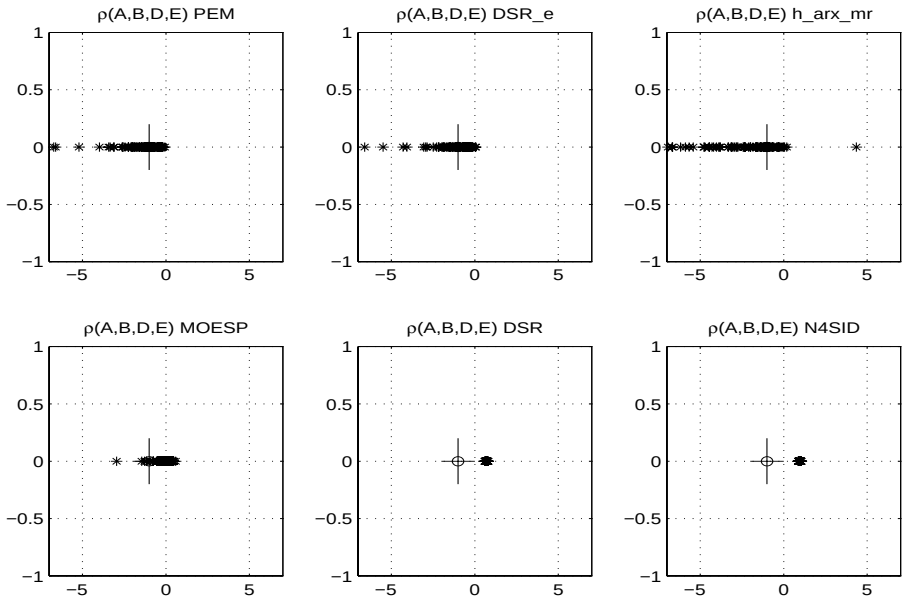

Figure 6.2: Estimated zeros using low frequent PRBS, r_k^1 page 33, as the reference signal in the closed loop system, Example 1, Section 3.1. PEM and N4SID are used with default parameters and $nk = 1$. DSR_e is used with $g = 0$, $L = 5$ and $J = 7$. In h_arx_mr the model order of the higher order ARX model used is 6 together with $L = 6$ and $J = 6$. MOESP is used with $s = 10$. DSR is used with $g = 0$, $L = 4$ and $J = 4$.

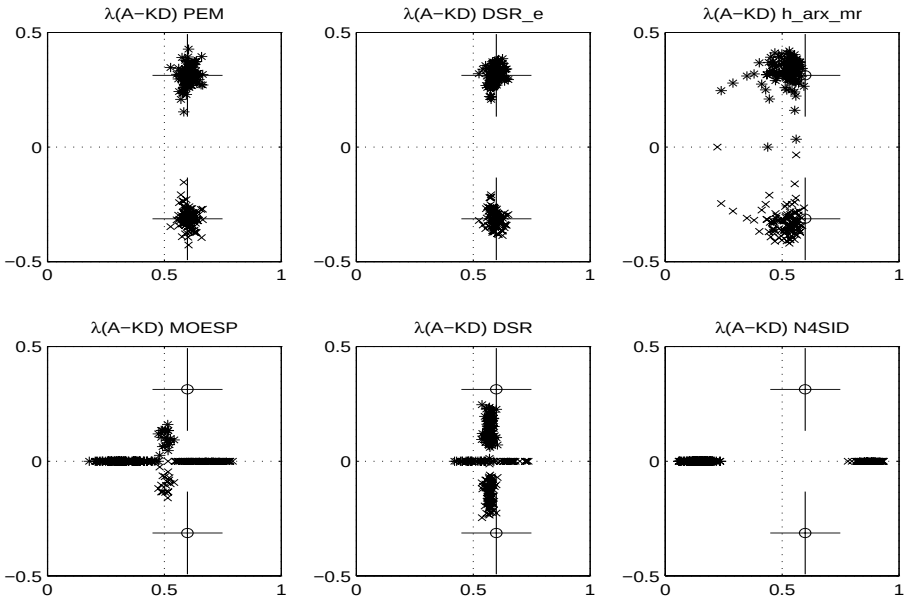

Figure 6.3: Estimated eigenvalues of the Kalman filter using low frequent PRBS, r_k^1 page 33, as the reference signal in the closed loop system, Example 1, Section 3.1. PEM and N4SID are used with default parameters and $nk = 1$. DSR_e is used with $g = 0$, $L = 5$ and $J = 7$. In h_arx_mr the model order of the higher order ARX model used is 6 together with $L = 6$ and $J = 6$. MOESP is used with $s = 10$. DSR is used with $g = 0$, $L = 4$ and $J = 4$.

Regarding the estimates of the zeros all methods have a bias. But the only method which is comparable to PEM, which is to be considered as the benchmark, is DSR_e. The only method with large variance is h_arx_mr. It has to be commented on that when N4SID, or any other SID algorithm, gives good a estimation of the zeros when the estimates of the eigenvalues are poor it is just pure luck. The reason for this argument is that in SID algorithms projections to estimate the extended observability matrix, and thereby the eigenvalues, are performed first. These results are used later when the rest of the state space model is identified. Therefore it is not possible to estimate zeros in a proper way if the estimates of the eigenvalues are poor.

The only methods that give proper estimates of the eigenvalues of the Kalman filter are PEM and DSR_e. Also here DSR_e is comparable to PEM. The classic SID algorithms suffer from large bias and h_arx_mr suffers from large variance and also has a small bias.

Using a low frequently PRBS as reference signal when the noise variance is increased

The low frequently PRBS is still used as reference signal but the variance of the noise is increased 50 times. PEM and N4SID are used with default parameters and $n_k = 1$. DSR_e is used with $g = 0$, $L = 5$ and $J = 7$. In h_arx_mr the model order of the higher order ARX model used is 3 together with $L = 3$ and $J = 3$. MOESP is used with $s = 15$. DSR is used with $g = 0$, $L = 6$ and $J = 7$. The estimated eigenvalues are shown in Figure 6.4. The estimated zeros are presented in Figure 6.5 and the estimated eigenvalues of the Kalman filter are shown in Figure 6.6.

PEM and DSR_e are the only methods which give reasonable estimates of the eigenvalues. The estimates from both methods are considered unbiased. The increased variance is the result of the increased noise variance. The rest of the methods give such poor results that they will not be commented on further.

The transition zero of the system has the value 1. The mean value of the estimated zeros found by PEM is approximately 6.5. The corresponding value for DSR_e is 1.2. The standard deviation of the estimates found by PEM is approximately 36 and the corresponding value for DSR_e is 18. This means that in this simulation DSR_e provides better estimates of the zeros than PEM.

The estimates of the eigenvalues of the Kalman filter from PEM are unbiased, but the estimates from DSR_e are not. The reason for this can be the parameter choice in DSR_e. The parameters chosen are the values which minimize the Squared Eigenvalue Error criterion, V^1. So far no other criteria have been con-

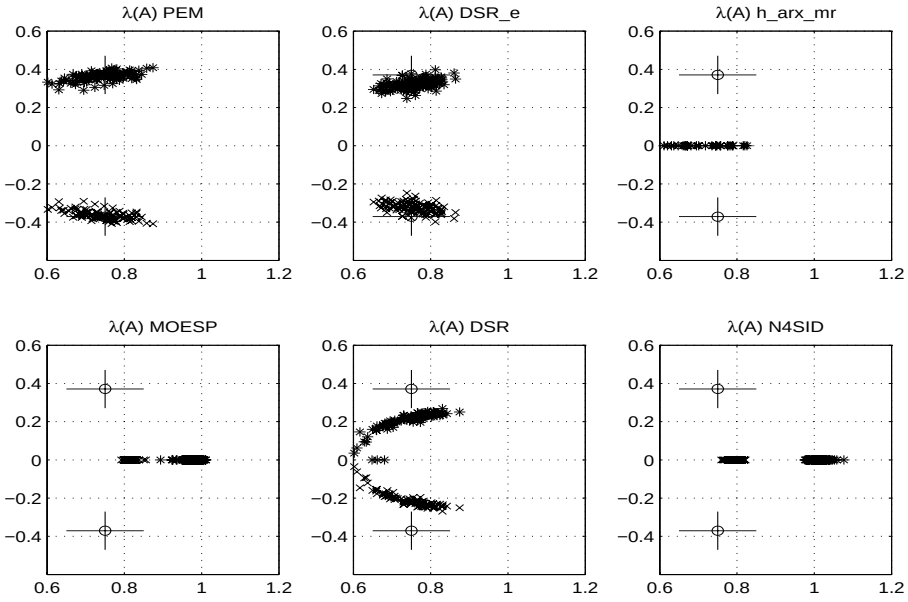

Figure 6.4: Eigenvalue estimates using low frequent PRBS, r_k^1 page 33, as the reference signal in the closed loop system, Example 1, Section 3.1, when the variance of the noise is increased 50 times. PEM and N4SID are used with default parameters and $n_k = 1$. DSR_e is used with $g = 0$, $L = 5$ and $J = 7$. In h_arx_mr the model order of the higher order ARX model used is 3 together with $L = 3$ and $J = 3$. MOESP is used with $s = 15$. DSR is used with $g = 0$, $L = 6$ and $J = 7$.

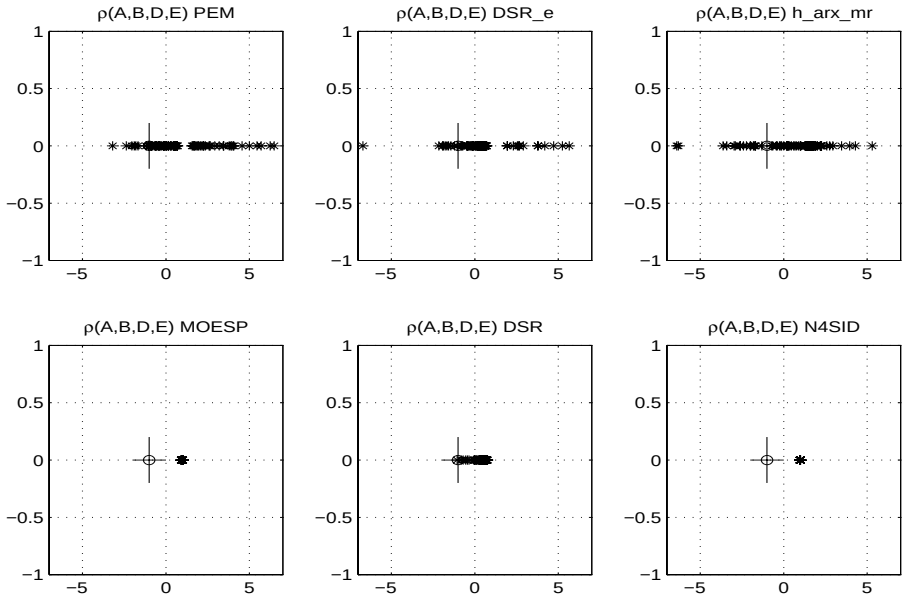

Figure 6.5: Estimated zeros using low frequent PRBS, r_k^1 page 33, as the reference signal in the closed loop system, Example 1, Section 3.1, when the variance of the noise is increased 50 times. PEM and N4SID are used with default parameters and $nk = 1$. PEM and N4SID are used with default parameters and $n_k = 1$. DSR_e is used with $g = 0$, $L = 5$ and $J = 7$. In h_arx_mr the model order of the higher order ARX model used is 3 together with $L = 3$ and $J = 3$. MOESP is used with $s = 15$. DSR is used with $g = 0$, $L = 6$ and $J = 7$.

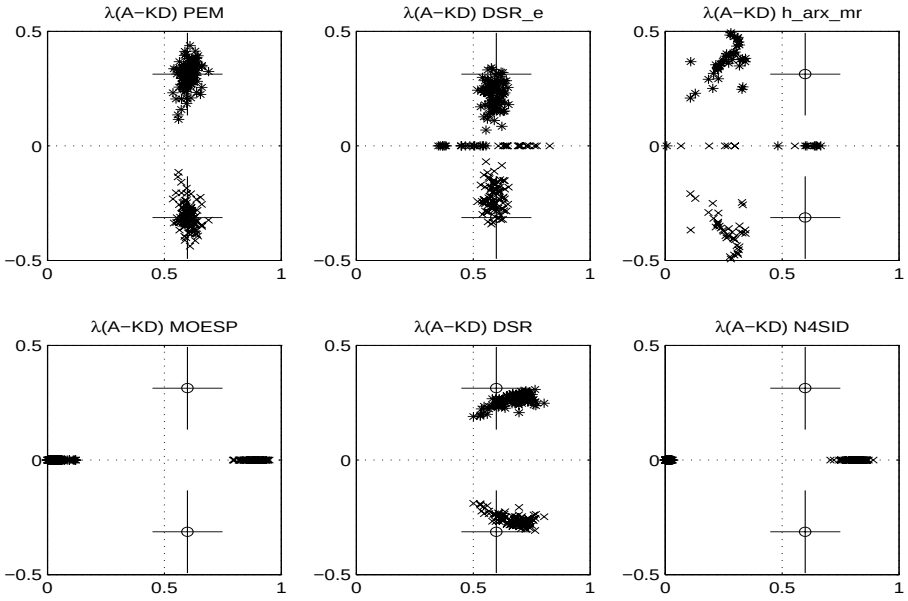

Figure 6.6: Estimated eigenvalues of the Kalman filter using low frequent PRBS, r_k^1 page 33, as the reference signal in the closed loop system, Example 1, Section 3.1, when the variance of the noise is increased 50 times. PEM and N4SID are used with default parameters and $n_k = 1$. DSR_e is used with $g = 0$, $L = 5$ and $J = 7$. In h_arx_mr the model order of the higher order ARX model used is 3 together with $L = 3$ and $J = 3$. MOESP is used with $s = 15$. DSR is used with $g = 0$, $L = 6$ and $J = 7$.

sidered. Functions for identification of system order will be presented in Section 6.4. These functions together with DSR_e can be considered as an automatic version of DSR_e where no parameters have to be set. A procedure based on visual inspection of the singular value plot from DSR_e to identify the system order succeeded by a search for the parameter settings which provide the smallest squared prediction error is presented in Section 6.5.

Using a high frequently PRBS as reference signal when the noise variance is increased

Now the high frequently PRBS r_k^6, introduced in Section 5.1 Figure 5.3 page 72, is chosen as the reference signal. The variance of the noise is 50 times larger than the level described in Section 3.1. PEM and N4SID are used with default parameters and $n_k = 1$. DSR_e is used with $g = 0$, $L = 6$ and $J = 7$. In h_arx_mr the model order of the higher order ARX model used is 3 together with $L = 3$ and $J = 3$. MOESP is used with $s = 10$. DSR is used with $g = 0$, $L = 4$ and $J = 5$. The estimated eigenvalues are shown in Figure 6.4. The estimated zeros are presented in Figure 6.5 and the estimated eigenvalues of the Kalman filter are shown in Figure 6.6.

The increase in the order of persistent excitation has led to eigenvalue estimates from PEM and DSR_e with smaller variance. The estimates from PEM and DSR_e are also comparable here. DSR has got reduced variance and bias, but is still biased. A question is if the bias on the estimates from DSR is of any practical interest, but that kind of consideration will not be treated in this work. The rest of the methods give poor results and will not be commented on any further.

Regarding the estimates of the transition zeros of the systems both PEM and DSR_e have a small bias, but DSR_e still has smaller variance than PEM. The increase in the order of persistent excitation has also led to improved estimates here.

The estimates of the eigenvalues of the Kalman filter are comparable for the PEM and DSR_e methods. This is an improvement for DSR_e since the estimates now are unbiased.

Using two sinusoid signals as the reference signal when the noise variance is increased

Now a reference signal consisting of two sinusoid signals is used. The reference signal is given by

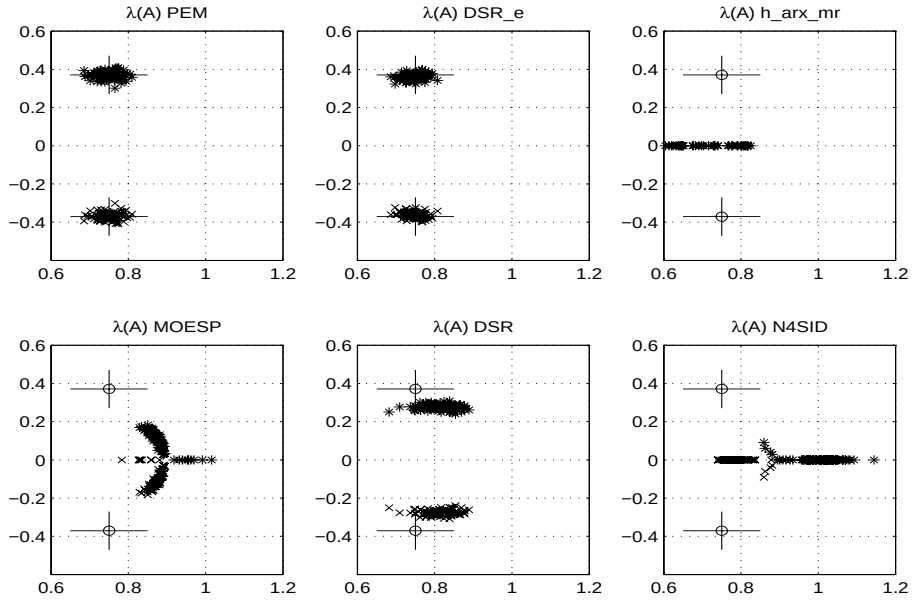

Figure 6.7: Eigenvalue estimates using high frequent PRBS, r_k^6 page 72, as the reference signal in the closed loop system, Example 1, Section 3.1, when the variance of the noise is increased by 50 times. PEM and N4SID are used with default parameters and $n_k = 1$. DSR_e is used with $g = 0$, $L = 6$ and $J = 7$. In h_arx_mr the model order of the higher order ARX model used is 3 together with $L = 3$ and $J = 3$. MOESP is used with $s = 10$. DSR is used with $g = 0$, $L = 4$ and $J = 5$.

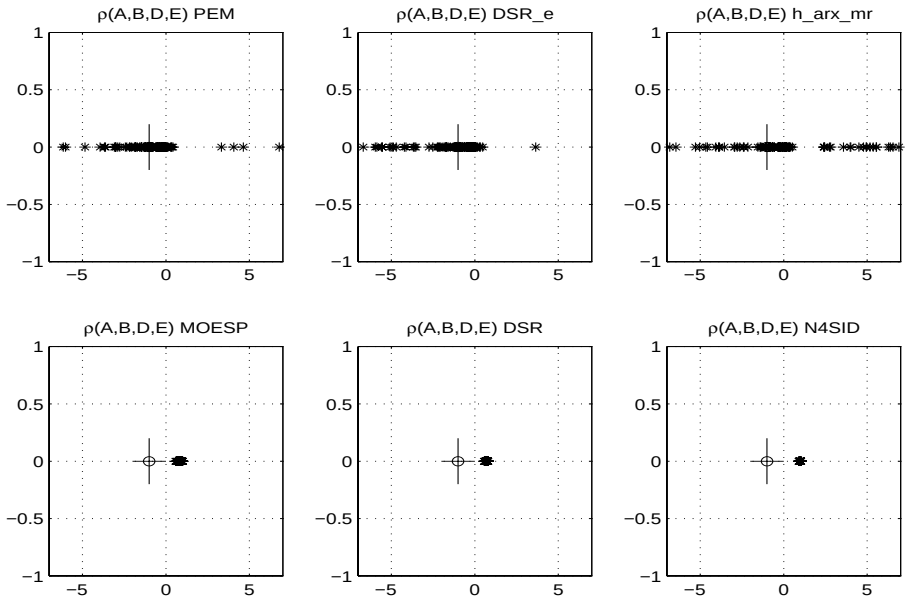

Figure 6.8: Estimated zeros using high frequent PRBS, r_k^6 page 72, as the reference signal in the closed loop system, Example 1, Section 3.1, when the variance of the noise is increased 50 times. PEM and N4SID are used with default parameters and $n_k = 1$. DSR_e is used with $g = 0$, $L = 6$ and $J = 7$. In h_arx_mr the model order of the higher order ARX model used is 3 together with $L = 3$ and $J = 3$. MOESP is used with $s = 10$. DSR is used with $g = 0$, $L = 4$ and $J = 5$.

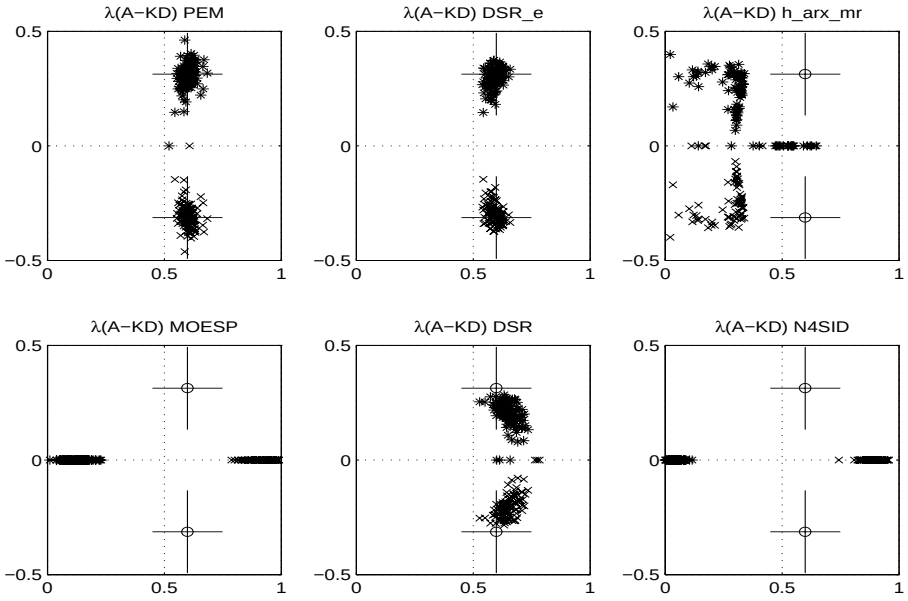

Figure 6.9: Estimated eigenvalues of the Kalman filter using high frequent PRBS, r_k^6 page 72, as the reference signal in the closed loop system, Example 1, Section 3.1, when the variance of the noise is increased 50 times. PEM and N4SID are used with default parameters and $n_k = 1$. DSR_e is used with $g = 0$, $L = 6$ and $J = 7$. In h_arx_mr the model order of the higher order ARX model used is 3 together with $L = 3$ and $J = 3$. MOESP is used with $s = 10$. DSR is used with $g = 0$, $L = 4$ and $J = 5$.

$$r_k^9 = \sin(k) + \sin\left(\frac{k}{2}\right). \tag{6.29}$$

The variance of the noise is 50 times larger than the level described in Section 3.1. PEM and N4SID are used with default parameters and $n_k = 1$. DSR_e is used with $g = 0$, $L = 3$ and $J = 7$. In h_arx_mr the model order of the higher order ARX model used is 14 together with $L = 14$ and $J = 14$. MOESP is used with $s = 10$. DSR is used with $g = 0$, $L = 5$ and $J = 6$. The estimated eigenvalues are shown in Figure 6.10. The estimated zeros are presented in Figure 6.11 and the estimated eigenvalues of the Kalman filter are shown in Figure 6.12.

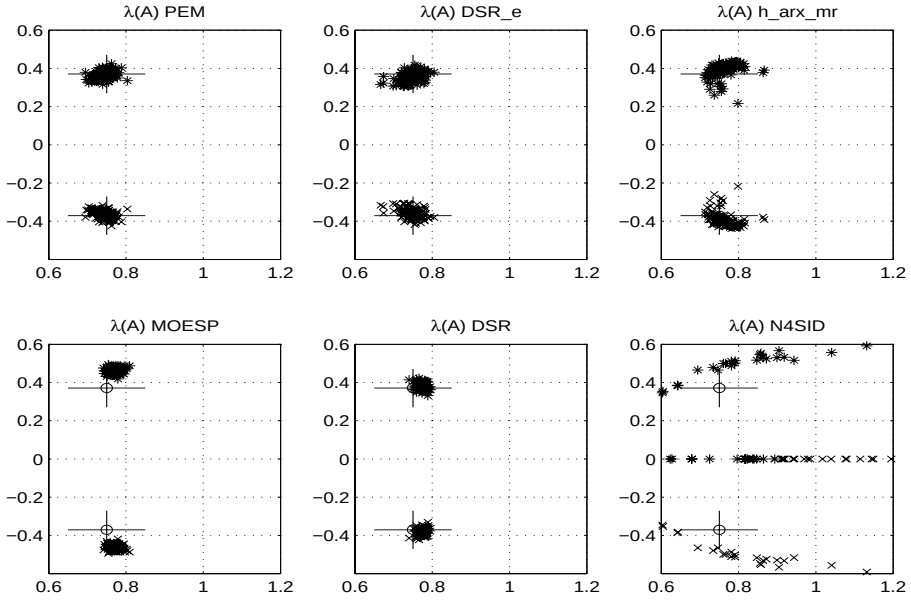

Figure 6.10: Eigenvalue estimates using a reference signal consisting of two sinusoid signals, r_k^9 page 115, in the closed loop system, Example 1, Section 3.1, when the variance of the noise is increased 50 times. PEM and N4SID are used with default parameters and $n_k = 1$. DSR_e is used with $g = 0$, $L = 3$ and $J = 7$. In h_arx_mr the model order of the higher order ARX model used is 14 together with $L = 14$ and $J = 14$. MOESP is used with $s = 10$. DSR is used with $g = 0$, $L = 5$ and $J = 6$.

Now the variance of the estimated eigenvalues is reduced for all the methods. PEM and DSR_e are still the only methods that give unbiased estimates, but the estimates from h_arx_mr, MOESP and DSR only have a small bias. MOESP and DSR are the methods with the smallest variance. The estimates from N4SID are still poor and will not be commented on any further.

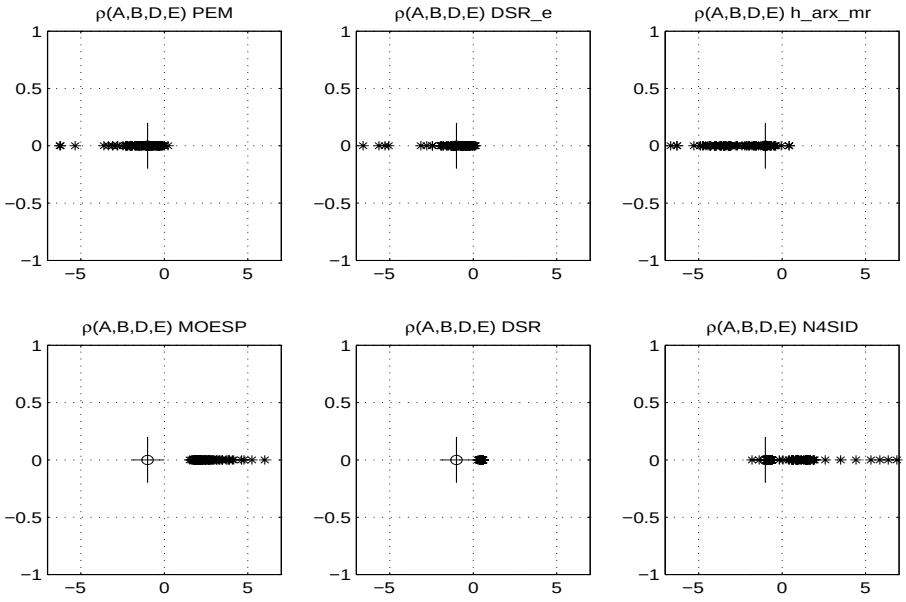

Figure 6.11: Estimated zeros using a reference signal consisting of two sinusoid signals, r_k^9 page 115, in the closed loop system, Example 1, Section 3.1, when the variance of the noise is increased 50 times. PEM and N4SID are used with default parameters and $n_k = 1$. DSR_e is used with $g = 0$, $L = 3$ and $J = 7$. In h_arx_mr the model order of the higher order ARX model used is 14 together with $L = 14$ and $J = 14$. MOESP is used with $s = 10$. DSR is used with $g = 0$, $L = 5$ and $J = 6$.

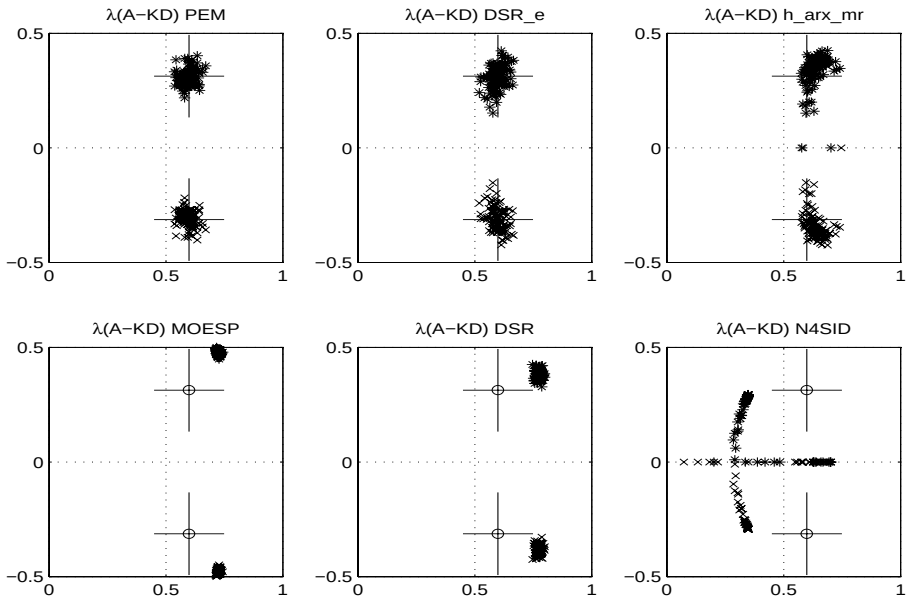

Figure 6.12: Estimated eigenvalues of the Kalman filter using a reference signal consisting of two sinusoid signals, r_k^9 page 115, in the closed loop system, Example 1, Section 3.1, when the variance of the noise is increased 50 times. PEM and N4SID are used with default parameters and $n_k = 1$. DSR_e is used with $g = 0$, $L = 3$ and $J = 7$. In h_arx_mr the model order of the higher order ARX model used is 14 together with $L = 14$ and $J = 14$. MOESP is used with $s = 10$. DSR is used with $g = 0$, $L = 5$ and $J = 6$.

All methods, except of MOESP and N4SID, have reduced variance on the estimated transition zeros. PEM, DSR_e and DSR only have a small bias.

Regarding the estimated eigenvalues of the Kalman filter, PEM has a reduction in the variance of the estimates. DSR_e is still comparable to PEM. The estimates from h_arx_mr are now unbiased, but still with a variance which is larger than for PEM and DSR_e. The estimates from MOESP and DSR are still biased, but the variance is small.

6.3.2 Example 2

Example 2 used in this section was introduced in Section 3.2. The noise level used is the noise variance presented in Section 3.2. The references used are the same as used in Section 4.6 with very little excitation, both in the level of persistent excitation and in the size of the amplitude.

The estimated eigenvalues of the system, the estimated zeros of the system and the estimated eigenvalues of the Kalman filter will also here be focused on. For this system the eigenvalues of the Kalman filter are hard to estimate. The Kalman filter gain has a very small numerical value, $K = [0.0043\ 0.0021]^T$. The deterministic steady state gain also has a small numerical value, $H^d(1) = 0.0160$, and the stochastic steady state gain is $H^s(1) = 1.0304$. Therefore this example is not a good example for considering its ability to estimate a full Kalman filter. This is also a system where DSR gives unbiased estimates of both eigenvalues and zeros, if used with the appropriate parameter setting and with an appropriate reference signal. But it is the goal to test both DSR_e and h_arx_mr on all examples introduced in this work to have a variety of examples for comparison.

Using a constant reference superposed with a sinusoid signal

The reference used is a constant value superposed with a sinusoid signal, with frequency $\omega = 0.725$ and with magnitude ± 0.1, introduced in Section 4.6.1, Equation (4.67). PEM and N4SID are used with default parameters and $n_k = 1$. DSR_e is used with $g = 0$, $L = 9$ and $J = 11$. In h_arx_mr the model order of the higher order ARX model used is 9 together with $L = 9$ and $J = 9$. MOESP is used with $s = 11$. DSR is used with $g = 0$, $L = 11$ and $J = 12$. The estimated eigenvalues are shown in Figure 6.13. The estimated zeros are presented in Figure 6.14 and the estimated eigenvalues of the Kalman filter are shown in Figure 6.15.

The methods that give eigenvalue estimates with considerable bias are h_arx_mr and N4SID, this is due to the need for input with a high order of persistent excitation. In this case DSR_e is not directly comparable to PEM.

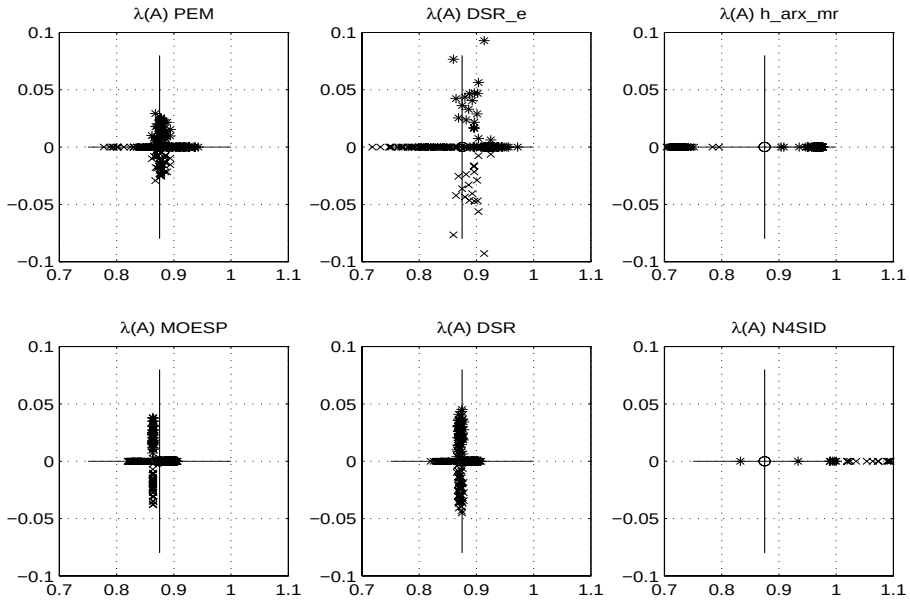

Figure 6.13: Eigenvalue estimates using a constant reference superposed with a sinusoid signal, $r_k^3 = 1 + 0.1\sin(\omega k)$ with $\omega = 0.725$, in the closed loop system, Example 2, Section 3.2. PEM and N4SID are used with default parameters and $n_k = 1$. DSR_e is used with $g = 0$, $L = 9$ and $J = 11$. In h_arx_mr the model order of the higher order ARX model used is 9 together with $L = 9$ and $J = 9$. MOESP is used with $s = 11$. DSR is used with $g = 0$, $L = 11$ and $J = 12$.

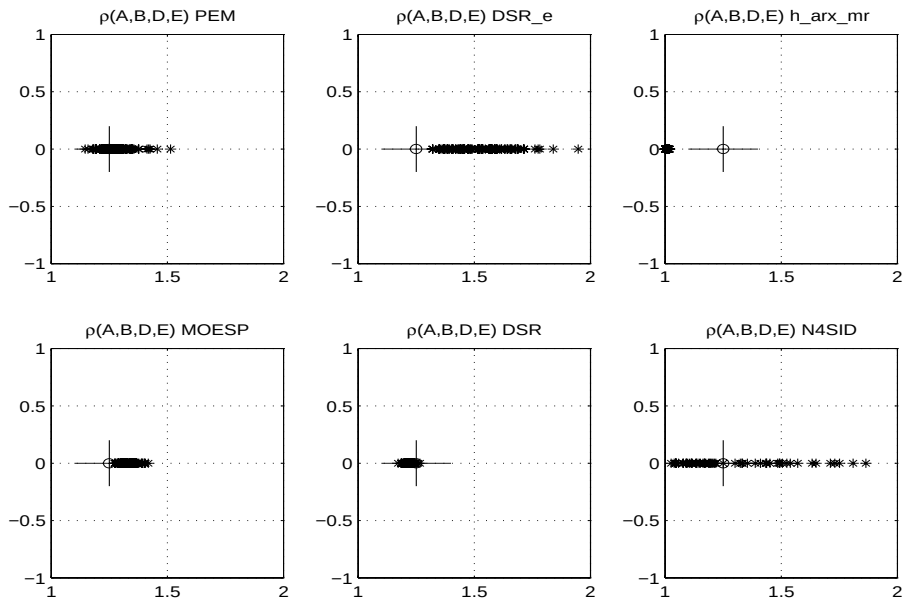

Figure 6.14: Estimated zeros using a constant reference superposed with a sinusoid signal, $r_k^3 = 1 + 0.1\sin(\omega k)$ with $\omega = 0.725$, in the closed loop system, Example 2, Section 3.2. PEM and N4SID are used with default parameters and $n_k = 1$. DSR_e is used with $g = 0$, $L = 9$ and $J = 11$. In h_arx_mr the model order of the higher order ARX model used is 9 together with $L = 9$ and $J = 9$. MOESP is used with $s = 11$. DSR is used with $g = 0$, $L = 11$ and $J = 12$.

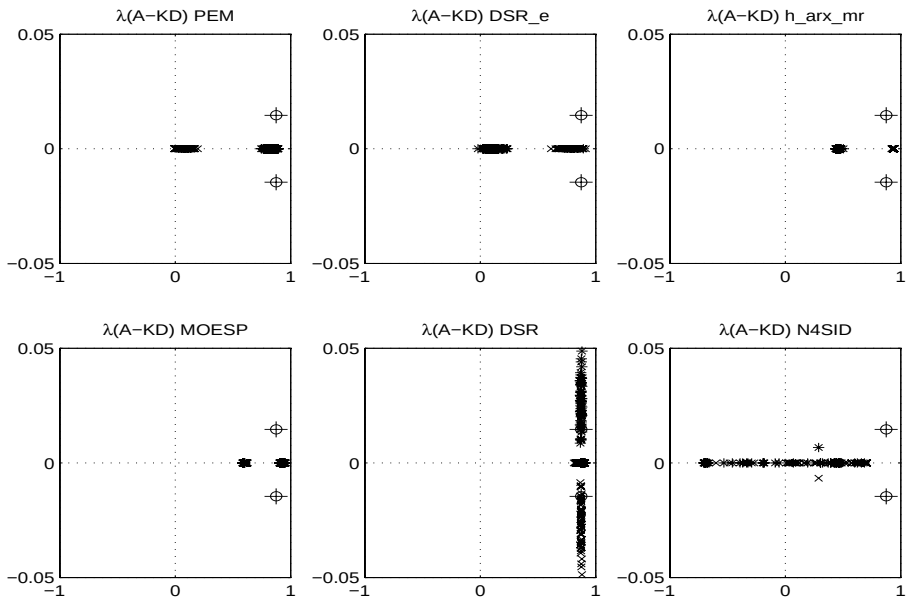

Figure 6.15: Estimated eigenvalues of the Kalman filter using a constant reference superposed with a sinusoid signal, $r_k^3 = 1 + 0.1\sin(\omega k)$ with $\omega = 0.725$, in the closed loop system, Example 2, Section 3.2. PEM and N4SID are used with default parameters and $n_k = 1$. DSR_e is used with $g = 0$, $L = 9$ and $J = 11$. In h_arx_mr the model order of the higher order ARX model used is 9 together with $L = 9$ and $J = 9$. MOESP is used with $s = 11$. DSR is used with $g = 0$, $L = 11$ and $J = 12$.

The methods that give the best estimates of the transition zeros of the system are PEM and DSR. The reason for this is the need for signals with a higher order of persistent excitation in the other methods. The good results from DSR must also be connected to the fact that the frequency used for the sinusoid signal is the frequency which is optimal for DSR.

DSR also gives a reasonable estimate of the eigenvalues of the Kalman filter. It is not possible to state if this is random or due to the fact that the frequency used for the sinusoid signal is the frequency which is optimal for DSR.

Using a constant reference superposed with a PRBS

The use of a reference signal consisting of a constant value superposed with a sinusoid signal leads to the assumption that except for PEM and DSR_e, the other methods needed a signal with higher order of persistent excitation. Therefore the reference used now is a constant value superposed with a PRBS generated from Equation (4.71) with $B = 0.2$, introduced in Section 4.6.1.

PEM and N4SID are used with default parameters and $n_k = 1$. DSR_e is used with $g = 0$, $L = 6$ and $J = 15$. In h_arx_mr the model order of the higher order ARX model used is 6 together with $L = 6$ and $J = 6$. MOESP is used with $s = 13$. DSR is used with $g = 0$, $L = 9$ and $J = 10$. The estimated eigenvalues are shown in Figure 6.16. The estimated zeros are presented in Figure 6.17 and the estimated eigenvalues of the Kalman filter are shown in Figure 6.18.

In this case all the methods give unbiased estimates of the eigenvalues, but N4SID gives estimates with larger variance than the other methods. The estimates of the transition zeros from all the methods, except MOESP, are considered unbiased but PEM and DSR_e give the best estimates. None of the methods give a reasonable estimate of the eigenvalues of the Kalman filter.

6.3.3 Example 3

Example 3 used in this section was introduced in Section 3.3 together with the noise level, unit variance, and the reference consisting of white noise with variance 4. PEM and N4SID are used with default parameters and $n_k = 1$. DSR_e is used with $g = 0$, $L = 1$ and $J = 10$. In h_arx_mr the model order of the higher order ARX model used is 3 together with $L = 3$ and $J = 3$. MOESP is used with $s = 3$. DSR is used with $g = 0$, $L = 2$ and $J = 3$. The estimated eigenvalues are shown in Figure 6.19. There are no zeros in this process. The estimated eigenvalues of the Kalman filter are shown in Figure 6.20.

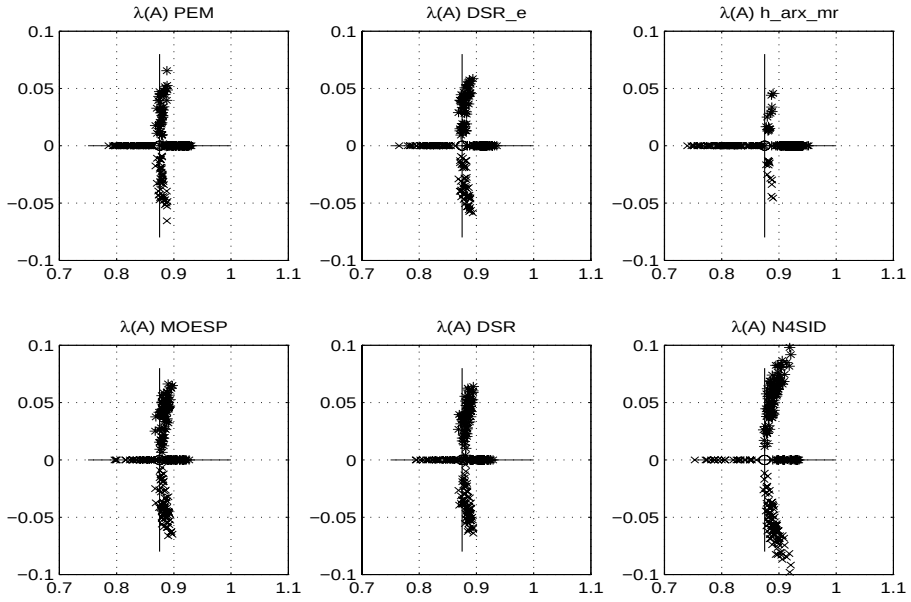

Figure 6.16: Eigenvalue estimates using a constant reference superposed with a PRBS, $R^4 = \text{ones}(N,1) + 0.1 \cdot \text{idinput}(N,'prbs',[0\ B])$ with $B = 0.2$, in the closed loop system, Example 2 Section 3.2. PEM and N4SID are used with default parameters and $n_k = 1$. DSR_e is used with $g = 0$, $L = 6$ and $J = 15$. In h_arx_mr the model order of the higher order ARX model used is 6 together with $L = 6$ and $J = 6$. MOESP is used with $s = 13$. DSR is used with $g = 0$, $L = 9$ and $J = 10$.

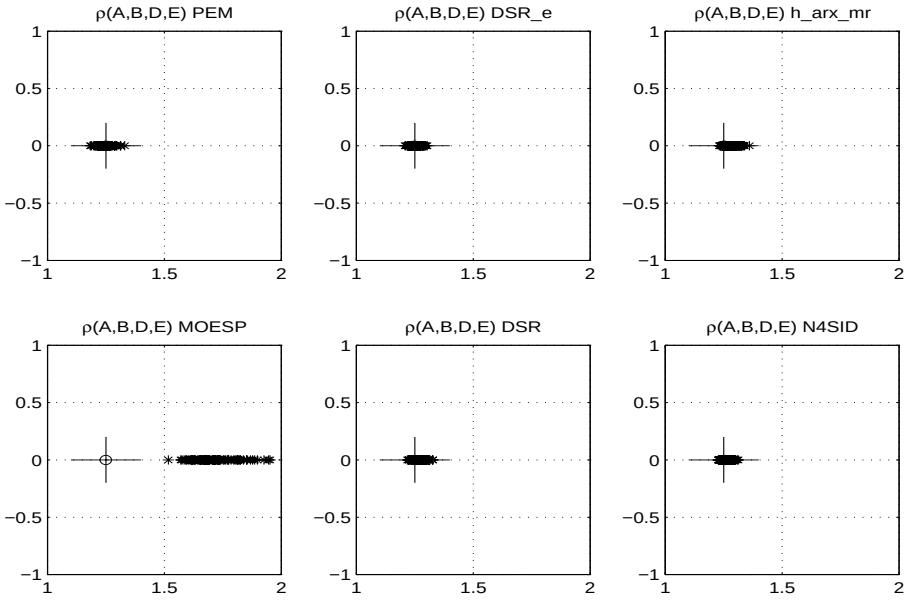

Figure 6.17: Estimated zeros using a constant reference superposed with a PRBS, $R^4 = \text{ones}(N,1) + 0.1 \cdot \text{idinput}(N,'prbs',[0\ B])$ with $B = 0.2$, in the closed loop system, Example 2 Section 3.2. PEM and N4SID are used with default parameters and $n_k = 1$. DSR_e is used with $g = 0$, $L = 6$ and $J = 15$. In h_arx_mr the model order of the higher order ARX model used is 6 together with $L = 6$ and $J = 6$. MOESP is used with $s = 13$. DSR is used with $g = 0$, $L = 9$ and $J = 10$.

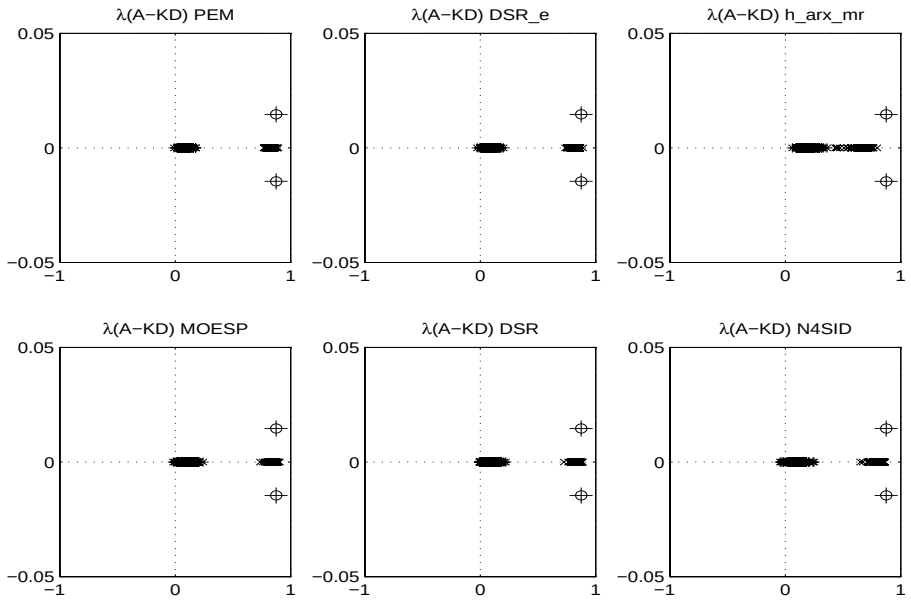

Figure 6.18: Estimated eigenvalues of the Kalman filter using a constant reference superposed with a PRBS, $R^4 = \text{ones}(N,1) + 0.1 \cdot \text{idinput}(N,'prbs',[0\ B])$ with $B = 0.2$, in the closed loop system, Example 2 Section 3.2. PEM and N4SID are used with default parameters and $n_k = 1$. DSR_e is used with $g = 0$, $L = 6$ and $J = 15$. In h_arx_mr the model order of the higher order ARX model used is 6 together with $L = 6$ and $J = 6$. MOESP is used with $s = 13$. DSR is used with $g = 0$, $L = 9$ and $J = 10$.

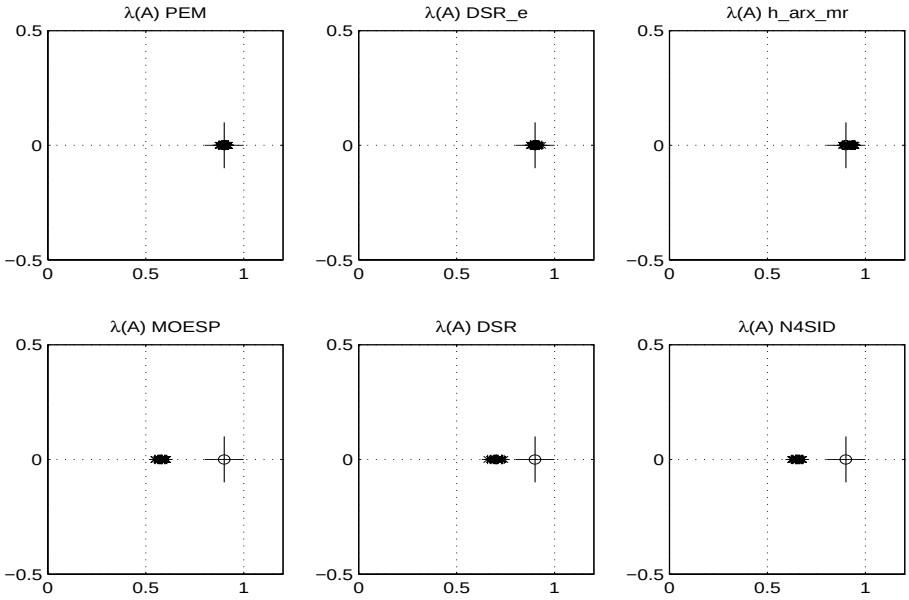

Figure 6.19: Eigenvalue estimates using a white noise reference with variance 4 in the closed loop system, Example 3, Section 3.3. PEM and N4SID are used with default parameters and $n_k = 1$. DSR_e is used with $g = 0$, $L = 1$ and $J = 10$. In h_arx_mr the model order of the higher order ARX model used is 3 together with $L = 3$ and $J = 3$. MOESP is used with $s = 3$. DSR is used with $g = 0$, $L = 2$ and $J = 3$.

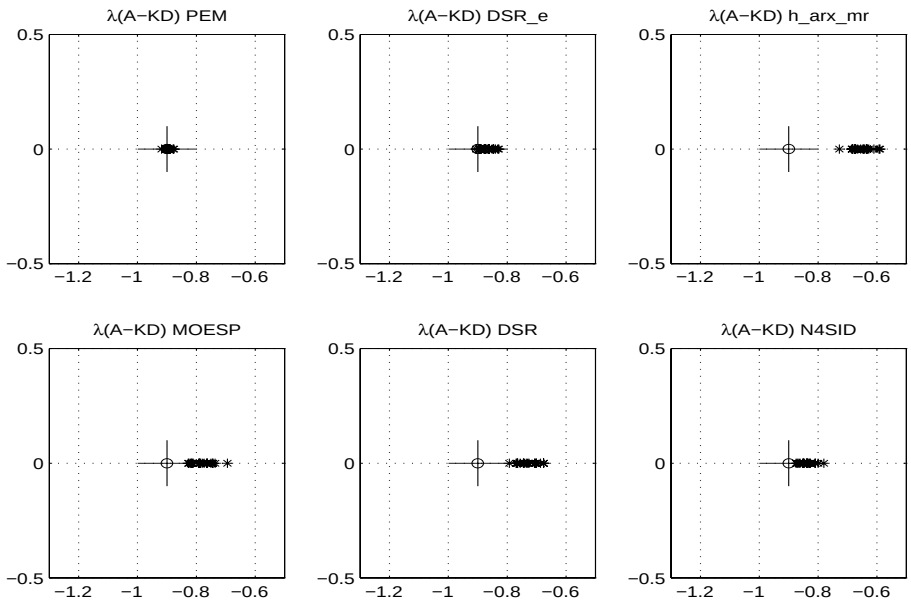

Figure 6.20: Estimated eigenvalues of the Kalman filter using a white noise reference with variance 4 in the closed loop system, Example 3, Section 3.3. PEM and N4SID are used with default parameters and $n_k = 1$. DSR_e is used with $g = 0$, $L = 1$ and $J = 10$. In h_arx_mr the model order of the higher order ARX model used is 3 together with $L = 3$ and $J = 3$. MOESP is used with $s = 3$. DSR is used with $g = 0$, $L = 2$ and $J = 3$.

PEM, DSR_e and h_arx_mr give unbiased estimates of the eigenvalues. The classic SID algorithms MOESP, DSR and N4SID all give biased estimates. In Section 4.5 it was shown that by using $g = 0$, $L = 1$ and $J = 250$ the eigenvalue estimates from DSR would be unbiased. The parameter setting is a special case and will not be used here.

PEM is the only method which gives unbiased estimates of the eigenvalues of the Kalman filter. DSR_e gives estimates of the eigenvalues of the Kalman filter that can be considered as unbiased. The classic SID algorithms MOESP, DSR and N4SID all give estimates with smaller bias than h_arx_mr. Since the classic SID algorithms give biased estimates of the eigenvalues of the system it is a coincidence that the estimates of the eigenvalues of the Kalman filter have smaller bias than the estimates from h_arx_mr.

6.3.4 Example 4

Example 4 used in this section was introduced in Section 3.4 together with the noise level. The reference, r_k^{10}, used is generated from Equation (3.21) with a white noise input with unit variance. Figure 6.21 shows the reference, r_k^{10}, with the corresponding input, u_k, and output, y_k, for a particular noise realization.

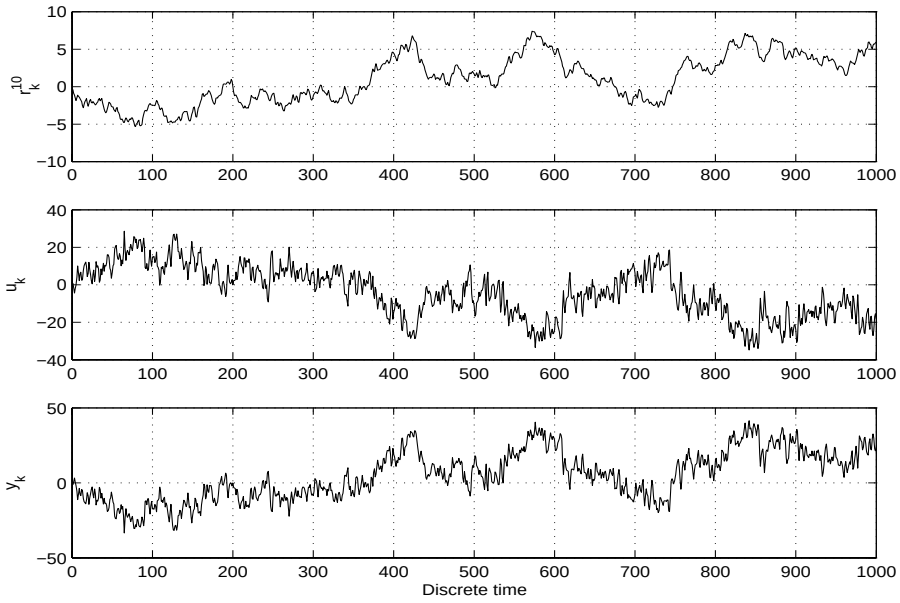

Figure 6.21: The reference, r_k^{10}, with the corresponding input, u_k, and output, y_k, for a particular noise realization in Example 4, Section 3.4

Choosing the parameter values which minimize the Squared Eigenvalue Error

criterion, V^1, gives poor estimation of the transition zeros of the system and the eigenvalues of the Kalman filter. An alternative is to consider the augmented state space model of the system, Equations (2.11) - (2.12), given by the matrices in Equation (3.20). The system to be estimated may be considered in innovation form. On observable canonical form the system matrices have the following structure

$$A = \begin{bmatrix} 0 & 1 \\ a_{21} & a_{22} \end{bmatrix}, B = \begin{bmatrix} b_{11} \\ b_{21} \end{bmatrix}, K = \begin{bmatrix} k_{11} \\ k_{21} \end{bmatrix}, D = \begin{bmatrix} 1 & 0 \end{bmatrix}. \tag{6.30}$$

The parameters to be estimated in the observable canonical form are collected in a parameter vector

$$\theta = [a_{21} \;\; a_{22} \;\; b_{11} \;\; b_{21} \;\; k_{11} \;\; k_{21}]. \tag{6.31}$$

The true parameter vector is

$$\theta = [0 \;\; 3 \;\; 2.5 \;\; 7.5 \;\; 3.6657 \;\; 10.6640]. \tag{6.32}$$

The parameter settings used in DSR_e, h_arx_mr, MOESP and DSR are the values which minimize the squared deviation from the true parameter vector, Equation (6.32). PEM and N4SID are used with default parameters and $n_k = 1$. DSR_e is used with $g = 0$, $L = 2$ and $J = 3$. In h_arx_mr the model order of the higher order ARX model used is 2 together with $L = 2$ and $J = 2$. MOESP is used with $s = 6$. DSR is used with $g = 0$, $L = 11$ and $J = 12$. The estimated eigenvalues are shown in Figure 6.22. Figure 6.23 shows the estimated transition zeros of the system. The estimated eigenvalues of the Kalman filter are shown in Figure 6.24.

PEM, DSR_e and h_arx_mr give unbiased estimates of the eigenvalues. The estimates from h_arx_mr and DSR_e have smaller variance than the estimates from PEM. The reason for this is that PEM is used with default parameters and DSR_e and h_arx_mr are used with a parameter which minimizes the squared deviation from the parameter vector Equation (6.32). The eigenvalue estimates from the ordinary SID algorithms have considerable bias. PEM and DSR_e are the only methods with unbiased estimates of the transition zeros. The estimates from h_arx_mr have a small bias and the ordinary SID algorithms give biased estimates. All the methods give biased estimates of the eigenvalues of the Kalman filter. PEM does not perform well which may lead to the conclusion that when PEM has problems estimating the eigenvalues of the Kalman filter there is no reason to believe that other methods will perform well. Another possibility is to again consider the capability to estimate the parameters in the parameter vector Equation (6.32) of the Kalman filter.

The plots of the estimated parameter vector for the respective methods are shown in Appendix A. The mean and standard deviation of the estimated parameter

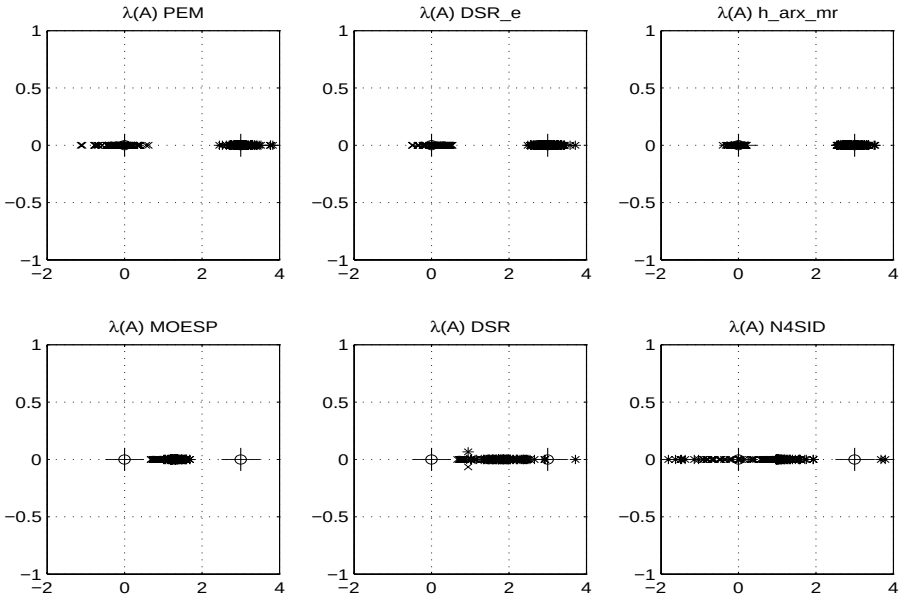

Figure 6.22: Eigenvalue estimates using a reference r_k^{10}, page 128, in the closed loop system, Example 4, Section 3.4. PEM and N4SID are used with default parameters and $n_k = 1$. DSR_e is used with $g = 0$, $L = 2$ and $J = 3$. In h_arx_mr the model order of the higher order ARX model used is 2 together with $L = 2$ and $J = 2$. MOESP is used with $s = 6$. DSR is used with $g = 0$, $L = 11$ and $J = 12$.

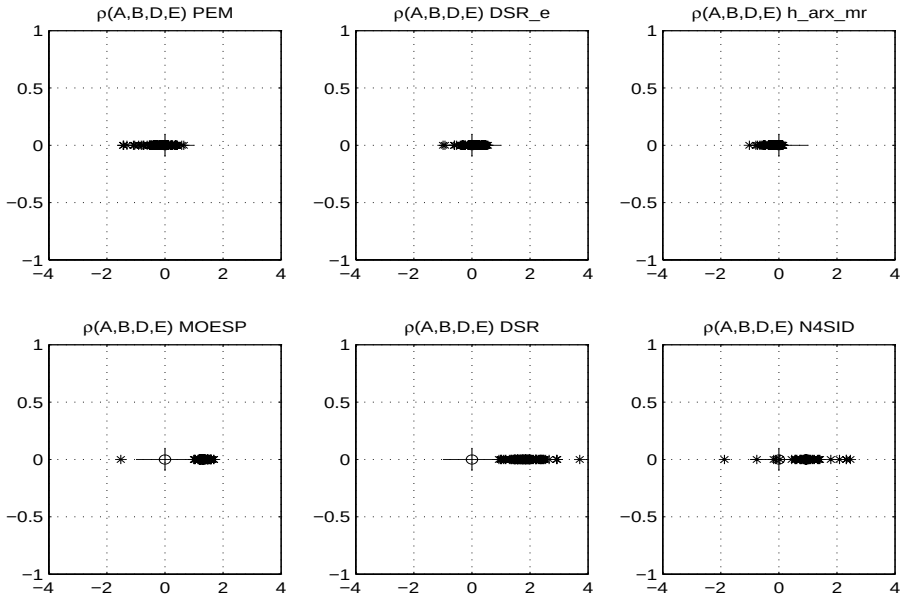

Figure 6.23: Estimated zeros using a reference r_k^{10}, page 128, in the closed loop system, Example 4, Section 3.4. PEM and N4SID are used with default parameters and $n_k = 1$. DSR_e is used with $g = 0$, $L = 2$ and $J = 3$. In h_arx_mr the model order of the higher order ARX model used is 2 together with $L = 2$ and $J = 2$. MOESP is used with $s = 6$. DSR is used with $g = 0$, $L = 11$ and $J = 12$.

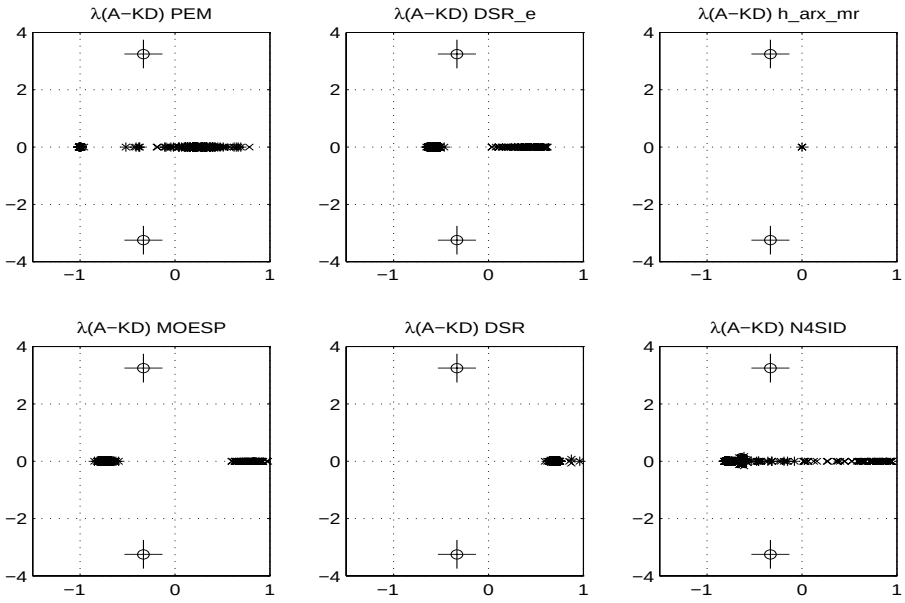

Figure 6.24: Estimated eigenvalues of the Kalman filter using a reference r_k^{10}, page 128, in the closed loop system, Example 4, Section 3.4. PEM and N4SID are used with default parameters and $n_k = 1$. DSR_e is used with $g = 0$, $L = 2$ and $J = 3$. In h_arx_mr the model order of the higher order ARX model used is 2 together with $L = 2$ and $J = 2$. MOESP is used with $s = 6$. DSR is used with $g = 0$, $L = 11$ and $J = 12$.

Table 6.1: Mean values from the Monte Carlo simulation of Example 4, Section 3.4, using the reference r_k^{10}, page 128. PEM is used with default parameters, except $n = 2$ and $n_k = 1$, for identification. DSR_e is used with $n = 2$, $g = 0$, $L = 2$ and $J = 3$. The model order of the higher order ARX model in h_arx_mr is 2 together with $L = 2$ and $J = 2$. MOESP is used with $n = 2$ and $s = 6$. DSR is used with $n = 2$, $g = 0$, $L = 11$ and $J = 12$. N4SID is used with default parameters, except $n = 2$ and $n_k = 1$.

	a_{21}	a_{22}	b_{11}	b_{21}	k_{11}	k_{21}
PEM	0.4012	2.9216	2.5245	7.8787	3.6884	11.0685
DSR_e	-0.4546	3.1332	2.3965	7.3511	3.3089	9.8089
h_arx_mr	0.0355	2.9466	2.1385	6.7643	2.9466	8.8394
MOESP	-1.0916	2.1399	-0.1936	-0.1607	2.0718	2.8077
DSR	-1.4932	2.6524	-0.2310	-0.1850	0.1096	0.0876
N4SID	-1.1761	2.2151	1.3442	59.2547	2.7317	62.6588

vector are shown in Table 6.1 and Table 6.2.

PEM and DSR_e are the only methods that give estimates of the parameters in the parameter vector Equation (6.31) which can be considered as unbiased. The estimates from h_arx_mr have a small bias. The ordinary SID algorithms all have a considerable bias. This is very clear in the plots in Appendix A where only PEM, DSR_e and h_arx_mr are plotted using the same axes. When plotting the results from the rest of the methods other axes have to be used.

By considering the estimates of the parameter vector Equation (6.32) of the full Kalman filter it is clear that the estimates from PEM and DSR_e are unbiased, even if the estimated eigenvalues of the Kalman filter indicate the opposite. Even if the methods gives consistent estimates of the parameter vector, Equation (6.32), small changes in the parameters can lead to large deviations in the calculations of the eigenvalues, the transition zeros or the eigenvalues of the Kalman filter. The estimates of the parameter vector Equation (6.32) also shows that h_arx_mr provides good estimates with a small bias.

If the system had been analysed by considering the frequency response of the estimated deterministic transfer function Equation (2.18) from input to output of the system, like in Chiuso et al. (2004), the problems regarding the Kalman filter would never been revealed or discussed. In addition when a closed loop SID algorithm is analysed the capability to estimate the full Kalman filter has to be considered to evaluate whether the method is comparable to PEM.

Table 6.2: Standard deviation from the Monte Carlo simulation of Example 4, Section 3.4, using the reference r_k^{10}, page 128. PEM is used with default parameters, except $n = 2$ and $n_k = 1$, for identification. DSR_e is used with $n = 2$, $g = 0$, $L = 2$ and $J = 3$. The model order of the higher order ARX model in h_arx_mr is 2 together with $L = 2$ and $J = 2$. MOESP is used with $n = 2$ and $s = 6$. DSR is used with $n = 2$, $g = 0$, $L = 11$ and $J = 12$. N4SID is used with default parameters, except $n = 2$ and $n_k = 1$.

	a_{21}	a_{22}	b_{11}	b_{21}	k_{11}	k_{21}
PEM	0.9656	0.4038	0.3749	1.7457	0.3871	2.2365
DSR_e	0.6989	0.4027	0.3686	1.6246	0.3680	1.9524
h_arx_mr	0.3734	0.3506	0.3515	1.4379	0.3506	1.7154
MOESP	0.1592	0.1744	0.0527	0.0400	0.2723	0.8480
DSR	0.3868	0.4576	0.0361	0.0243	0.0077	0.0135
N4SID	7.4122	7.6741	7.5801	370.2546	7.6516	385.7195

6.3.5 Example 5

Example 5 used in this section was introduced in Section 3.5 together with the corresponding noise level. The reference was presented in Section 5.4.2, Figure 5.19. Like in the previous examples a Monte Carlo simulation with 100 runs with different noise realization in each run is carried out. The system order $n = 3$ and the fact that the matrix E is the zero matrix is assumed to be known. PEM and N4SID are used with default parameters and $nk = 1$. It is not possible to find a parameter choice for the classic SID algorithms, or h_arx_mr, which results in unbiased estimates. For simplicity the following parameter settings are used. DSR_e is used with $g = 0$, $L = 10$ and $J = 10$. In h_arx_mr the model order of the higher order ARX model used is 10 together with $L = 10$ and $J = 10$. MOESP is used with $s = 10$. DSR is used with $g = 0$, $L = 10$ and $J = 10$. Figure 6.25 shows the estimated eigenvalues.

PEM and DSR_e give unbiased eigenvalue estimates with comparable variance. The eigenvalue estimates from h_arx_mr, MOESP, DSR and N4SID are of no use.

Figure 6.26 shows the estimated zeros. All the methods give biased estimates of the zeros. The bias is not visible in the figure due to the axes used. DSR_e is comparable to PEM. MOESP, DSR, N4SID and h_arx_mr estimate the zeros with smaller bias and variance than PEM and DSR_e. This has to be a coincidence since the eigenvalues are needed to estimate the zeros in these subspace methods.

Figure 6.27 shows the estimated eigenvalues of the Kalman filter.

PEM and DSR_e give unbiased estimates of the eigenvalues of the Kalman filter.

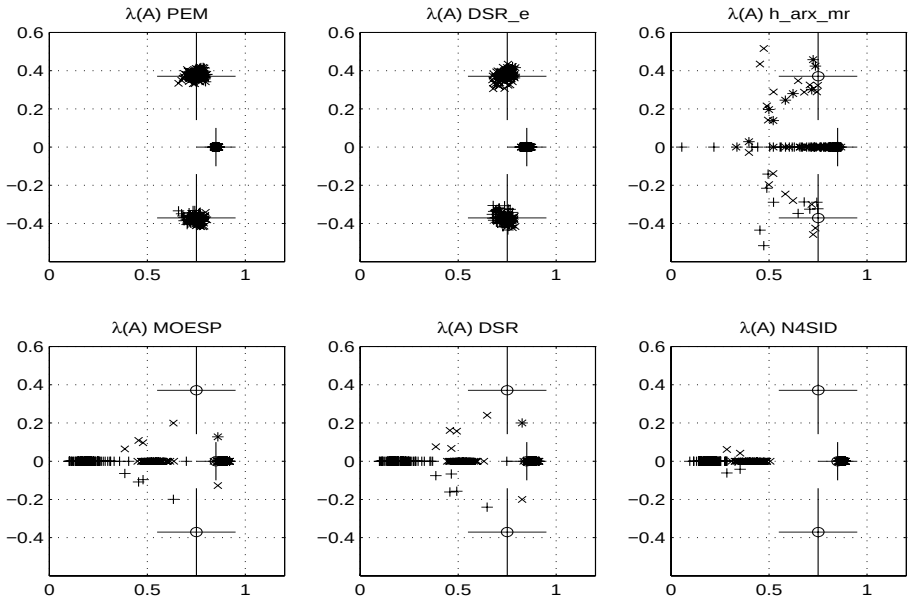

Figure 6.25: The estimated eigenvalues when r_k^7 and r_k^8, page 90, are used as reference in the closed loop simulation, Example 5, Section 3.5. PEM and N4SID are used with default parameters and $nk = 1$. DSR_e is used with $g = 0$, $L = 10$ and $J = 10$. In h_arx_mr the model order of the higher order ARX model used is 10 together with $L = 10$ and $J = 10$. MOESP is used with $s = 10$. DSR is used with $g = 0$, $L = 10$ and $J = 10$.

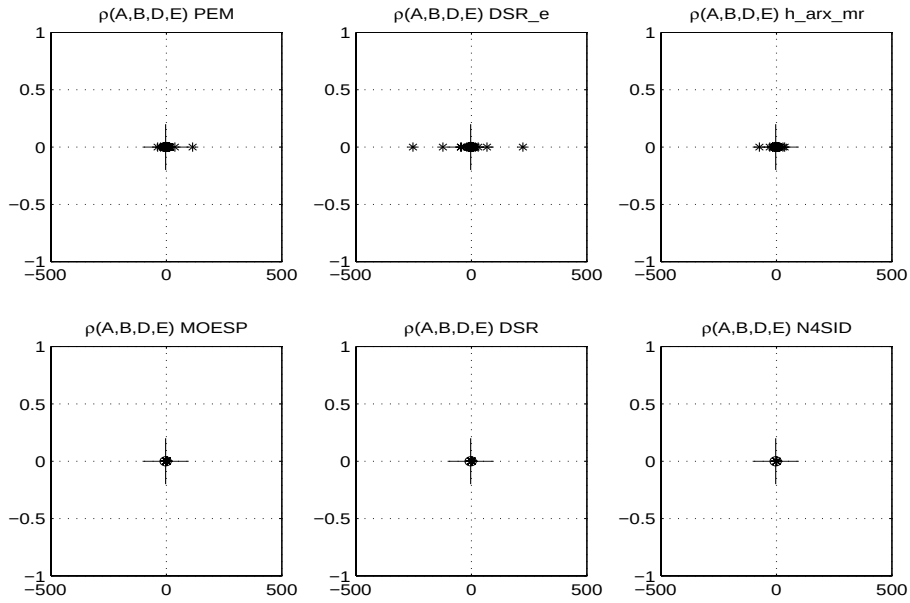

Figure 6.26: The estimated zeros when r_k^7 and r_k^8, page 90, are used as reference in the closed loop simulation, Example 5, Section 3.5. PEM and N4SID are used with default parameters and $nk = 1$. DSR_e is used with $g = 0$, $L = 10$ and $J = 10$. In h_arx_mr the model order of the higher order ARX model used is 10 together with $L = 10$ and $J = 10$. MOESP is used with $s = 10$. DSR is used with $g = 0$, $L = 10$ and $J = 10$.

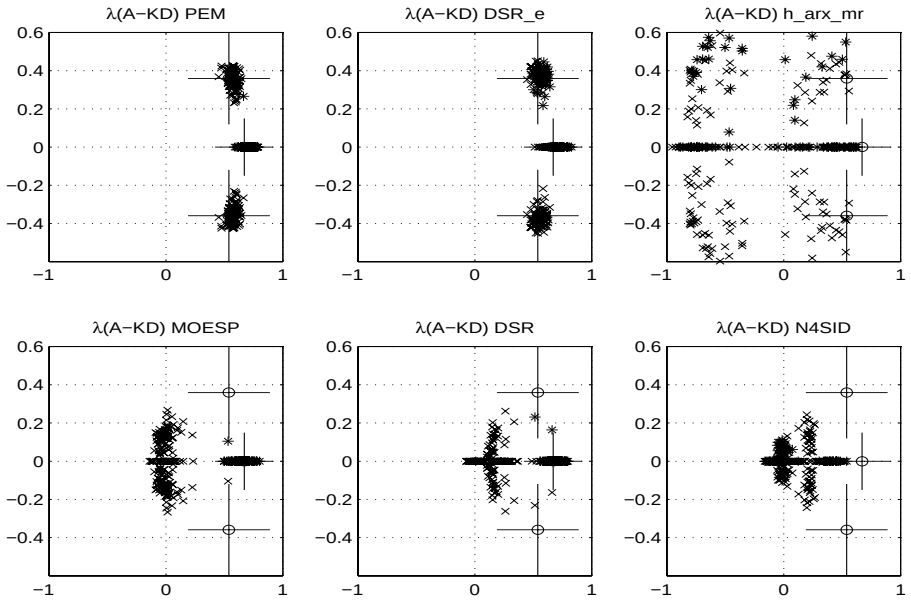

Figure 6.27: The estimated eigenvalues of the Kalman filter when r_k^7 and r_k^8, page 90, are used as reference in the closed loop simulation, Example 5, Section 3.5. PEM and N4SID are used with default parameters and $nk = 1$. DSR_e is used with $g = 0$, $L = 10$ and $J = 10$. In h_arx_mr the model order of the higher order ARX model used is 10 together with $L = 10$ and $J = 10$. MOESP is used with $s = 10$. DSR is used with $g = 0$, $L = 10$ and $J = 10$.

The rest of the methods give biased estimates.

Also in this case DSR_e is comparable to PEM. In previous sections a reasonable parameter choice have been to choose a small future horizon, L, when there is considerable noise. The past horizon J has to be chosen larger than when DSR is used on open loop data. In this case a large J has been chosen. If a small L, for instance $L = 3$, is chosen a small bias is introduced in the eigenvalue estimates. The estimated zeros and the estimated eigenvalues of the Kalman filter are still unbiased. If $L = 5$ is used the results are unbiased as when $L = 10$. The plots for the cases where $L = 3$ and $L = 5$ are shown in Appendix B. This indicates that choosing a minimum value of L can lead to a small bias.

In this case h_arx_mr does not work. The method needs inputs with a considerably higher order of persistent excitation than PEM and DSR_e. The reason for this is the large number of parameters to be estimated in the high order ARX model.

6.4 Automatic identification of system order

In the previous sections the parameter choice in both DSR and DSR_e has been based on knowledge regarding the eigenvalues or the full state space model of the system. This section presents different methods to estimate the system order. The methods are meant as help for users without experience in using SID algorithms. The methods will be compared to each other and PEM. The model found by PEM implemented in Matlab 6.5 is initialized by N4SID, and then further adjusted by optimizing the prediction error fit if the system order is not specified. The system order is assumed to be lower than 10.

Some basic assumptions have to be stated before the methods can be presented. The first assumption is that when there is no knowledge regarding the process, a model of higher order than 10 will not be chosen. The past horizon will be chosen larger than the future horizon, but never larger than 20. If two models have approximately the same performance, the model with the lowest order is preferred. This is inspired by Akaike's Information Criterion, Ljung (1999), which gives a tradeoff between performance and model complexity. All methods are programmed for both DSR and DSR_e, but only the functionality related to DSR_e will be discussed due to the focus on closed loop functionality.

It is natural to use the built-in functionality in DSR_e. A search is performed on the logarithm of the singular values to identify the the first significant drop in numeric value. This drop indicates the system order. When this functionality is used it is referred to as DSR_e. In all examples the future horizon chosen is

$L = 5$ and the past horizon chosen is $J = 10$.

An alternative approach is to utilize the squared prediction error when DSR_e is used for system identification. Appendix C contains the Matlab function orderfindPE.m. The parameters which can be chosen by the user are $meth$, g, val and n. If $meth = 0$, DSR is used for identification. When $meth = 1$, DSR_e is used for identification. The parameter g is the structure parameter, which has to be zero when DSR_e is used on closed loop data. If $val = 0$, all data points are used for identification. If $val = 1$, 75% of the data points are used for identification and 25% of the data points are used for validation. A search is performed for the future horizon in the interval $\frac{5}{m} \leq L \leq 10$ and the past horizon in the interval $10 \leq J \leq 20$ to find the system with the smallest squared prediction error. Then the systems with up to 1% larger squared prediction error are evaluated, and the system with lowest system order is chosen. If the user has chosen the system order n, the past and future horizon which give the smallest squared prediction error are chosen. The parameters used in this thesis are $meth = 1$, $val = 1$ and the system order is not chosen, unless something else is specified.

The squared simulated error when DSR_e is used for system identification can also be utilized. Appendix D contains the Matlab function orderfindSE.m. The parameters in orderfindSE.m are the same as in orderfindPE.m. The parameters used in this thesis are $meth = 1$, $val = 1$ and the system order is not chosen, unless something else is specified. A search is performed for the future horizon in the interval $\frac{5}{m} \leq L \leq 10$ and the past horizon in the interval $10 \leq J \leq 20$ to find the system with the smallest squared simulated error. Then the systems with up to 1% larger squared simulated error are evaluated, and the system with lowest system order is chosen. If the user has chosen the system order n, the past and future horizons which give the smallest squared simulated error are chosen. The parameters used in this thesis are $meth = 1$, $val = 1$ and the system order is not chosen, unless something else is specified.

The methods are tested on all the five examples that were introduced. The fact that there is no direct feedthrough from input to output, the matrix E is the zero matrix, is assumed known. Table 6.3 contains the results from the different methods used on data from a Monte Carlo simulation performed on the model introduced in Example 1, Section 3.1, using the low frequent PRBS, r_k^1 page 33, as reference.

DSR_e estimates the system order correct 100% of the times. Also orderfindSE.m and orderfindPE.m performs well with correctly estimated system order respectively 83% and 68% of the time. PEM only estimates the system order correctly 14% of the time.

Table 6.3: The results from the methods used on data from a Monte Carlo simulation performed on the model introduced in Example 1, Section 3.1, using the low frequent PRBS, r_k^1 page 33, as reference.

Method	$n = 1$	$n = 2$	$n = 3$	$n = 4$	$n \geq 5$
DSR_e	0	100	0	0	0
orderfindPE	0	68	19	10	3
orderfindSE	5	83	12	0	0
PEM	0	14	42	14	30

Table 6.4 contains the results from the different methods used on data from a Monte Carlo simulation performed on the model introduced in Example 1, Section 3.1, using the high frequent PRBS, r_k^6 page 72, as reference.

Table 6.4: The results from the methods used on data from a Monte Carlo simulation performed on the model introduced in Example 1, Section 3.1, using the high frequent PRBS, r_k^6 page 72, as reference.

Method	$n = 1$	$n = 2$	$n = 3$	$n = 4$	$n \geq 5$
DSR_e	0	100	0	0	0
orderfindPE	0	75	13	9	3
orderfindSE	0	95	5	0	0
PEM	0	82	9	6	3

The system order is estimated correct 100% of the time by DSR_e. Also orderfindSE.m and orderfindPE.m perform well with correctly estimated system order respectively 95% and 75% of the time. PEM estimates the system order correctly 82% of the time.

Table 6.5 contains the results from the different methods used on data from a Monte Carlo simulation performed on the model introduced in Example 2, Section 3.2, using a constant reference superposed a low frequent PRBS. Figure 4.11, page 39, shows the reference signal, r_k and the corresponding input, u_k, and output, y_k, for two particular noise realizations v_k and w_k.

DSR_e was not able to estimate the system order correctly. The function orderfindPE.m performs well with correct estimated system order 70% of the time. Here orderfindSE.m only estimates the system order correct 1% of the time. PEM estimates the system order correctly 87% of the times. It has to be mentioned that if DSR_e had been used with $L = 10$ and $J = 15$ the system order would have been estimated correct 94% of the time.

Table 6.5: The results from the methods used on data from a Monte Carlo simulation performed on the model introduced in Example 2, Section 3.2, using a constant reference superposed a low frequent PRBS, Figure 4.11 page 39.

Method	$n = 1$	$n = 2$	$n = 3$	$n = 4$	$n \geq 5$
DSR_e	100	0	0	0	0
orderfindPE	1	70	11	12	6
orderfindSE	24	1	47	12	16
PEM	9	87	4	0	0

Table 6.6 contains the results from the different methods used on data from a Monte Carlo simulation performed on the model introduced in Example 2, Section 3.2, using a constant reference superposed a high frequent PRBS. Figure 6.28 shows the reference signal, r_k^{11}, and the corresponding input, u_k, and output, y_k, for two particular noise realizations v_k and w_k.

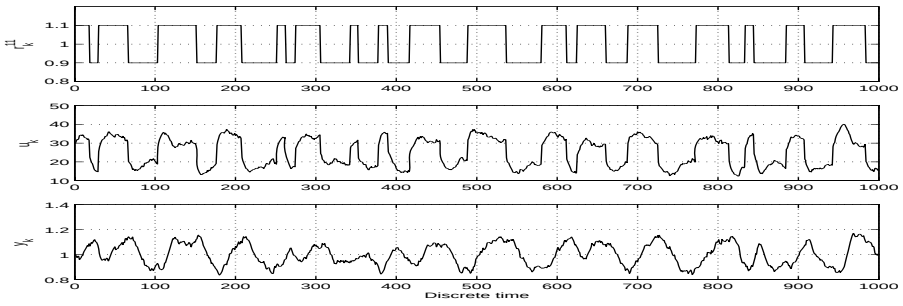

Figure 6.28: The reference signal, r_k^{11}, using a constant reference superposed a high frequent PRBS and the corresponding input, u_k, and output, y_k, for two particular noise realizations v_k and w_k.

Table 6.6: The results from the methods used on data from a Monte Carlo simulation performed on the model introduced in Example 2, Section 3.2, using a constant reference superposed a high frequent PRBS, Figure 6.28 page 141.

Method	$n = 1$	$n = 2$	$n = 3$	$n = 4$	$n \geq 5$
DSR_e	4	96	0	0	0
orderfindPE	0	68	18	10	4
orderfindSE	0	9	61	6	24
PEM	0	93	0	6	1

The system order is estimated correct 96% of the time when DSR_e is used. Here orderfindPE.m performs well with correct estimated system order 68% of the

time. But orderfindSE.m only estimates the system order correct 9% of the time. PEM estimates the system order correctly 93% of the time.

Table 6.7 contains the results from the different methods used on data from a Monte Carlo simulation performed on the 1^{st} order model introduced in Example 3, Section 3.3, using a Gaussian white noise with variance 4 as reference.

Table 6.7: The results from the methods used on data from a Monte Carlo simulation performed on the 1^{st} order model introduced in Example 3, Section 3.3, using a Gaussian white noise with variance 4 as reference.

Method	$n = 1$	$n = 2$	$n = 3$	$n = 4$	$n \geq 5$
DSR_e	78	22	0	0	0
orderfindPE	80	6	9	5	0
orderfindSE	0	100	0	0	0
PEM	0	0	0	4	96

DSR_e estimates the system order correct 78% of the time. Here orderfindPE.m estimates the system order correct 80% of the time. But orderfindSE.m is not able to estimate the system order correctly. PEM never estimates the system order correctly.

Table 6.8 contains the results from the different methods used on data from a Monte Carlo simulation performed on the model introduced in Example 4, Section 3.4, using r_k^{10} as reference.

Table 6.8: The results from the methods used on data from a Monte Carlo simulation performed on the model introduced in Example 4, Section 3.4, using r_k^{10}, page 128, as reference.

Method	$n = 1$	$n = 2$	$n = 3$	$n = 4$	$n \geq 5$
DSR_e	15	48	37	0	0
orderfindPE	10	69	12	4	5
orderfindSE	73	24	3	0	0
PEM	0	0	0	15	85

DSR_e estimates the system order correct 48% of the time. Here orderfindPE.m performs well with correct estimated system order 69% of the time. But orderfindSE.m only estimates the system order correctly 24% of the time. PEM never estimates the system order correctly.

Table 6.9 contains the results from the different methods used on data from a Monte Carlo simulation performed on the 3^{rd} order multiple input multiple output model introduced in Example 5, Section 3.5, using r_k^7 and r_k^8 as references.

Table 6.9: The results from the methods used on data from a Monte Carlo simulation performed on the 3^{rd} order multiple input multiple output model introduced in Example 5, Section 3.5, using r_k^7 and r_k^8, page 90, as references.

Method	$n = 1$	$n = 2$	$n = 3$	$n = 4$	$n \geq 5$
DSR_e	4	96	0	0	0
orderfindPE	0	6	68	15	11
orderfindSE	0	20	23	25	32
PEM	0	1	18	53	28

DSR_e was not able to estimate the system order correctly. Here orderfindPE.m performs well with correct estimated system order 68% of the time. But orderfindSE.m only estimates the system order correct 23% of the time. PEM estimates the system order correct 18% of the time. It has to be mentioned that if DSR_e had been used with $L = 10$ and $J = 15$ the system order would have been estimated correct 67% of the time.

The function orderfindPE.m has the best overall performance in the simulations so far. It had never less than 68% correct estimates of the system order. DSR_e worked well in most cases, except for the multiple input multiple output case where it failed totally. The results where DSR_e is used with other horizons than the horizons chosen initially cannot be used in this comparison because the results are based on knowledge of using SID algorithms. It has to be mentioned that there are only one case where PEM performs best. This simulation study indicates that the solutions presented here are not suitable to replace inspection of singular values and validation. But the function orderfindPE.m may be a help for users who are not familiar with the use of SID algorithms.

6.5 A closed loop system identification procedure

The simulations using the functions introduced in Section 6.4 show that none of the functions work perfectly. Therefore a direct closed loop system identification procedure combining the visual inspection of the singular values and the search for the minimum prediction error is suggested.

The procedure can be described by the following three steps:

- Step 1. Identify the system order using the singular value plot from DSR_e
- Step 2. Use the orderfindPE.m function with $val = 1$ to find the parameter setting in DSR_e which gives the minimum squared prediction error on validation data
- Step 3. Use DSR_e to identify the full state space model inclusive of the Kalman filter gain.

The singular values are stochastic variables. That is the reason why the estimated system order is not the same in all the simulations in the Monte Carlo simulations performed in Section 6.4. Therefore the estimated system order is not necessarily the same in each run of a Monte Carlo simulation. This is illustrated in Figure 6.29 which shows the distribution of the singular values from DSR_e with $L = 5$ and $J = 10$ when used on Example 1, Section 3.1, when a low frequent PRBS, r_k^1 page 33, is used as reference.

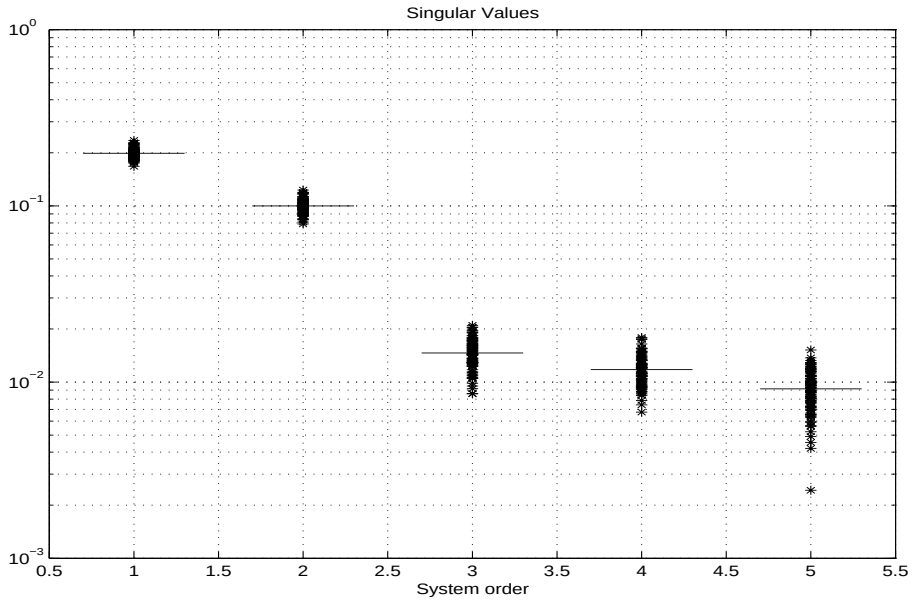

Figure 6.29: The distribution of the singular values from DSR_e with $L = 5$ and $J = 10$ when used on Example 1, Section 3.1, when a low frequent PRBS, r_k^1 page 33, is used as reference.

Appendix F contains the singular value distribution for the rest of the inputs and examples used in the previous sections.

When the procedure is tested, only the singular value plot of the first run of the Monte Carlo simulation is inspected for identification of the system order. The reason for this is that in a practical approach a Monte Carlo simulation will not be available.

The procedure is tested on all the five examples introduced. The fact that there is no direct feedthrough from input to output, the matrix E is the zero matrix, is assumed known. For DSR_e it is also assumed that the correct system order is estimated from the singular value plot in Step 1. It is assumed that the correct order is found for PEM by validation. All singular value plots used are with $L = 5$ and $J = 10$. Appendix E contains the corresponding plots with $L = 10$ and $J = 20$ to show that the choice of future and past horizons is not that significant when identifying the system order. The choice of past and future horizons found in step 2 and used in step 3 will not be listed.

Figure 6.30 shows the singular value plot from DSR_e with $L = 5$ and $J = 10$ when used on Example 1, Section 3.1, when a low frequent PRBS, r_k^1 page 33, is used as reference.

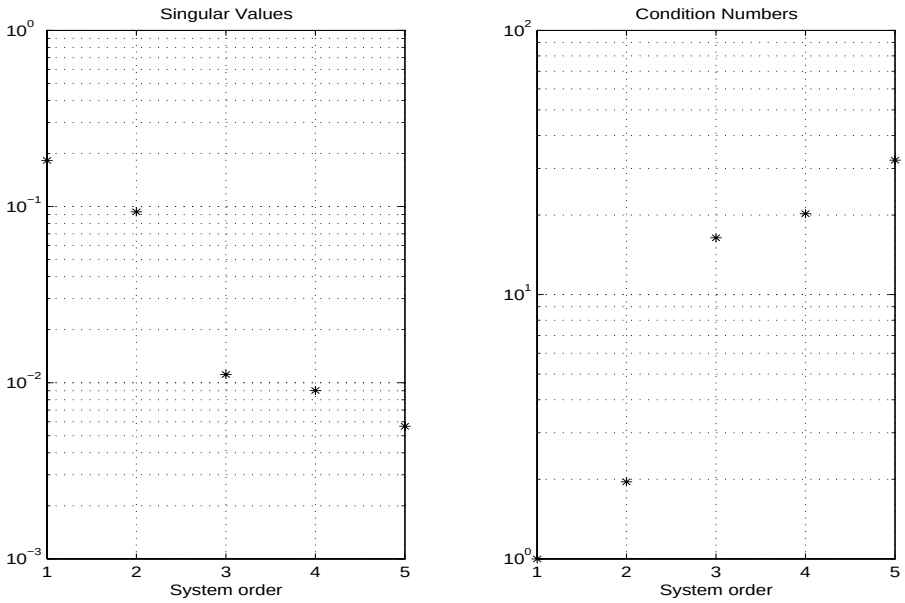

Figure 6.30: Singular value plot from DSR_e with $L = 5$ and $J = 10$ when used on Example 1, Section 3.1, when a low frequent PRBS, r_k^1 page 33, is used as reference.

The singular value plot indicates that this is a 2^{nd} order system. It is assumed that the correct model order, $n = 2$, is chosen. Figure E.1 gives the same result where $L = 10$ and $J = 20$ are chosen. The parameter estimates are visualized in

Figure 6.31.

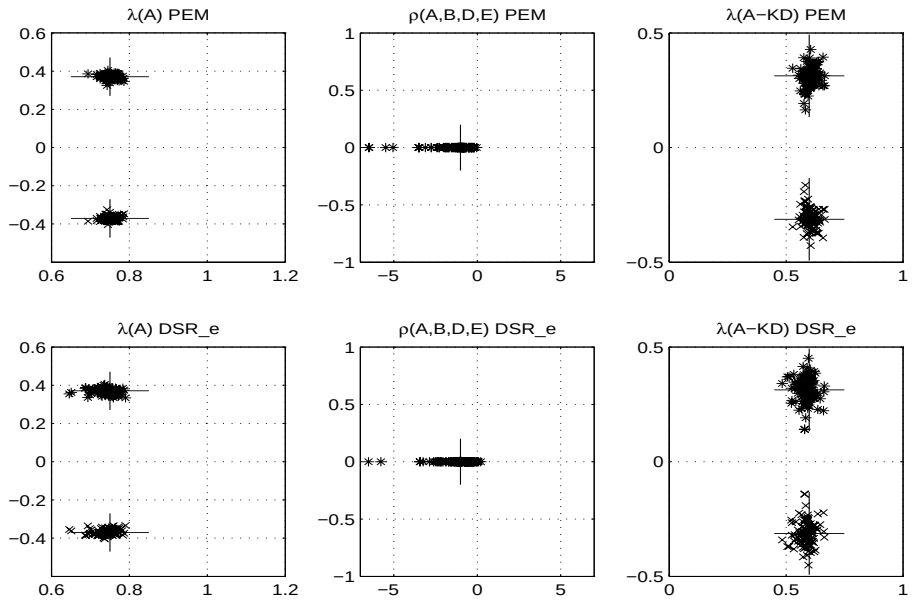

Figure 6.31: Estimates from DSR_e and PEM when used on Example 1, Section 3.1, when a low frequent PRBS, r_k^1 page 33, is used as reference.

All the estimates are unbiased and comparable to the results from PEM.

Figure 6.32 shows the singular value plot from DSR_e with $L = 5$ and $J = 10$ when used on Example 1, Section 3.1, when a high frequent PRBS, r_k^6 page 72, is used as reference.

The singular value plot also here indicates that this is a 2^{nd} order system. It is assumed that the correct model order, $n = 2$, is chosen. Figure E.2 gives the same result where $L = 10$ and $J = 20$ are chosen. The parameter estimates are visualized in Figure 6.33.

Also here all the estimates are unbiased and comparable to the results from PEM.

Figure 6.34 shows the singular value plot from DSR_e with $L = 5$ and $J = 10$ when used on Example 2, Section 3.2, using a constant reference superposed a low frequent PRBS, Figure 4.11 page 39.

The singular value plot indicates that this is a 1^{st} or 2^{nd} order system. It is assumed that the correct model order, $n = 2$, is chosen. Figure E.3 gives the same result where $L = 10$ and $J = 20$ are chosen. The parameter estimates are

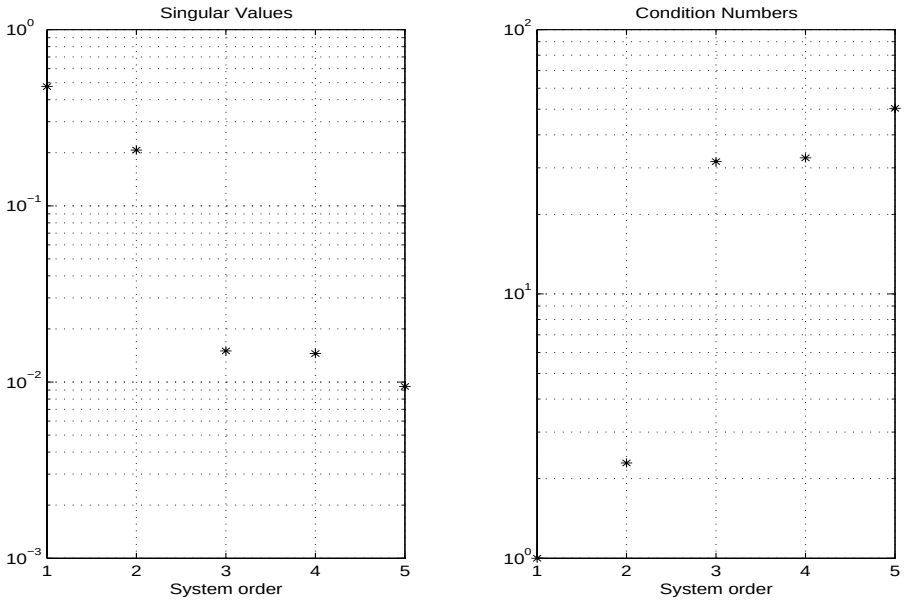

Figure 6.32: Singular value plot from DSR_e with $L = 5$ and $J = 10$ when used on Example 1, Section 3.1, when a high frequent PRBS, r_k^6 page 72, is used as reference.

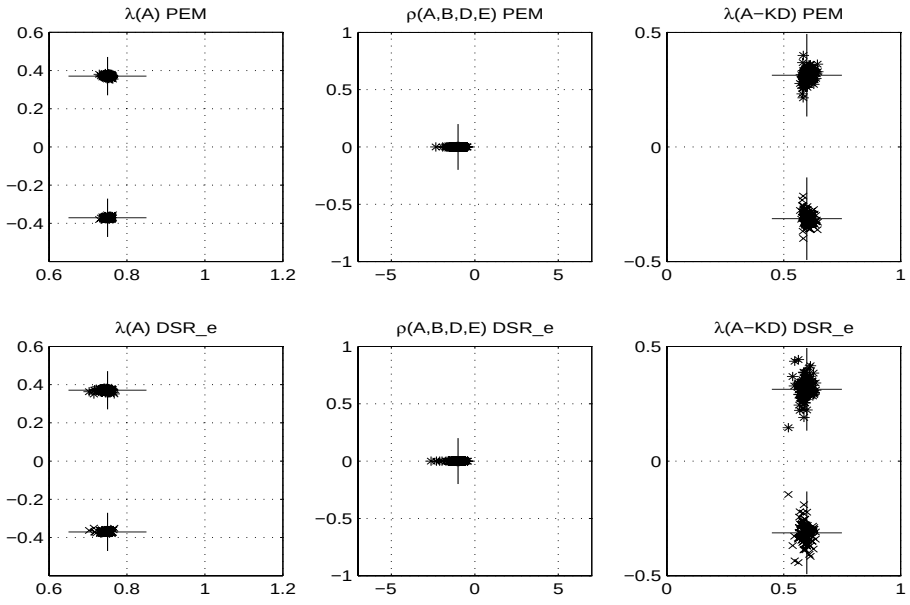

Figure 6.33: Estimates from DSR_e and PEM when used on Example 1, Section 3.1, when a high frequent PRBS, r_k^6 page 72, is used as reference.

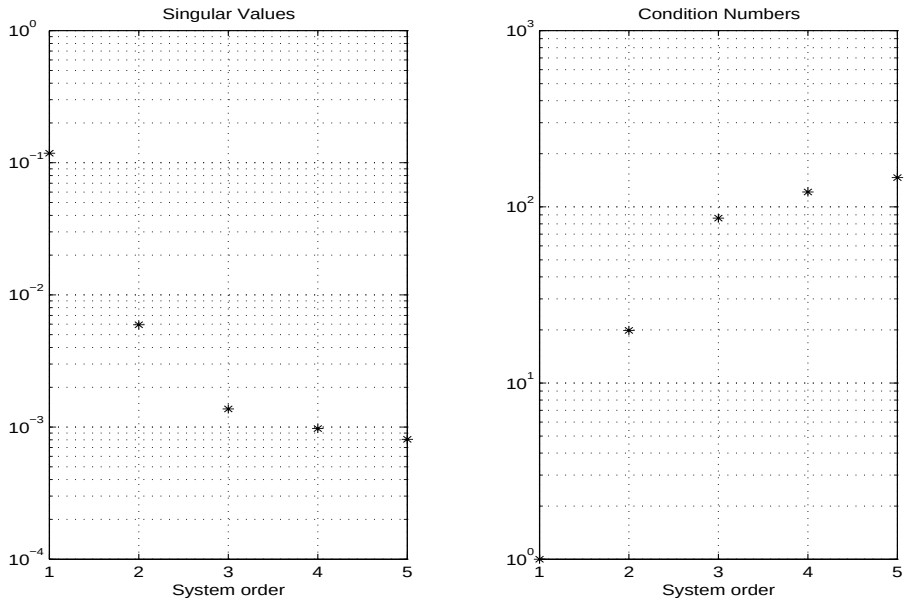

Figure 6.34: Singular value plot from DSR_e with $L = 5$ and $J = 10$ when used on Example 2, Section 3.2, with a constant reference superposed a low frequent PRBS, Figure 4.11 page 39.

visualized in Figure 6.35.

The results coincide with the results found in Section 6.3.2. The estimates of the eigenvalues are unbiased. In this case there is a small bias in the estimated zeros from DSR_e. The bias may be caused by the fact that the reference signal used here has a lower level of persistent excitation than the PRBS signal used in Section 6.3.2. In both cases the estimation of the eigenvalues of the Kalman filter failed.

Figure 6.36 shows the singular value plot from DSR_e with $L = 5$ and $J = 10$ when used on Example 2, Section 3.2, using a constant reference superposed a high frequent PRBS, Figure 6.28 page 141.

The singular value plot here also indicates that this is a 1^{st} or 2^{nd} order system. It is assumed that the correct model order, $n = 2$, is chosen. Figure E.4 gives the same result where $L = 10$ and $J = 20$ are chosen. The parameter estimates are visualized in Figure 6.37.

The results coincide with the results found in Section 6.3.2. The estimates of the eigenvalues are unbiased. In this case there is a small bias in the estimated zeros from DSR_e. The reference signal used here has a lower level of persistent excitation than the PRBS signal used in Section 6.3.2. Since the bias is reduced

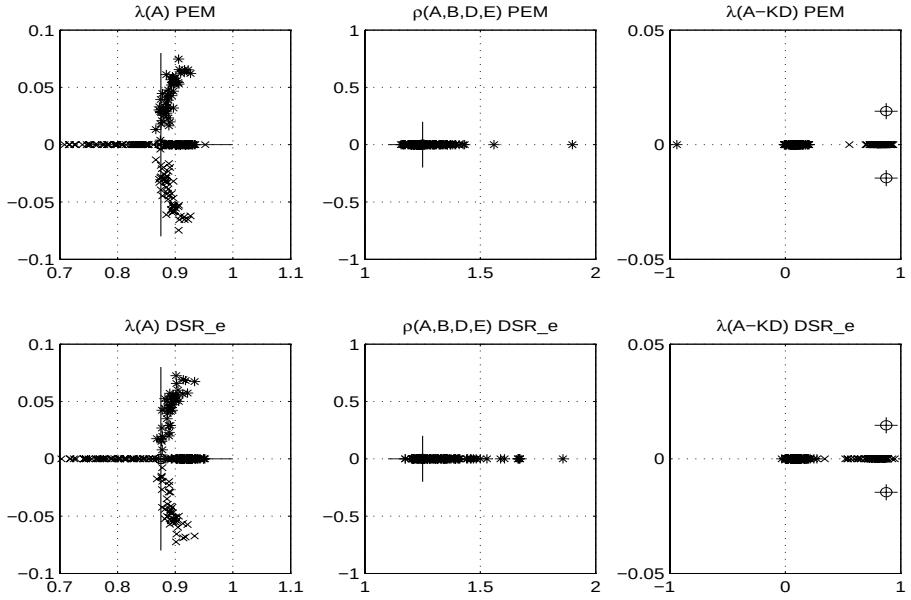

Figure 6.35: Estimates from DSR_e and PEM when used on Example 2, Section 3.2, when a low frequent PRBS, r_k^1 page 33, is used as reference.

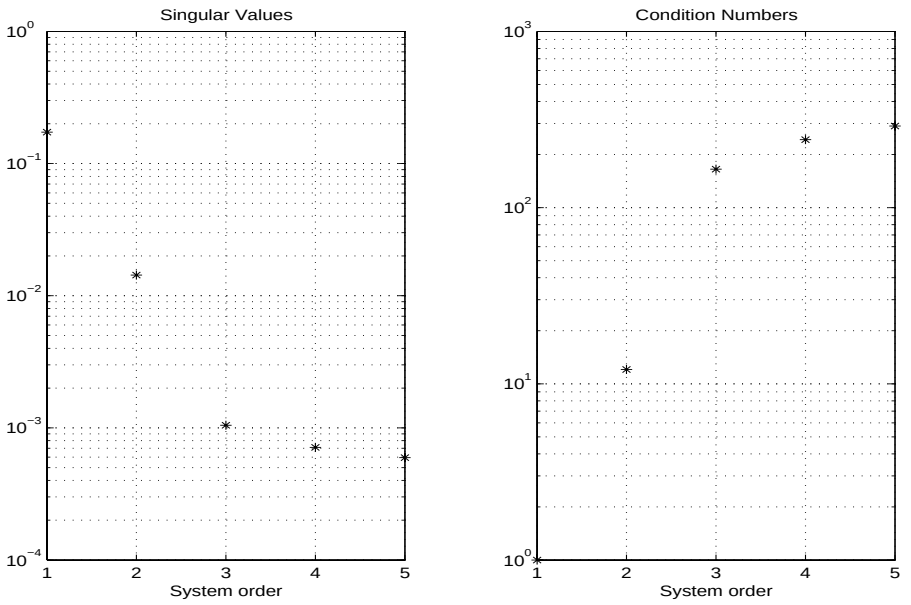

Figure 6.36: Singular value plot from DSR_e with $L = 5$ and $J = 10$ when used on Example 2, Section 3.2, with a constant reference superposed a high frequent PRBS, Figure 6.28 page 141.

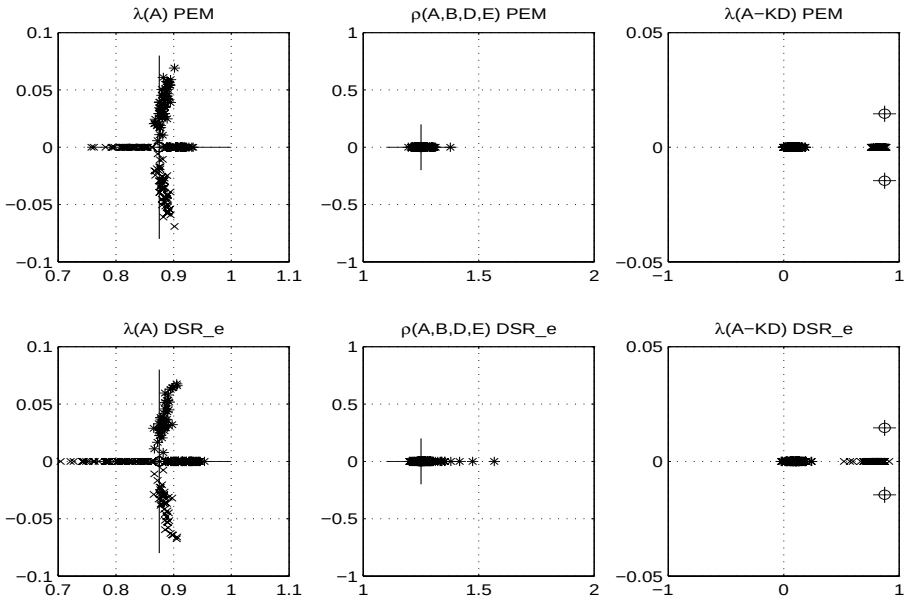

Figure 6.37: Estimates from DSR_e and PEM when used on Example 2, Section 3.2, with a constant reference superposed a high frequent PRBS, Figure 6.28 page 141.

compared to Figure 6.35 the low order of persistent excitation is assumed to be the cause of the bias in the zeros. Also here the estimation of the eigenvalues of the Kalman filter failed.

Figure 6.38 shows the singular value plot from DSR_e with $L = 5$ and $J = 10$ when used on Example 3, Section 3.3, when a Gaussian white noise with variance 4 is used as reference.

The singular value plot indicates that this is a 1^{st} or 3^{rd} order system. It is assumed that the correct model order, $n = 1$, is chosen. Figure E.5 where $L = 10$ and $J = 20$ also indicates that this is a 1^{st} or 3^{rd} order system. The parameter estimates are visualized in Figure 6.39.

The estimates of the eigenvalues are unbiased. In this case there is a small bias on the estimated eigenvalues of the Kalman filter from DSR_e. In Section 6.3.3 the estimated eigenvalues of the Kalman filter was unbiased for both DSR_e and PEM.

Figure 6.40 shows the singular value plot from DSR_e with $L = 5$ and $J = 10$ when used on Example 4, Section 3.4, using r_k^{10}, page 128, as reference.

The singular value plot indicates that this is a 1^{st} or 2^{nd} order system. It is

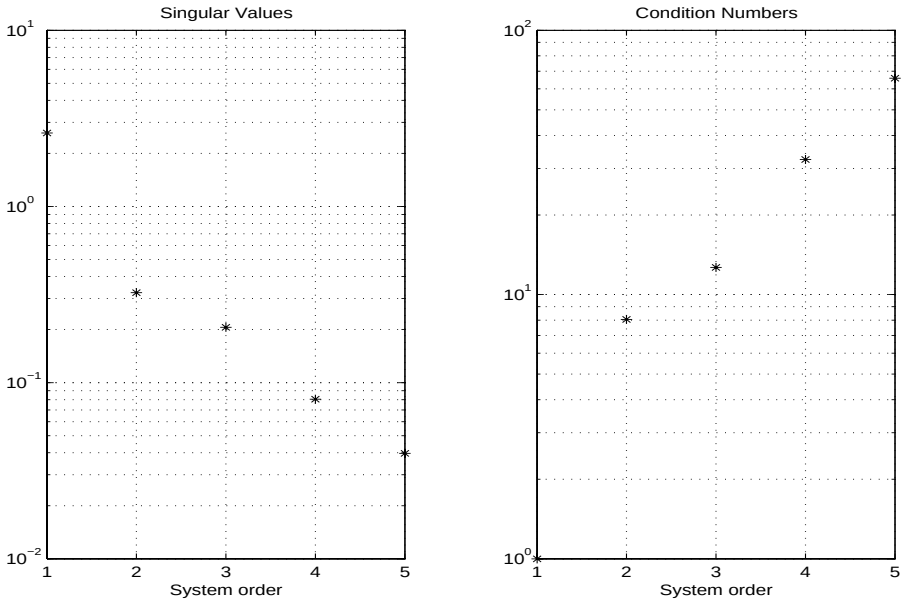

Figure 6.38: Singular value plot from DSR_e with $L = 5$ and $J = 10$ when used on Example 3, Section 3.3, when a Gaussian white noise with variance 4 is used as reference.

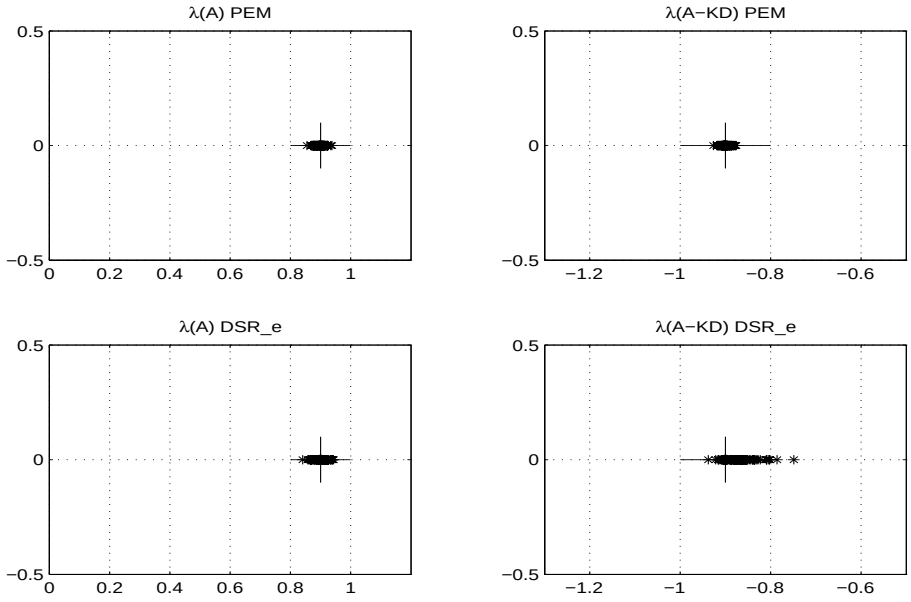

Figure 6.39: Estimates from DSR_e and PEM when used on Example 3, Section 3.3, when a Gaussian white noise with variance 4 is used as reference.

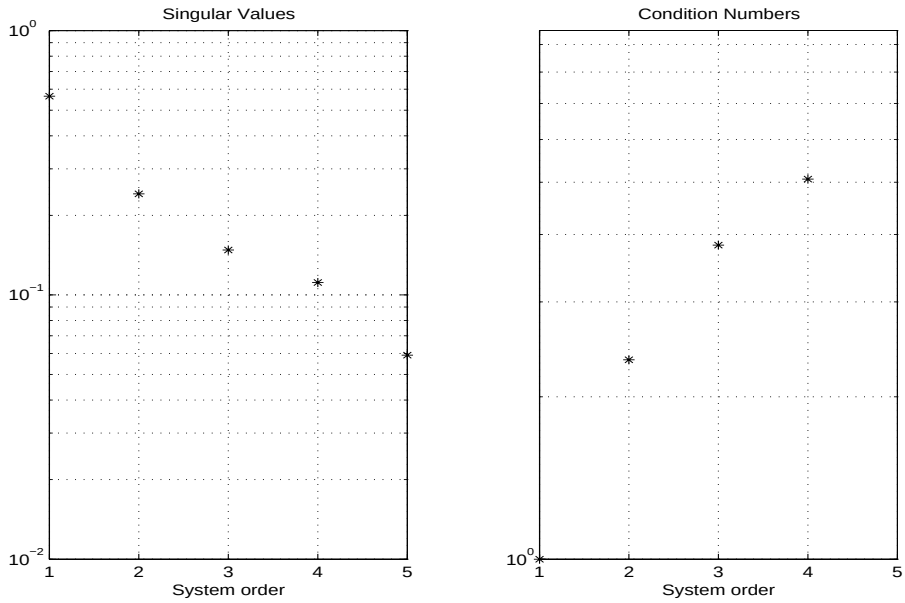

Figure 6.40: Singular value plot from DSR_e with $L = 5$ and $J = 10$ when used on Example 4, Section 3.4, when r_k^{10}, page 128, is used as reference.

assumed that the correct model order, $n = 2$, is chosen. Figure E.6 where $L = 10$ and $J = 20$ are chosen gives a clearer indication of the system order. The parameter estimates are visualized in Figure 6.41.

In this case the results coincide with the results in Section 6.3.4. The estimates of the eigenvalues are unbiased. The estimates of the zeros are unbiased. The estimation of the eigenvalues of the Kalman filter fails totally for both methods.

Figure 6.42 shows the singular value plot from DSR_e with $L = 5$ and $J = 10$ when used on Example 5, Section 3.5, using r_k^7 and r_k^8, page 90, as references.

The singular value plot indicates that this is a system with system order 3, or less. It is assumed that the correct model order, $n = 3$, is chosen. Figure E.7 where $L = 10$ and $J = 20$ are chosen gives a clearer indication of the system order. The parameter estimates are visualized in Figure 6.43.

In this case the results are slightly different than the results in Section 6.3.5. Now a small bias is introduced in the estimates of the eigenvalues from DSR_e. There is also an increase in the variance. The estimates of the zeros are still unbiased and the variance is reduced. The estimation of the eigenvalues of the Kalman filter is still unbiased, but with an increase in the variance.

The simulations show that the procedure combining the visual inspection of sin-

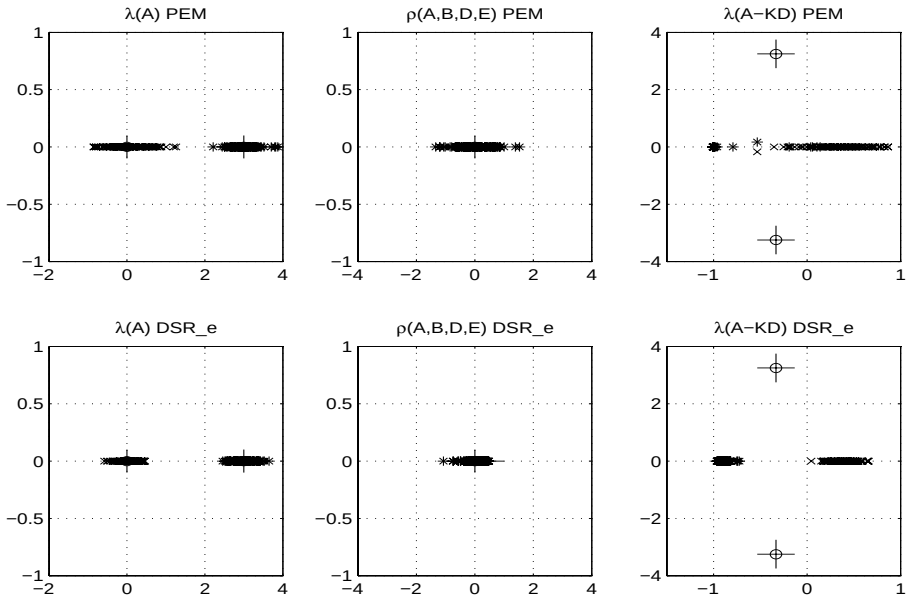

Figure 6.41: Estimates from DSR_e and PEM when used on Example 4, Section 3.4, when r_k^{10}, page 128, is used as reference.

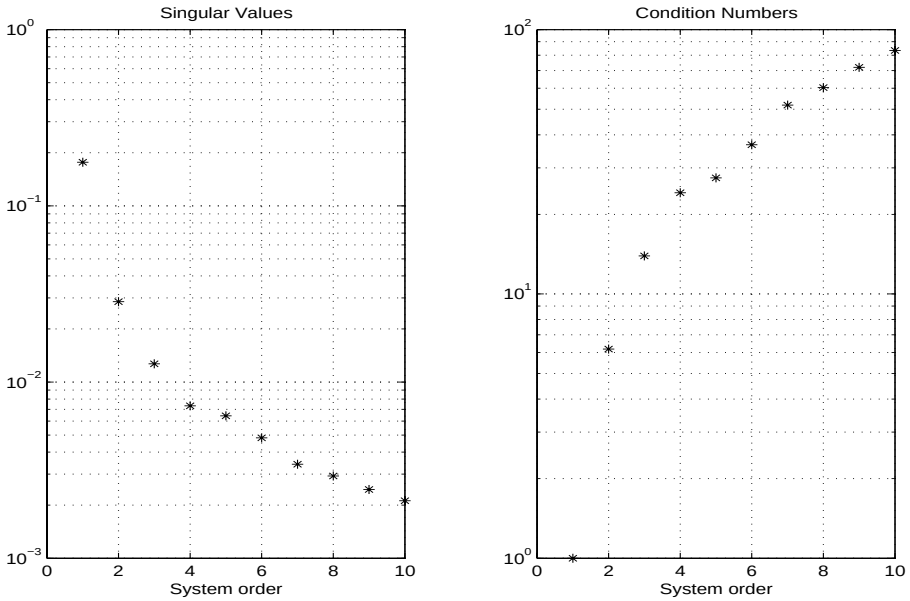

Figure 6.42: Singular value plot from DSR_e with $L = 5$ and $J = 10$ when used on Example 5, Section 3.5, when r_k^7 and r_k^8, page 90, are used as references.

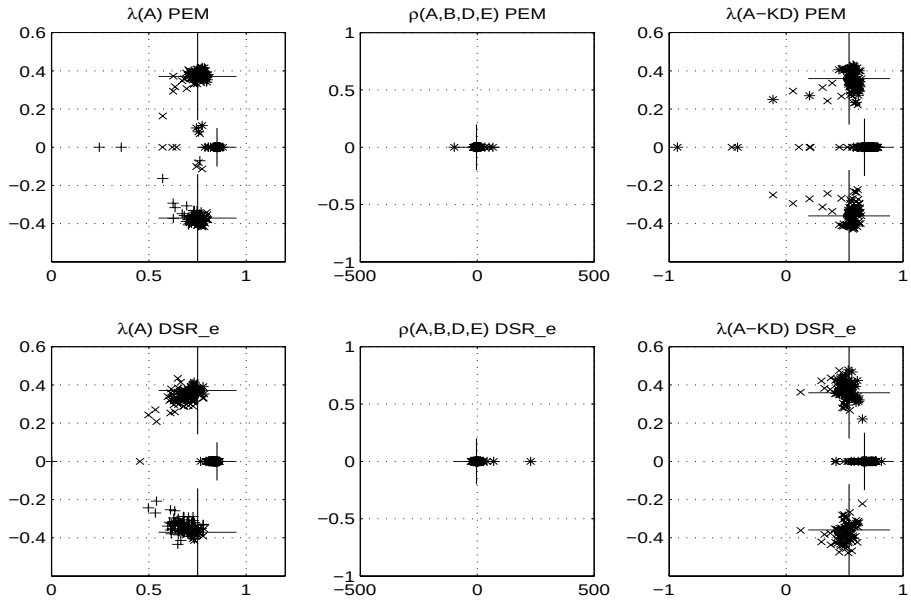

Figure 6.43: Estimates from DSR_e and PEM when used on Example 5, Section 3.5, when r_k^7 and r_k^8, page 90, are used as references.

gular values and the search for the minimum prediction error suggested is a good practical approach to the use of DSR_e.

Chapter 7

Concluding remarks

The projections used in DSR to estimate the extended observability matrix, and the eigenvalues, have been compared to the projections used in other SID algorithms. There were no indications of that any of the other projections should be used instead of the ones used in DSR, both for use on finite open and closed loop data sets.

When an appropriate dithering signal is chosen in closed loop identification the eigenvalue estimates from DSR are unbiased if the identification horizon, L, is chosen large enough. Identification of zeros is a harder task. Two solutions were found to help reducing the bias in the zeros. An increase in the level of persistent excitation of the dithering signal helped to reduce the bias in the zeros. Using a dithering signal on the input instead of the reference also helped to reduce the bias in the zeros.

It is well known that in open loop cases with a low level of persistent excitation on the input signal it is favourable to choose the identification, L, as small as possible. Therefore a version of DSR is implemented with two sets of horizons. Long horizons for identification of eigenvalues and short horizons for identification of zeros. For simplicity the method is named DSR_2LJ. The assumption is wrong that the horizon, L_z, for identification of zeros has to be chosen as small as possible when inputs with a low level of persistent excitation are used. Choosing the horizon L_z smaller than L_e gives an improvement in the estimation of the zeros.

Another approach to reduce the bias in closed loop identification is to modify the closed loop to either reduce the noise in the feedback or make the noise through the feedback uncorrelated to the noise on the output. The effect of using different types of filters in the feedback loop have been investigated. Using a filter in the feedback to the controller can reduce and in some cases remove the bias when DSR is used for estimation. The optimal filter used in the feedback is not the

noise free output or a 1^{st} order low-pass filter but the Kalman filter. The reason for this is that the Kalman filter gives an estimate of the output which contains the optimal amount of information. There is also reason to believe that measurement noise can lead to an increased excitation of the input of the system and in this way help to reduce bias and variance of the estimates from SID methods if the input and the noise on the output are uncorrelated.

A new three-step closed loop subspace identifications algorithm based on the DSR algorithm and the Kalman filter properties is presented. In an initial step DSR is used for identification of the process model, including the Kalman filter gain. This model may have a bias when the system is operating in closed loop and there is noise present. The next step is to implement the Kalman filter in the feedback in such a way that the controller uses the filtered output from the filter, not the actual process measurement. The idea is that the Kalman filter found by DSR will give an output which is sufficiently uncorrelated with the noise on the output of the actual process, and in this way reduce or eliminate the bias problem. The final step is to use DSR to identify the process model when the feedback is filtered through the Kalman filter. This model will be unbiased if the Kalman filter is correct.

The simulation studies have shown that even when a Kalman filter with a bias is used in the closed loop the estimated model in the final step seems to be unbiased from visual inspection.

It is stated that the goal for closed loop SID algorithms is to be as easy to use for direct identification on finite closed loop data sets as the original SID algorithms are to use on finite open loop data sets and in addition provide results comparable to the results from PEM. The closed loop SID algorithm named DSR_e is a solution fulfilling these requirements. The DSR_e algorithm is a modification of the existing DSR algorithm. The algorithm is based on the fact that the noise innovation process can be identified directly from the data in a first step. The estimation of the noise innovation process is consistent, both in the case of open and closed loop data.

The system identification process have been considered in two different ways. One approach is when all information regarding the process is known and a benchmark is performed to see how good the performance can be. The other is when the system order has to be estimated from the process data. In both cases DSR_e is an alternative comparable to PEM.

Methods to estimate the system order by subspace identification have been presented. The methods are meant to help users without experience in using SID algorithms. The method with the best results was the function orderfindPE.m

utilizing the squared prediction error when DSR_e is used for system identification. The method identified the system order correctly 68% of the times, or better. A search was performed for the future horizon in the interval $\frac{5}{m} \leq L \leq 10$, where m is the number of outputs, and the past horizon in the interval $10 \leq J \leq 20$ to find the system with the smallest squared prediction error. Then the systems with up to 1% larger squared prediction error were evaluated, and the system with lowest system order was chosen.

The simulation studies showed that none of the functions for identification of system order worked perfectly. Therefore a procedure combining the visual inspection of singular values from DSR_e and the search for the minimum prediction error is suggested. Visual inspection of the singular values gave the correct estimate of system order every time, independent of the choice of past and future horizons. The parameter settings for DSR_e found searching for the minimum prediction error resulted in estimates that were comparable to the estimates from PEM. This indicates that it is good practical approach for the use of DSR_e for direct closed loop system identification.

If further work is to be done it should be a test on real data where the model selection is from the validation of real data. Models must always be evaluated on validation data and compared to alternative models. It would also be of interest to compare the results to PEM and closed loop subspace identification software when used on real data.

References

Chiuso, A and Picci, G. (2004), Consistency Analysis of Certain Closed-Loop Subspace Identification Methods. Provisionally accepted for publication in Automatica, short version submitted to CDC04

Chou, C. T. and Verhaegen, M. (1997). Subspace Algorithms for the Identification of Multivariable Dynamic Errors-in-Variables Models. Automatica, Vol. 33, No. 10, pp. 1857-1997.

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

Di Ruscio, D. (1996). DSR Toolbox. Available on request.

Di Ruscio, D. (1997). A method for identification of combined deterministic stochastic systems. In :*Application 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, 6-11 December.

Di Ruscio, D. (2001). Subspace System Identification of the Kalman Filter Gain. *The European Control Conference, ECC2001, Porto, Portugal 2001.*

Di Ruscio, D. (2003a). Subspace System Identification of the Kalman Filter. Modeling, Identification and Control, Vol. 24, No. 3.

Di Ruscio, D. (2003b). Subspace System Identification. Theory and applications. Lecture notes. Telemark University College, Porsgrunn, Norway.

Di Ruscio, D. (2004). Subspace System Identification of the Kalman Filter. Internal report. Telemark University College, Porsgrunn, Norway.

Forssell, U. and Ljung, L. (1997). Closed-loop Identification Revisited. Department of Electrical Engineering, Linköping University. Technical Report LiTH-ISY-R-1959.

Gustafsson. T. (2001). Subspace identification using instrumental variable techniques. Automatica, Vol. 37, pp. 2005-2010.

Jansson, M. (2003). Subspace Identification and ARX Modelling, In the proceedings of *13th IFAC Symposium on System Identification*, 27-29 August 2003, Rotterdam, The Netherlands

Katayama, T. (2005a). Subspace Methods for System Identification. Springer.

Katayama, T. Tanaka, H. and Enomoto, T. (2005b). A simple subspace identification method of closed-loop systems using orthogonal decomposition. *Preprints 16th IFAC World Congress*, July 2005, Prague.

Ljung, L. (1999). System Identification. Theory for the User. Second Edition. Prentice Hall.

Nilsen, G. W. and Di Ruscio, D. (2003). On the Total Least Squares and Ridge Regression Problems. *Nordic Process Control Workshop 11*, Trondheim, Norway 2003.

Nilsen, G. W. and Di Ruscio, D. (2004a). Using a dithering signal in the reference to improve the estimates from subspace identification methods on closed loop data. *The 7^{th} International Conference on Dynamics and control of Process Systems, DYCOPS 7 - 2004*, Boston, USA 2004.

Nilsen, G. W. and Di Ruscio, D. (2004b). Using a dithering signal in the reference to improve the estimates from subspace identification methods on closed loop data. *The 8th World Multi-Conference on Systemics, Cybernetics and Informatics, SCI 2004,* Orlando, Florida, USA 2004.

Nilsen, G. W. and Di Ruscio, D. (2004c). A comparison of the estimates from subspace identification methods used on closed loop data. *Nordic Process Control Workshop 12,* Gothenburg, Sweden 2004.

Nilsen, G. W. and Di Ruscio, D. (2005). Closed Loop Subspace Identification. Modeling, Identification and Control, Vol. 26, No. 3.

Quin, S. J. and Ljung, L (2003). Closed-loop subspace identification with innovation estimation. In the proceedings of *13th IFAC Symposium on System Identification*, 27-29 August 2003, Rotterdam, The Netherlands

Söderström, T. and Stoica, P. (1983). Instrumental Variable methods for System Identification. Springer-Verlag.

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

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

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

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

Appendix A

Plots Example 4

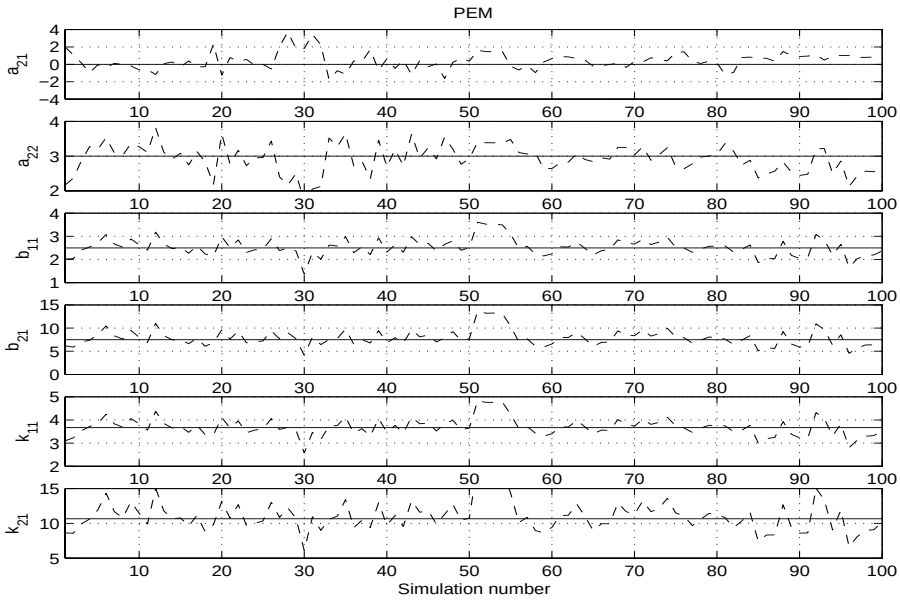

Figure A.1: Parameter vector, Equation (6.31), in Example 4 identified using PEM with $n = 2$ and $nk = 1$.

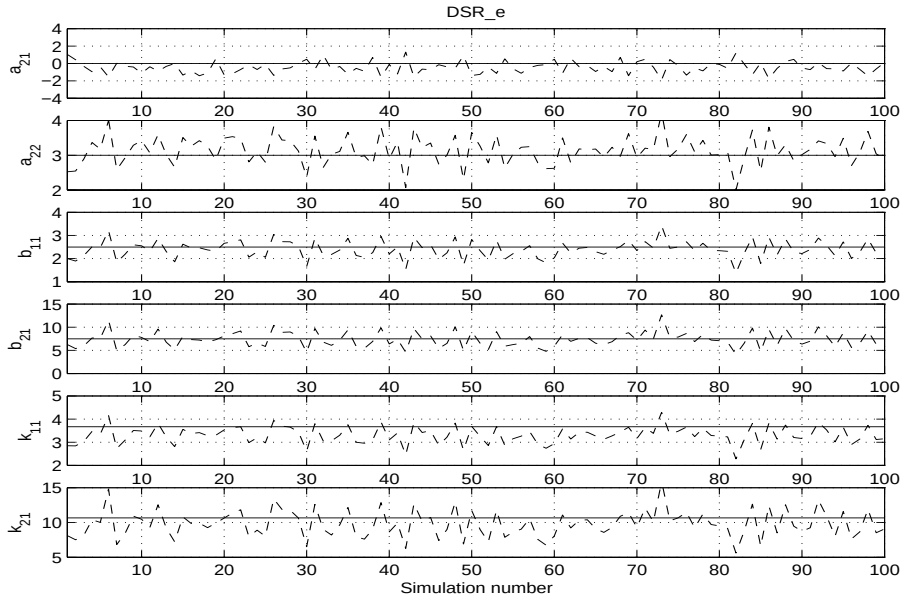

Figure A.2: Parameter vector, Equation (6.31), in Example 4 identified using DSR_e with $n = 2$, $g = 0$, $L = 2$ and $J = 3$.

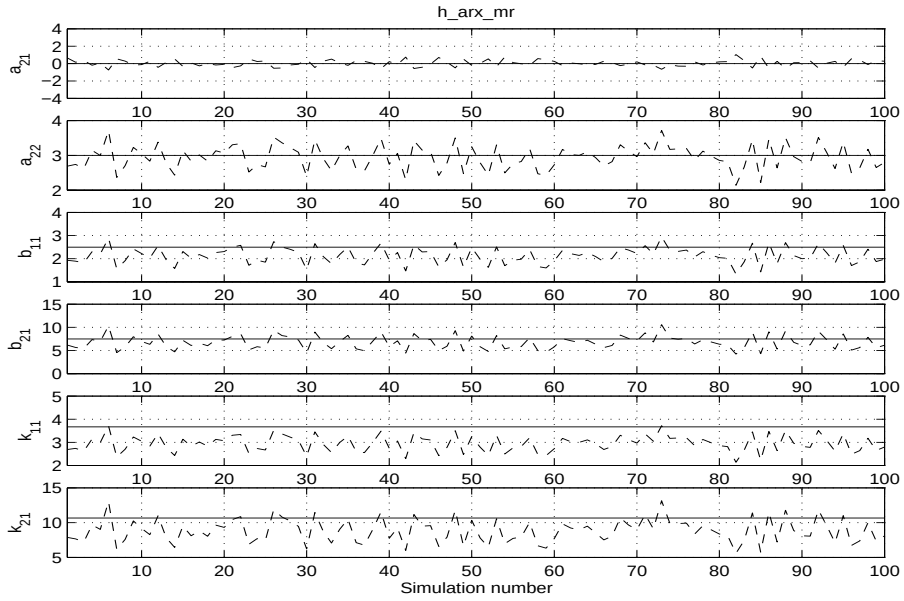

Figure A.3: Parameter vector, Equation (6.31), in Example 4 identified using h_arx_mr with $n = 2$ and the model order of the higher order ARX model used is 2 together with $L = 2$ and $J = 2$.

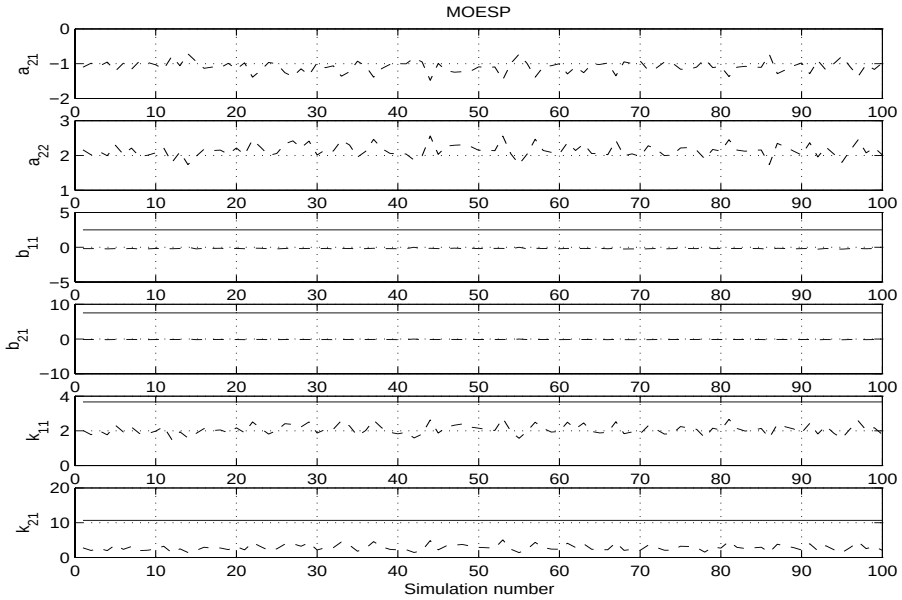

Figure A.4: Parameter vector, Equation (6.31), in Example 4 identified using MOESP with $n = 2$ and $s = 6$.

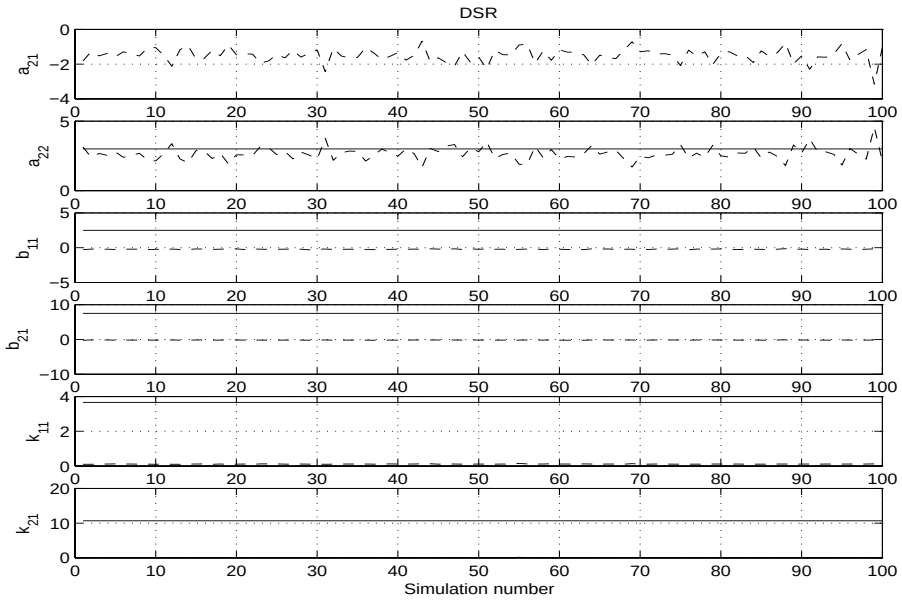

Figure A.5: Parameter vector, Equation (6.31), in Example 4 identified using DSR with $n = 2$, $g = 0$, $L = 11$ and $J = 12$.

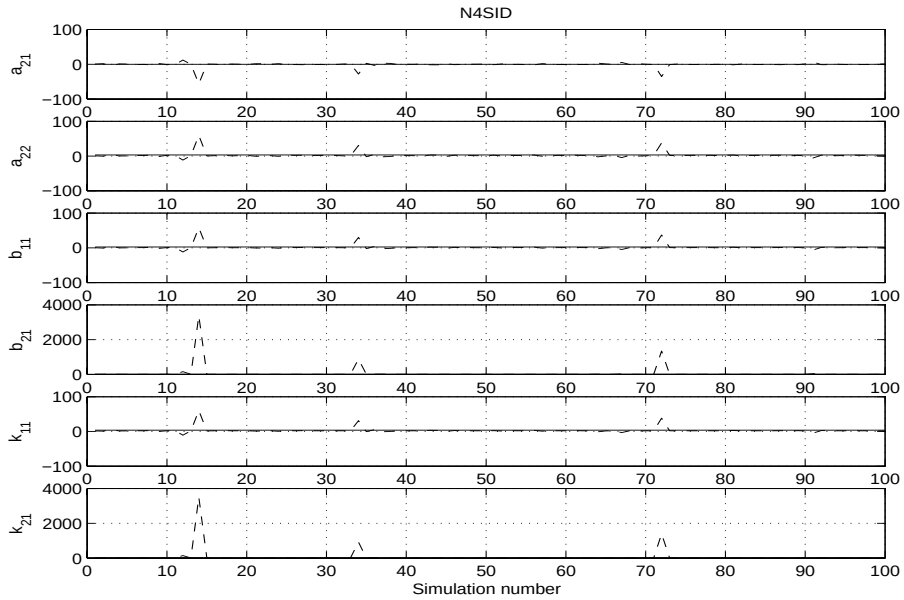

Figure A.6: Parameter vector, Equation (6.31), in Example 4 identified using N4SID with $n = 2$ and $nk = 1$.

Appendix B

Plots Example 5

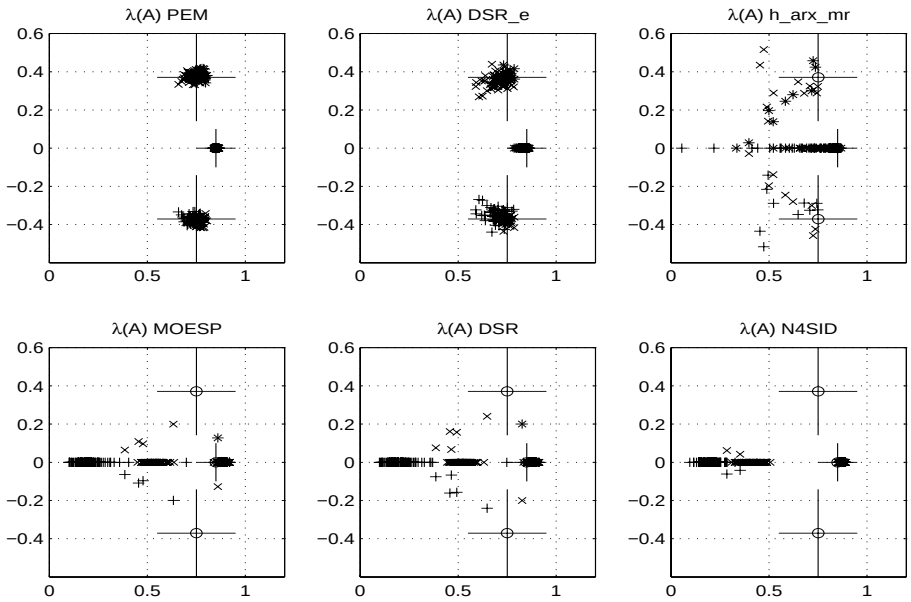

Figure B.1: The estimated eigenvalues when r_k^7 and r_k^8, page 90, are used as reference in the closed loop simulation, Example 5 Section 3.5. PEM and N4SID are used with default parameters and $nk = 1$. DSR_e is used with $g = 0$, $L = 5$ and $J = 10$. In h_arx_mr the model order of the higher order ARX model used is 10 together with $L = 10$ and $J = 10$. MOESP is used with $s = 10$. DSR is used with $g = 0$, $L = 10$ and $J = 10$.

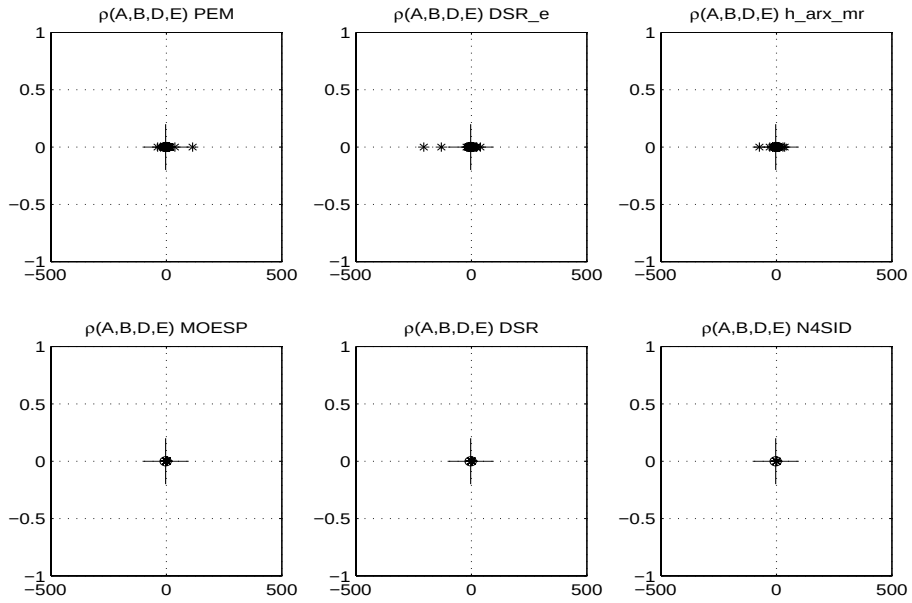

Figure B.2: The estimated zeros when r_k^7 and r_k^8, page 90, are used as reference in the closed loop simulation, Example 5 Section 3.5. PEM and N4SID are used with default parameters and $nk = 1$. DSR_e is used with $g = 0$, $L = 5$ and $J = 10$. In h_arx_mr the model order of the higher order ARX model used is 10 together with $L = 10$ and $J = 10$. MOESP is used with $s = 10$. DSR is used with $g = 0$, $L = 10$ and $J = 10$.

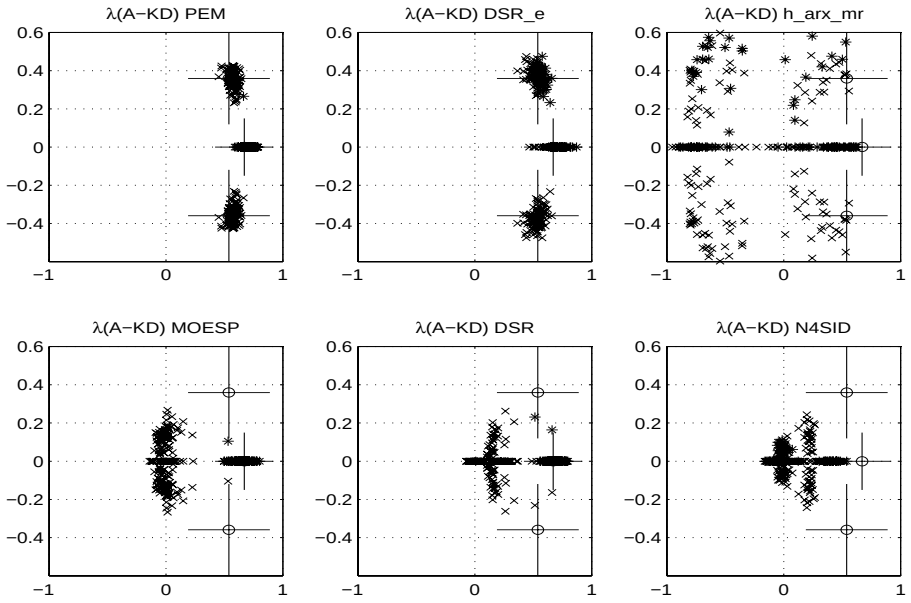

Figure B.3: The estimated eigenvalues of the Kalman filter when r_k^7 and r_k^8, page 90, are used as reference in the closed loop simulation, Example 5 Section 3.5. PEM and N4SID are used with default parameters and $nk = 1$. DSR_e is used with $g = 0$, $L = 5$ and $J = 10$. In h_arx_mr the model order of the higher order ARX model used is 10 together with $L = 10$ and $J = 10$. MOESP is used with $s = 10$. DSR is used with $g = 0$, $L = 10$ and $J = 10$.

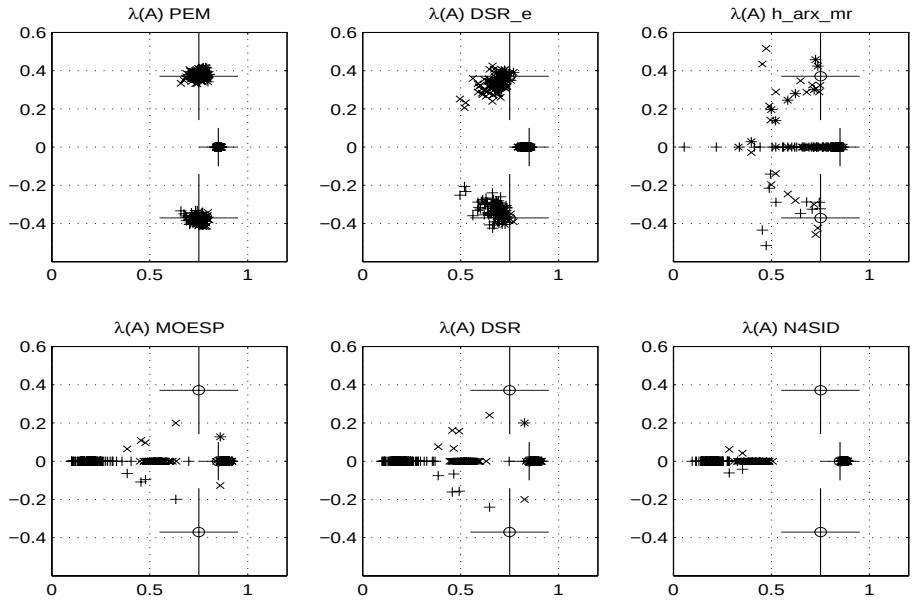

Figure B.4: The estimated eigenvalues when r_k^7 and r_k^8, page 90, are used as reference in the closed loop simulation, Example 5 Section 3.5. PEM and N4SID are used with default parameters and $nk = 1$. DSR_e is used with $g = 0$, $L = 3$ and $J = 10$. In h_arx_mr the model order of the higher order ARX model used is 10 together with $L = 10$ and $J = 10$. MOESP is used with $s = 10$. DSR is used with $g = 0$, $L = 10$ and $J = 10$.

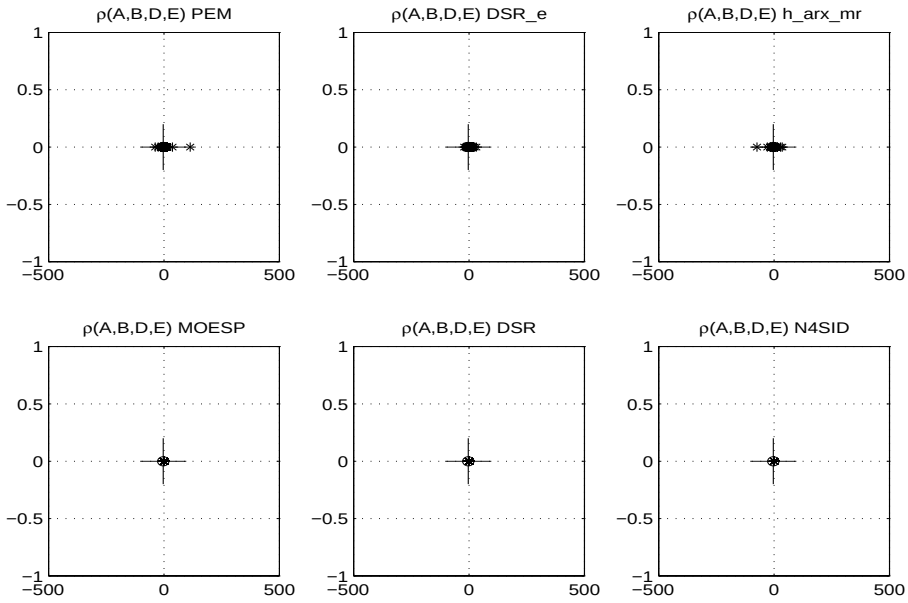

Figure B.5: The estimated zeros when r_k^7 and r_k^8, page 90, are used as reference in the closed loop simulation, Example 5 Section 3.5. PEM and N4SID are used with default parameters and $nk = 1$. DSR_e is used with $g = 0$, $L = 3$ and $J = 10$. In h_arx_mr the model order of the higher order ARX model used is 10 together with $L = 10$ and $J = 10$. MOESP is used with $s = 10$. DSR is used with $g = 0$, $L = 10$ and $J = 10$.

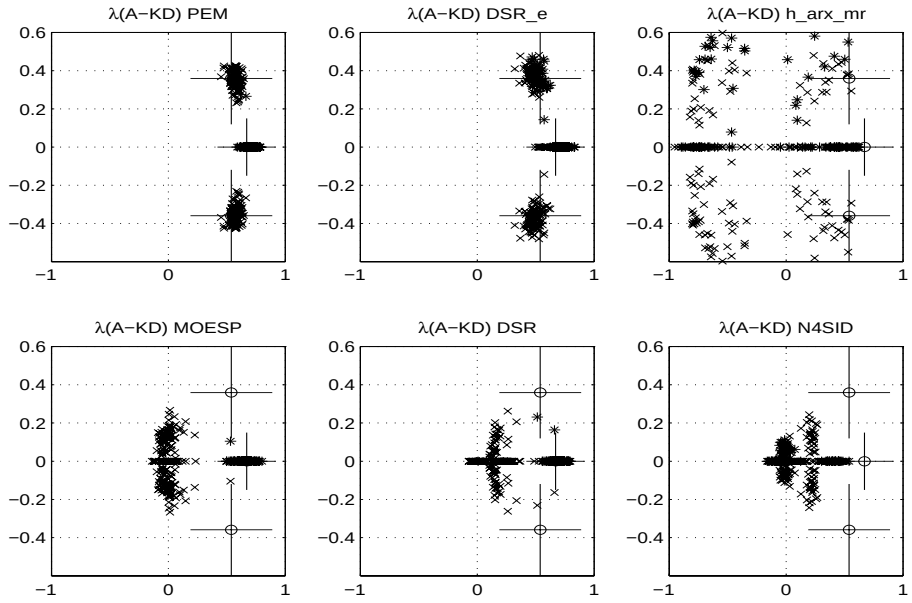

Figure B.6: The estimated eigenvalues of the Kalman filter when r_k^7 and r_k^8, page 90, are used as reference in the closed loop simulation, Example 5 Section 3.5. PEM and N4SID are used with default parameters and $nk = 1$. DSR_e is used with $g = 0$, $L = 3$ and $J = 10$. In h_arx_mr the model order of the higher order ARX model used is 10 together with $L = 10$ and $J = 10$. MOESP is used with $s = 10$. DSR is used with $g = 0$, $L = 10$ and $J = 10$.

Appendix C

Matlab code orderfindPE.m

```
function [n,L,J]=orderfindPE(Y,U,meth,g,val,n);
%[n,L,J]=orderfindPE(Y,U,meth,g,val,n);
%A function to find the system order (n), the future horizon (L) and the
% past horizon (J) that is optimal from a prediction error point of view.
%
%If val==0 all data points are used for identification. If val==1 75% of the data
%points are used for identification and 25% of the data points
%are used for validation.
%
%If meth==0 DSR is used for identification. If meth==1 DSR_e is used for
%identification (recommended if there is a feedback in the data)

if nargin < 2
    disp('Wrong number of inputs'); return
end

if nargin > 6
    disp('Wrong number of inputs');return
end

if exist('Y')==0;  disp('Output Y is missing'); end if
exist('U')==0;  disp('Input U is missing'); end if
exist('meth')==0;  meth=1; end if exist('g')==0;  g=0; end if
exist('val')==0;  val=1; end if exist('n')==0; n_mesh=1:1:6; end
if exist('n')==1; n_mesh=n; end

N=size(U,1); m=size(Y,2); Nid=ceil(0.75*N); L_mesh=1:1:ceil(10/m);
J_mesh=1:1:20; n_size=size(n_mesh,2); L_size=size(L_mesh,2);
J_size=size(J_mesh,2); if val==0
    Yid=Y; Uid=U; Yval=Y; Uval=U;
```

```
elseif val==1
    Yid=Y(1:Nid,:); Uid=U(1:Nid,:); Yval=Y(Nid+1:end,:); Uval=U(Nid+1:end,:);
end

it=1; for it1=1:n_size;
    for it2=n_mesh(it1):L_size
        for it3=L_mesh(it2):J_size
            if meth ==0 %%%DSR
                [A,B,D,E,CF,F,x0]
                    =dsr(Yid,Uid,L_mesh(it2),g,J_mesh(it3),1,n_mesh(it1));
                if val==0
                    [Ypred_val,Vpred_val,Xpred_val]
                        = dsropt(A,B,D,E,CF,F,Yid,Uid,x0);
                elseif val==1
                    [Ypred_id,Vpred_id,Xpred_id]
                        = dsropt(A,B,D,E,CF,F,Yid,Uid,x0);
                    x0_val=Xpred_id(end,:)';
                    [Ypred_val,Vpred_val,Xpred_val]
                        = dsropt(A,B,D,E,CF,F,Yval,Uval,x0_val);
                end

            elseif meth==1  %%%DSR_e
                [A,B,D,E,K,F,x0,Ef1]
                    =dsr_e(Yid,Uid,L_mesh(it2),g,J_mesh(it3),n_mesh(it1));
                CF=K*F; %to make it possible to use dsropt
                if val==0
                    [Ypred_val,Vpred_val,Xpred_val]
                        = dsropt(A,B,D,E,CF,F,Yid,Uid,x0);
                elseif val==1
                    [Ypred_id,Vpred_id,Xpred_id]
                        = dsropt(A,B,D,E,CF,F,Yid,Uid,x0);
                    x0_val=Xpred_id(end,:)';
                    [Ypred_val,Vpred_val,Xpred_val]
                        = dsropt(A,B,D,E,CF,F,Yval,Uval,x0);
                end
            end
            Re_sqpe_val(it,:)=1/N*trace((Yval-Ypred_val)'*(Yval-Ypred_val));
            Re_init_it(it,:)=[n_mesh(it1) L_mesh(it2) J_mesh(it3)];
            it=it+1;
        end
    end
end
```

```
[opt_Re_sqpe_val,ind_Re_sqpe_val]=sort(Re_sqpe_val);
PEcrit_min=opt_Re_sqpe_val(1,:);

PEcrit_min=PEcrit_min*1.01;

if exist('n')==1; n_mesh=n;
    n=n;
    L=Re_init_it(ind_Re_sqpe_val(1,:),2);
    J=Re_init_it(ind_Re_sqpe_val(1,:),3);
else
    for it4=1:50
        if opt_Re_sqpe_val(it4,:)<=PEcrit_min
            tmp(it4,:)=[Re_init_it(ind_Re_sqpe_val(it4,:),:)];
        end
    end

    [tmp2,Ind2] = sortrows(tmp,[1 2 3]);
    n=tmp2(1,1); L=tmp2(1,2); J=tmp2(1,3);
end
```

Appendix D

Matlab code orderfindSE.m

```
function [n,L,J,tmp4]=orderfindSE(Y,U,meth,g,val,n);
%[n,L,J]=orderfindSE(Y,U,meth,g,val,n);
%A function to find the system order (n), the future horizon (L) and the
% past horizon (J) that is optimal from a simulation error point of view.
%
%If val==0 all data points are used for identification. If val==1 75% of the data
%points are used for identification and 25% of the data points are used for
%validation
%
%If meth==0 DSR is used for identification. If meth==1 DSR_e is used for
%identification (recommended if there is a feedback in the data)

if nargin < 2
    disp('Wrong number of inputs'); return
end

if nargin > 6
    disp('Wrong number of inputs');return
end

if exist('Y')==0;  disp('Output Y is missing'); end if
exist('U')==0;  disp('Input U is missing'); end if
exist('meth')==0;  meth=1; end if exist('g')==0;  g=0; end if
exist('val')==0;  val=1; end if exist('n')==0; n_mesh=1:1:6; end
if exist('n')==1; n_mesh=n; end

N=size(U,1); m=size(Y,2); Nid=ceil(0.75*N); L_mesh=1:1:ceil(10/m);
J_mesh=1:1:20; n_size=size(n_mesh,2); L_size=size(L_mesh,2);
J_size=size(J_mesh,2); if val==0
    Yid=Y; Uid=U; Yval=Y; Uval=U;
```

```
elseif val==1
    Yid=Y(1:Nid,:); Uid=U(1:Nid,:); Yval=Y(Nid+1:end,:); Uval=U(Nid+1:end,:);
end

it=1; for it1=1:n_size;
    for it2=n_mesh(it1):L_size
        for it3=L_mesh(it2):J_size
            if meth ==0 %%%DSR
                [A,B,D,E,CF,F,x0]
                    =dsr(Yid,Uid,L_mesh(it2),g,J_mesh(it3),1,n_mesh(it1));
                if val==0
                    [Ysim_val,Vsim_val,Xsim_val] = dsrsim(A,B,D,E,Uid,x0);
                elseif val==1
                    [Ysim_id,Vsim_id,Xsim_id] = dsrsim(A,B,D,E,Uid,x0);
                    x0_val=Xsim_id(end,:)';
                    [Ysim_val,Vsim_val,Xsim_val] = dsrsim(A,B,D,E,Uval,x0_val);
                end

            elseif meth==1  %%DSR_e
                [A,B,D,E,K,F,x0,Ef1]
                    =dsr_e(Yid,Uid,L_mesh(it2),g,J_mesh(it3),n_mesh(it1));
                CF=K*F; %to make it possible to use dsropt
                if val==0
                    [Ysim_val] = dsrsim(A,B,D,E,Uid,x0);
                elseif val==1
                    [Ysim_id] = dsrsim(A,B,D,E,Uid,x0);
                    x0_val = x0id(Yval,Uval,A,B,D,E,zeros(size(K)),1);
                    [Ysim_val] = dsrsim(A,B,D,E,Uval,x0_val);
                end
            end
            Re_sqse_val(it,:)=1/N*trace((Yval-Ysim_val)'*(Yval-Ysim_val));
            Re_init_it(it,:)=[n_mesh(it1) L_mesh(it2) J_mesh(it3)];
            it=it+1;
        end
    end
end

[opt_Re_sqse_val,ind_Re_sqse_val]=sort(Re_sqse_val);
SEcrit_min=opt_Re_sqse_val(1,:);

SEcrit_min=SEcrit_min*1.01;

if exist('n')==1; n_mesh=n;
```

```
    n=n;
    L=Re_init_it(ind_Re_sqse_val(1,:),2);
    J=Re_init_it(ind_Re_sqse_val(1,:),3);
else
    for it4=1:50
        if opt_Re_sqse_val(it4,:)<=SEcrit_min
            tmp(it4,:)=[Re_init_it(ind_Re_sqse_val(it4,:),:)];
        end
    end

    [tmp2,Ind2] = sortrows(tmp,[1 2 3]);
    n=tmp2(1,1); L=tmp2(1,2); J=tmp2(1,3);
end
```

Appendix E

Alternative Singular Value plots, Section 6.5

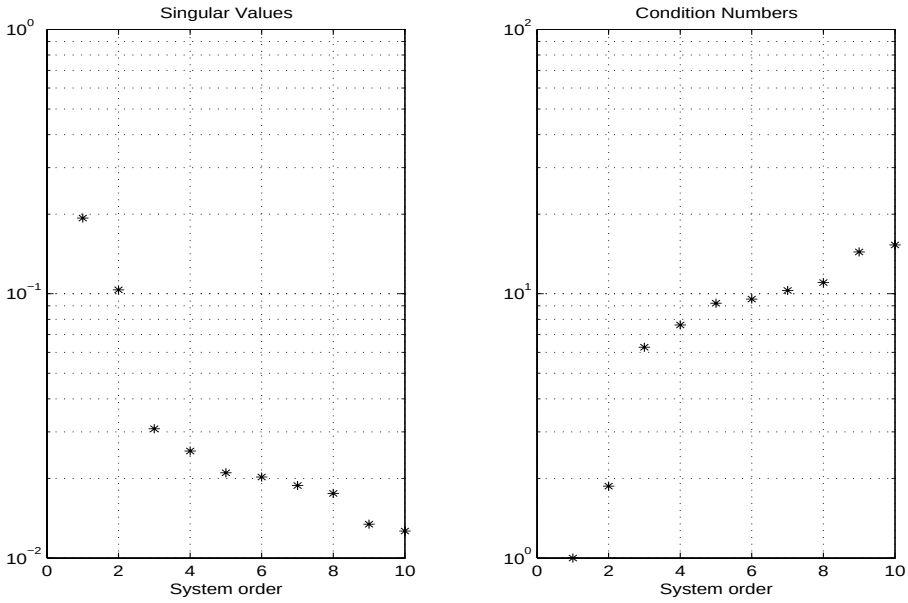

Figure E.1: Singular value plot from DSR_e with $L = 10$ and $J = 20$ when used on Example 1, Section 3.1, when a low frequent PRBS, r_k^1 page 33, is used as reference.

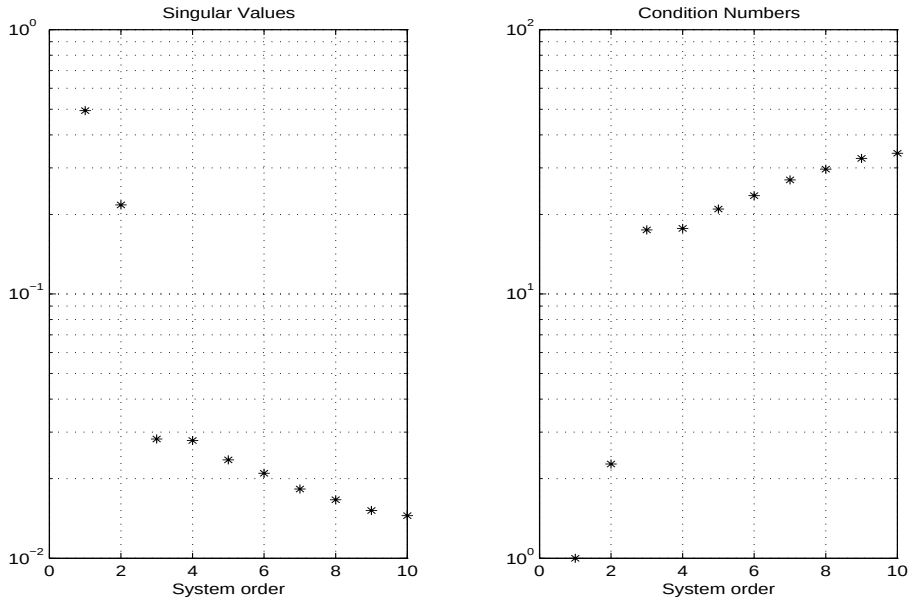

Figure E.2: Singular value plot from DSR_e with $L = 10$ and $J = 20$ when used on Example 1, Section 3.1, when a high frequent PRBS, r_k^6 page 72, is used as reference.

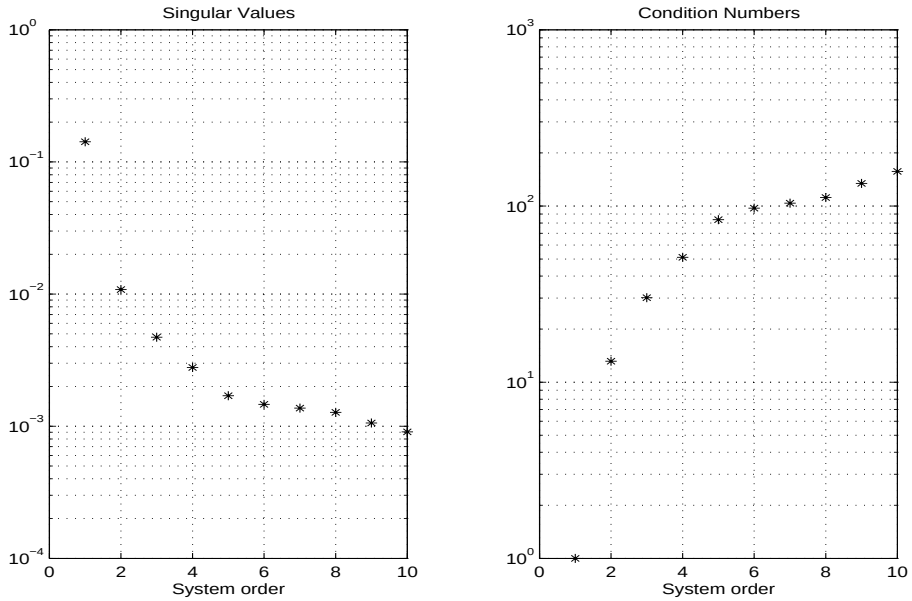

Figure E.3: Singular value plot from DSR_e with $L = 10$ and $J = 20$ when used on Example 2, Section 3.2, when a low frequent PRBS superposed a constant reference , Figure 4.11 page 39, is used as reference.

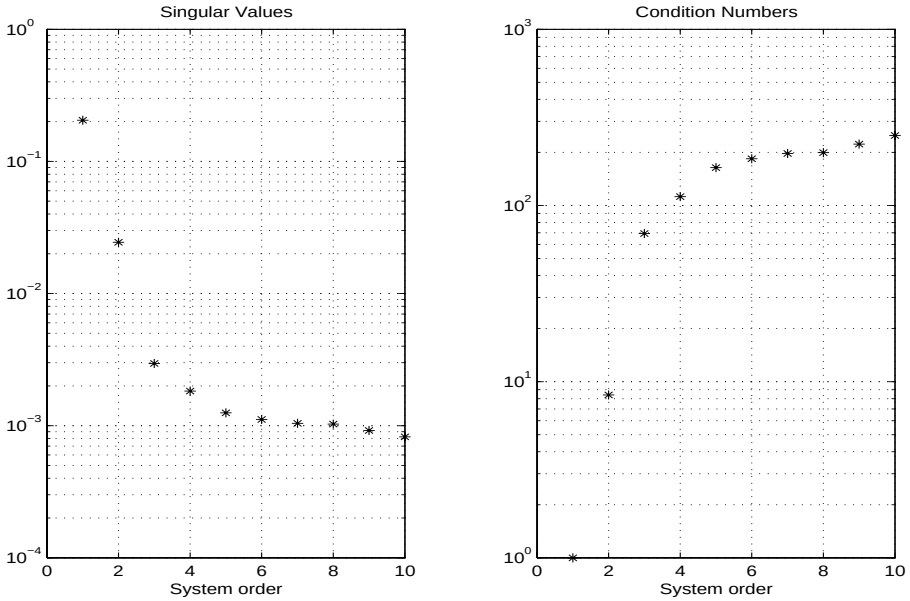

Figure E.4: Singular value plot from DSR_e with $L = 10$ and $J = 20$ when used on Example 2, Section 3.2, when a high frequent PRBS, Figure 6.28 page 141, is used as reference.

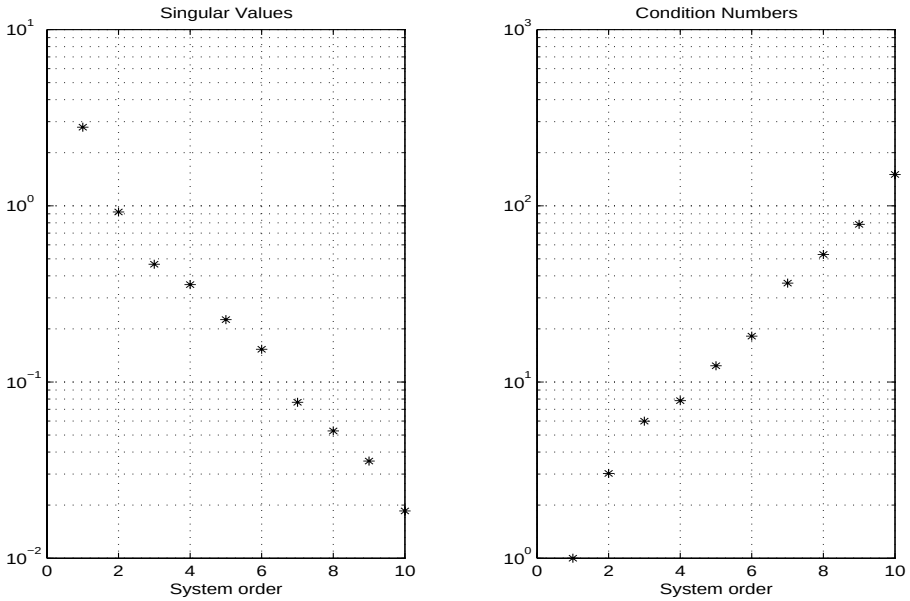

Figure E.5: Singular value plot from DSR_e with $L = 10$ and $J = 20$ when used on Example 3, Section 3.3, when a Gaussian white noise with variance 4 is used as reference.

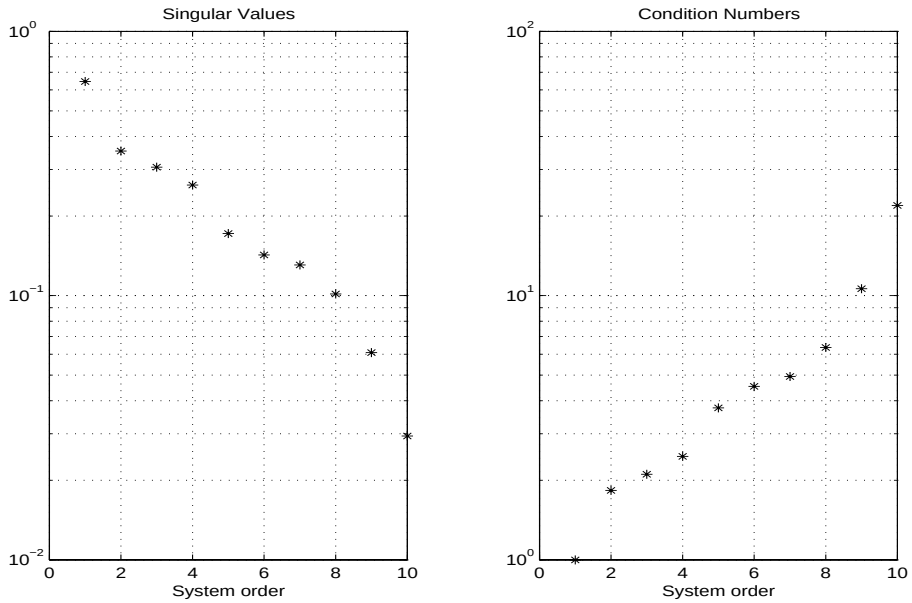

Figure E.6: Singular value plot from DSR_e with $L = 10$ and $J = 20$ when used on Example 4, Section 3.4, when r_k^{10} is used as reference.

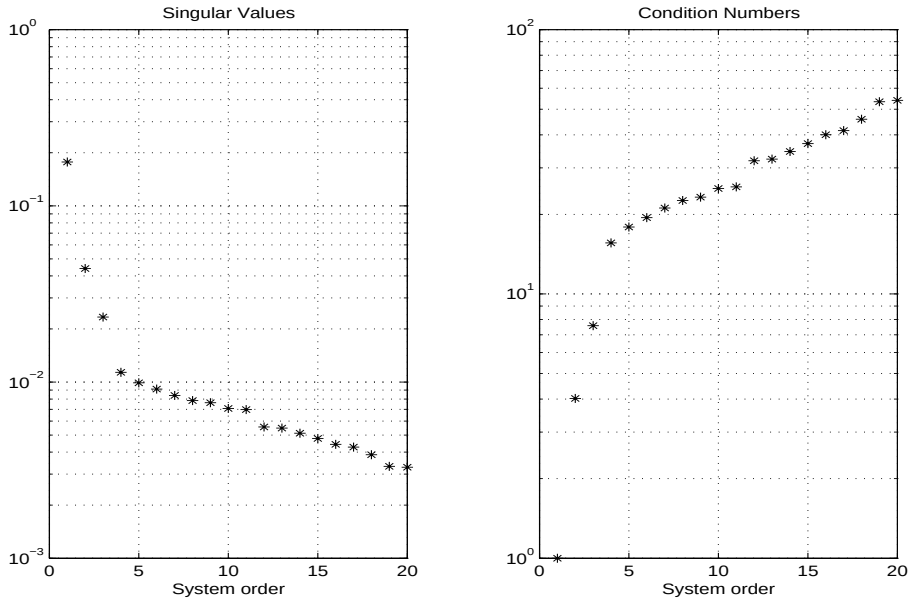

Figure E.7: Singular value plot from DSR_e with $L = 10$ and $J = 20$ when used on Example 5, Section 3.5, when r_k^7 and r_k^8, page 90, are used as references.

Appendix F

Singular Value distribution, Section 6.5

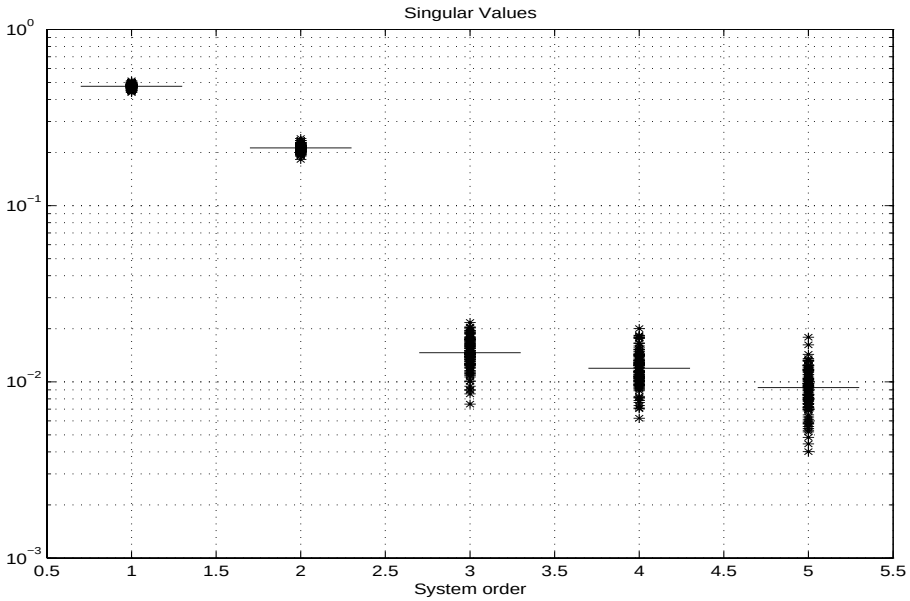

Figure F.1: The distribution of the singular values from DSR_e with $L = 5$ and $J = 10$ when used on Example 1, Section 3.1, when a high frequent PRBS, r_k^6, is used as reference.

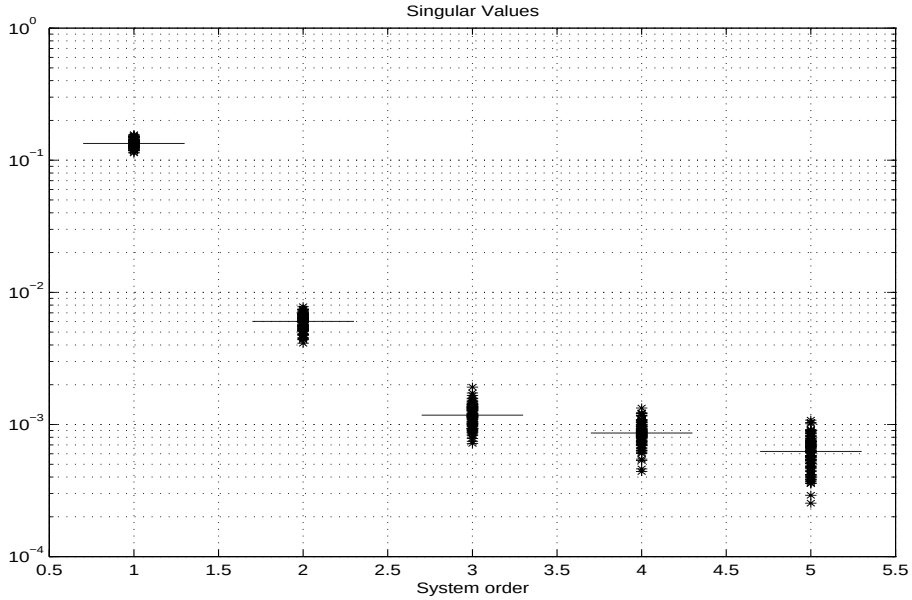

Figure F.2: The distribution of the singular values from DSR_e with $L = 5$ and $J = 10$ when used on Example 2, Section 3.2, with a constant reference superposed with a low frequent PRBS, Figure 4.11 page 39.

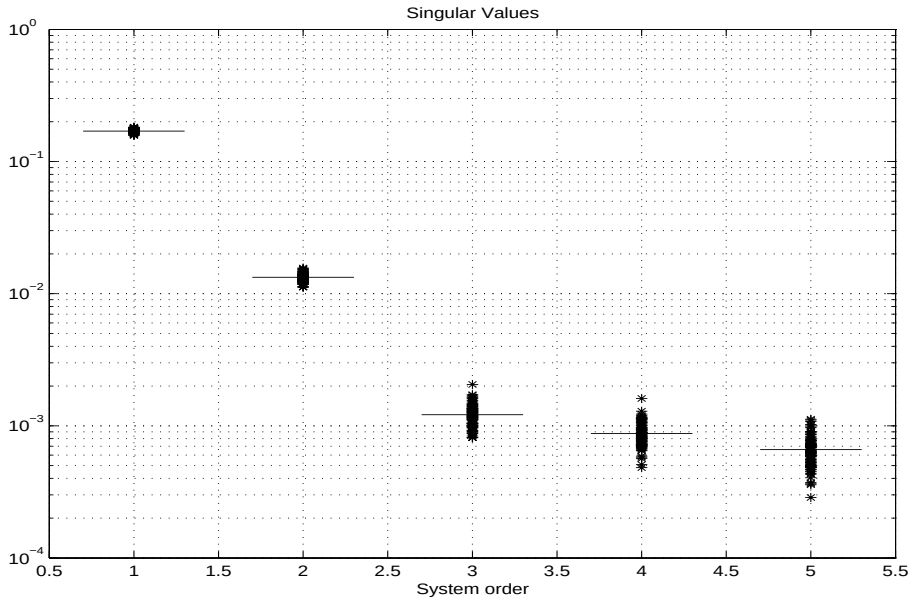

Figure F.3: The distribution of the singular values from DSR_e with $L = 5$ and $J = 10$ when used on Example 2, Section 3.2, with a constant reference superposed with a high frequent PRBS, Figure 6.28.

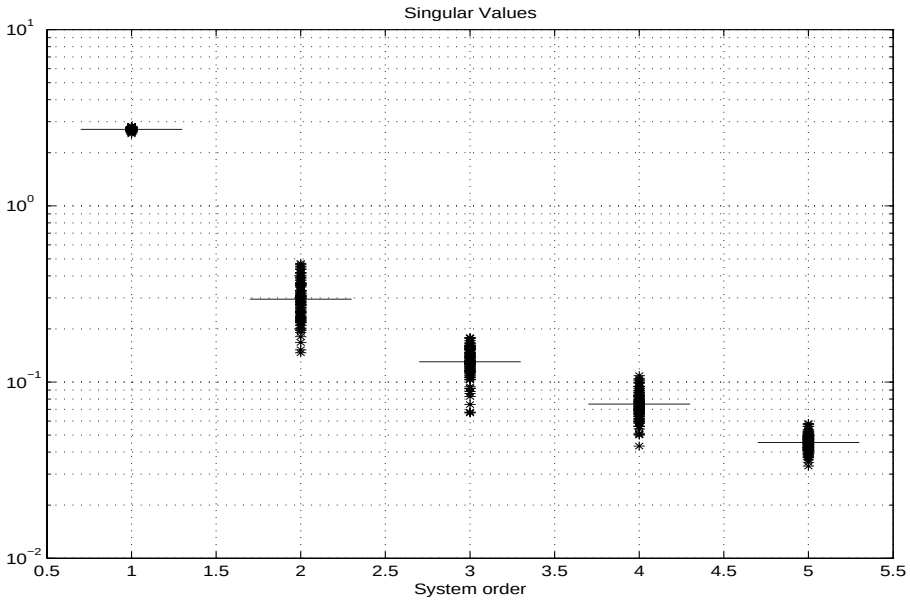

Figure F.4: The distribution of the singular values from DSR_e with $L = 5$ and $J = 10$ when used on Example 3, Section 3.3, when a Gaussian white noise with variance 4 is used as reference.

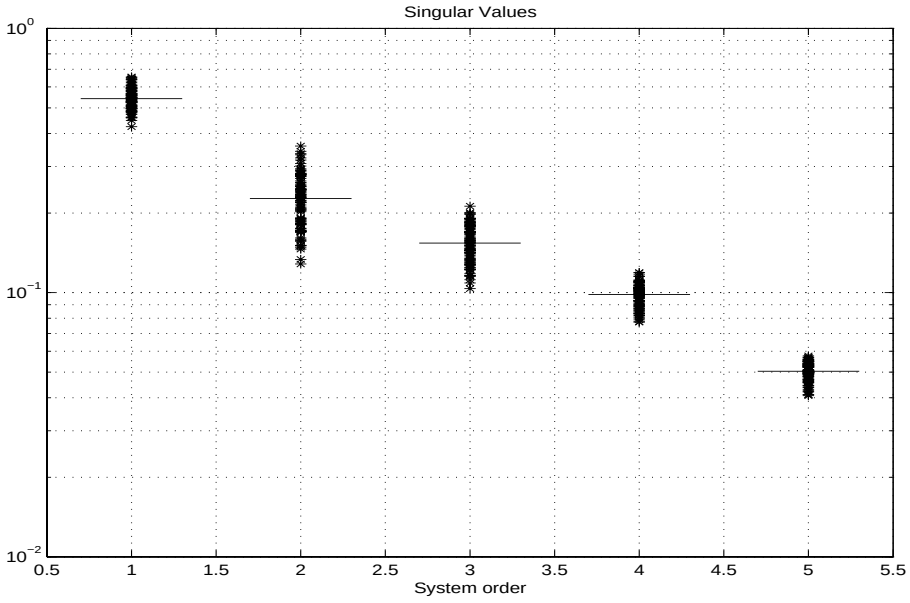

Figure F.5: The distribution of the singular values from DSR_e with $L = 5$ and $J = 10$ when used on Example 4, Section 3.4, when r_k^{10} is used as reference.

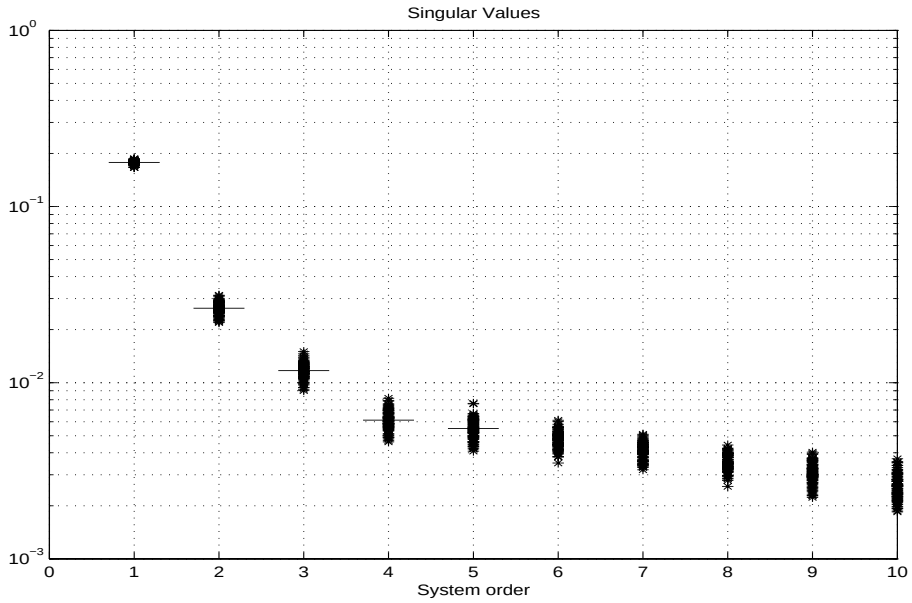

Figure F.6: Singular value plot from DSR_e with $L = 5$ and $J = 10$ when used on Example 5, Section 3.5, when r_k^7 and r_k^8 are used as references.